

`\documentclass{article}`

`\usepackage{amsmath}`

$$\frac{x}{a} = x|b$$

`\eegrn{eIntro}`

`\frac{a/b}`

$$\sqrt[n]{c} = \frac{1}{n} = \sqrt{x}$$

$$\sum_{n=1}^{\infty} n_{s=1} \frac{x^{2^2}}{2^2}$$

$$\sqrt{x}$$

`\frac{b1}{b}`

`\int nec^2`

`\ocumetriclass{amamnh}`

Dr. Kunwer Mrityunjay Singh, Ph.D

# ACADEMIC WRITING WITH LATEX

A TOOL FOR THESES, RESEARCH PAPERS,  
JOURNALS, AND ACADEMIC DOCUMENTS

# Copyright / कॉपीराइट

Copyright © 2026 Dr. Kunwer Mrityunjay Singh

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of the author.

This book is published as a digital edition and is available in PDF and EPUB formats.

**Title:** Academic Writing with LaTeX

**Author:** Dr. Kunwer Mrityunjay Singh

**ISBN:** 978-93-5773-338-0

**Edition:** First Edition, 2026

**Publisher:** Dr. Kunwer Mrityunjay Singh (Self Published)

**Location:** Uttar Pradesh, India

**Website:** [www.lecturestand.com](http://www.lecturestand.com)

कॉपीराइट © 2026 डा० कुंवर मृत्युंजय सिंह

सर्वाधिकार सुरक्षित। इस पुस्तक का कोई भी भाग लेखक की पूर्व लिखित अनुमति के बिना किसी भी रूप में या किसी भी माध्यम से—जैसे इलेक्ट्रॉनिक, यांत्रिक, फोटोकॉपी, रिकॉर्डिंग अथवा अन्य किसी माध्यम से—पुनः प्रस्तुत, संग्रहित या प्रसारित नहीं किया जा सकता।

यह पुस्तक डिजिटल संस्करण के रूप में प्रकाशित की गई है और PDF तथा EPUB प्रारूप में उपलब्ध है।

पुस्तक का शीर्षक: Academic Writing with LaTeX

लेखक: डा० कुंवर मृत्युंजय सिंह

आईएसबीएन: 978-93-5773-338-0

संस्करण: प्रथम संस्करण, 2026

प्रकाशक: डा० कुंवर मृत्युंजय सिंह (स्व-प्रकाशित)

स्थान: उत्तर प्रदेश, भारत

वेबसाइट: [www.lecturestand.com](http://www.lecturestand.com)

# Preface / प्रस्तावना

This book has been written with the objective of introducing LaTeX as a powerful and reliable document preparation system in a simple, practical, and learner-friendly manner. LaTeX is widely used in academics, research, engineering, and publishing because of its ability to handle complex formatting, mathematical expressions, references, and large documents with consistency and precision. However, many students and professionals hesitate to adopt LaTeX due to its command-based structure. This book attempts to remove that hesitation by presenting each concept clearly and progressively, supported by practical examples and classroom-oriented explanations.

The book begins by introducing the reader to LaTeX, its background, and its advantages over conventional word processing software. It then guides the learner through the complete installation process on major operating systems such as Windows, macOS, and Linux, ensuring that the reader is able to successfully set up a working LaTeX environment. This is followed by an explanation of the basic structure of LaTeX documents, helping readers understand how professional documents are logically organized using document classes, preamble settings, and sectioning commands.

As the reader progresses, the focus moves toward formatting and presentation techniques, including page layout, font selection, alignment, spacing, and style customization. The book then explores one of LaTeX's strongest features—mathematical typesetting—by covering equations, symbols, matrices, and advanced notation in a detailed yet accessible manner. This makes the book especially useful for students and professionals working in science, engineering, and mathematics.

Later chapters introduce practical document components such as figures, tables, cross-referencing, citations, and bibliography management. These topics are essential for preparing reports, theses, and research papers and are explained with clarity and relevance to real-world academic requirements. Advanced concepts such as designing custom document classes and creating professional presentations using the Beamer class are also included to extend the reader's skill set beyond basic document creation.

This book is written with a strong classroom orientation and includes bilingual support wherever necessary to assist learners from diverse linguistic backgrounds. Every effort has been made to maintain simplicity without sacrificing technical accuracy. For a better understanding of the concepts, tutorial videos are available at [www.lecturestand.com](http://www.lecturestand.com). It is hoped that this book will not only help readers learn LaTeX efficiently but will also inspire confidence and creativity in producing professional documents.

यह पुस्तक LaTeX को एक शक्तिशाली एवं विश्वसनीय दस्तावेज़ निर्माण प्रणाली के रूप में सरल, व्यावहारिक और शिक्षार्थी-अनुकूल शैली में प्रस्तुत करने के उद्देश्य से लिखी गई है। LaTeX का उपयोग शैक्षणिक, अनुसंधान, इंजीनियरिंग और प्रकाशन क्षेत्रों में व्यापक रूप से किया जाता है क्योंकि यह जटिल स्वरूपण, गणितीय अभिव्यक्तियों, संदर्भों और बड़े दस्तावेज़ों को सटीकता और समानता के साथ संभालने में सक्षम है। किंतु इसकी कमांड-आधारित संरचना के कारण कई विद्यार्थी और पेशेवर इसे अपनाने में संकोच करते हैं। यह पुस्तक प्रत्येक विषय को उदाहरणों के साथ क्रमबद्ध रूप से प्रस्तुत कर इस संकोच को दूर करने का प्रयास करती है।

पुस्तक की शुरुआत LaTeX के परिचय, इसकी पृष्ठभूमि तथा पारंपरिक वर्ड प्रोसेसर की तुलना में इसके लाभों से होती है। इसके बाद Windows, macOS और Linux जैसे प्रमुख ऑपरेटिंग सिस्टमों पर LaTeX को इंस्टॉल करने की संपूर्ण प्रक्रिया समझाई गई है, जिससे पाठक अपना स्वयं का कार्यशील वातावरण विकसित कर सकें। इसके पश्चात LaTeX दस्तावेज़ की मूल संरचना समझाई गई है, जिससे यह स्पष्ट होता है कि डॉक्यूमेंट क्लास, प्रीएम्बल और सेक्शन कमांड्स द्वारा दस्तावेज़ों को व्यवस्थित रूप से कैसे तैयार किया जाता है।

आगे के अध्यायों में दस्तावेज़ स्वरूपण और प्रस्तुति तकनीकों जैसे पेज लेआउट, फ्रॉन्ट चयन, संरक्षण, स्पेसिंग और शैली अनुकूलन पर चर्चा की गई है। इसके बाद LaTeX की सबसे सशक्त विशेषता—गणितीय टाइपसेटिंग—को विस्तार से समझाया गया है, जिसमें समीकरण, प्रतीक, मैट्रिस और उन्नत गणितीय संकेत शामिल हैं। यह अध्याय विशेष रूप से विज्ञान, इंजीनियरिंग और गणित से जुड़े विद्यार्थियों और पेशेवरों के लिए उपयोगी है।

अगले अध्यायों में चित्र, तालिकाएँ, क्रॉस-रेफरेंसिंग, सन्दर्भ और ग्रंथसूची प्रबंधन जैसे व्यावहारिक विषयों को शामिल किया गया है। ये विषय रिपोर्ट, थीसिस और शोध पत्र तैयार करने के लिए अत्यंत आवश्यक हैं और इन्हें शैक्षणिक आवश्यकताओं के अनुरूप स्पष्ट रूप से समझाया गया है। इसके अतिरिक्त कस्टम डॉक्यूमेंट क्लास बनाना तथा Beamer के माध्यम से प्रस्तुतियाँ तैयार करना जैसे उन्नत विषय भी सम्मिलित किए गए हैं।

यह पुस्तक कक्षा-केंद्रित दृष्टिकोण से लिखी गई है और विभिन्न भाषायी पृष्ठभूमि वाले विद्यार्थियों की सहायता हेतु द्विभाषी समर्थन भी प्रदान करती है। तकनीकी सटीकता बनाए रखते हुए विषय को सरल बनाने का हर संभव प्रयास किया गया है। अच्छी तरह

समझने के लिए वीडियो [www.lecturestand.com](http://www.lecturestand.com) पर देख सकते हैं। आशा है कि यह पुस्तक न केवल पाठकों को LaTeX सीखने में सहायक होगी, बल्कि उन्हें पेशेवर दस्तावेज़ तैयार करने में आत्मविश्वास और रचनात्मकता भी प्रदान करेगी।

# How to Use This Book / इस पुस्तक का उपयोग कैसे करें

This book is designed to be used both as a classroom textbook and as a self-learning guide. The structure of the chapters allows teachers to introduce concepts gradually while providing enough practical material for laboratory sessions and independent practice. Each topic is supported by examples that can be demonstrated during lectures, making learning more interactive and effective.

Teachers are encouraged to present LaTeX concepts using live demonstrations, where students can immediately compile documents and observe the output. This hands-on approach helps students understand not only the syntax but also the logic behind document formatting and layout management.

Assignments may be given at the end of selected chapters, requiring students to prepare documents such as reports, articles, or presentations using LaTeX. This will strengthen their understanding of formatting, mathematical typesetting, and structured writing. Group activities such as debugging LaTeX errors and optimizing document layouts can further enhance problem-solving skills and teamwork.

The bilingual format of this book supports learners from diverse linguistic backgrounds. Teachers may choose to explain concepts in either English or Hindi depending on the classroom environment, thereby improving comprehension and participation. With regular practice and guided instruction, students will develop confidence in using LaTeX for academic and professional purposes.

यह पुस्तक कक्षा-पुस्तक तथा स्व-अध्ययन मार्गदर्शिका दोनों रूपों में उपयोग के लिए तैयार की गई है। अध्यायों की संरचना इस प्रकार की गई है कि शिक्षक विषयों को क्रमबद्ध रूप से समझा सकें तथा प्रायोगिक सत्रों और व्यक्तिगत अभ्यास के लिए पर्याप्त सामग्री उपलब्ध हो। प्रत्येक विषय में उदाहरण दिए गए हैं जिन्हें कक्षा में प्रदर्शित किया जा सकता है, जिससे शिक्षण अधिक प्रभावी और रोचक बनता है।

शिक्षकों को सुझाव दिया जाता है कि वे LaTeX के विषयों को लाइव डेमोंस्ट्रेशन के माध्यम से पढ़ाएँ, जिससे विद्यार्थी तुरंत डॉक्यूमेंट को कंपाइल कर उसके परिणाम देख सकें। यह व्यावहारिक पद्धति विद्यार्थियों को न केवल कमांड समझने में सहायता करती है बल्कि डॉक्यूमेंट डिज़ाइन और लेआउट की तार्किक प्रक्रिया को भी स्पष्ट करती है।

चयनित अध्यायों के पश्चात असाइनमेंट दिए जा सकते हैं, जिनमें विद्यार्थियों से रिपोर्ट, लेख या प्रस्तुतीकरण तैयार कराने को कहा जाए। इससे वे फ़ॉर्मेटिंग, गणितीय टाइपसेटिंग तथा संरचित लेखन में दक्षता प्राप्त करेंगे। LaTeX त्रुटियों को ढूँढने और सुधारने जैसे समूह कार्य समस्या-समाधान कौशल तथा टीमवर्क को भी बढ़ावा देते हैं।

इस पुस्तक का द्विभाषी प्रारूप विभिन्न भाषायी पृष्ठभूमियों वाले विद्यार्थियों के लिए उपयोगी है। शिक्षक कक्षा की आवश्यकता के अनुसार अंग्रेज़ी या हिंदी किसी भी भाषा में विषय समझा सकते हैं, जिससे विषय-वस्तु की समझ अधिक स्पष्ट होती है। नियमित अभ्यास और मार्गदर्शन के साथ विद्यार्थी LaTeX में दक्ष होकर इसे शैक्षणिक और व्यावसायिक कार्यों में आत्मविश्वास के साथ उपयोग कर सकेंगे।

# Acknowledgments / आभार

There was a time in my life when I was suffering from total renal failure, and all hopes seemed shattered as life stood at the edge. At that critical moment, my mother, *Smt. Meera Singh*, saved my life by donating her kidney and granting me a second birth. Without her selfless sacrifice, neither my life nor this book would have been possible. I express my heartfelt gratitude to my father, *Shri Kunj Bihari Singh*, to whom I owe my very existence, values, and strength.

I am deeply thankful to my uncle, *Shri Lala Ram*, and my aunt, *Smt. Sarika Singh*, for their constant care, blessings, and encouragement.

After giving me a second life, my mother entrusted me to my wife, *Reshma*, whose unwavering strength, patience, and care sustained me throughout this journey. My daughter, *Anaya Singh*, lovingly known as *KuKu*, is my lifeline and the heartbeat of my existence. Their love remains my greatest personal strength.

I am grateful to my sisters and brothers-in-law for their unwavering support and affection:

<i>Dr. Minakshi Singh</i>	:	Aplinka Solutions & Technologies Pvt. Ltd.
<i>Dr. Rashmi Singh</i>	:	Institute of Seismological Research, Government of Gujarat, India
<i>Dr. Rajani Singh</i>	:	Copenhagen Business School, Denmark, Europe
<i>Dr. Richa Gautam</i>	:	Hamdard Institute of Medical Sciences & Research, New Delhi, India
<i>Romica Singh</i>	:	Union Bank of India, India
<i>Kusum Singh</i>	:	Lucknow
<i>Anand Gautam</i>	:	Gas Authority of India Limited, India
<i>Prateek Mandal</i>	:	Oil and Natural Gas Corporation, India
<i>Dr. Ashutosh Dhar Dwivedi</i>	:	Aalborg University, Denmark Europe
<i>Jasbeer Singh</i>	:	Rajbhasha Vibhag, Ministry of Home Affairs, India

I further express my gratitude to my cousins—

<i>Satish Kumar</i>	:	New Delhi
<i>Kamlesh Kumar</i>	:	Uttar Pradesh Police
<i>Aditya Kumar</i>	:	Kanpur, Uttar Pradesh
<i>Bodhisatva Vaibhav</i>	:	New Delhi
<i>Meghal</i>	:	Prayagraj, Uttar Pradesh

all of whom stood firmly by me as a constant source of support, protection, and motivation during the most challenging phases of my life.

The development of *Academic Writing with L<sup>A</sup>T<sub>E</sub>X* was greatly enriched by the guidance, encouragement, and constructive feedback of my academic mentors, colleagues, and professional peers. I sincerely acknowledge the support of

<i>Dr. Raj Bahadur Singh</i>	:	Department of Technical Education, Government of Uttar Pradesh, India
<i>Rewat</i>	:	Oil and Natural Gas Corporation, India
<i>Dr. Rajesh Devraj</i>	:	NVIDIA Corporation, India
<i>Dr. Rahul Gangopadhyay</i>	:	Indian Institute of Technology (ISM), Dhanbad, India
<i>Dr. Awnish Kumar</i>	:	National Institute of Technology, Agartala, India
<i>Dr. Satish Kumar</i>	:	Antevia Networks, United Kingdom, Europe
<i>Himanshu Maurya</i>	:	Department of Technical Education, Government of Uttar Pradesh, India
<i>Abhishek Kumar Pankaj</i>	:	Department of Technical Education, Government of Uttar Pradesh, India
<i>Dr. Neha Maurya</i>	:	New Delhi, India
<i>Pradyumna K. Siddharth</i>	:	New Delhi, India

I express my profound gratitude to my Ph.D. supervisors for their invaluable guidance and scholarly support during my doctoral studies.

Prof. Santosh Biswas : IIT Bhilai  
 Prof. Jatindra Kumar Deka : IIT Guwahati  
 Prof. Arnab Sarkar : IIT Kharagpur  
 Prof. Partha Sarathi Mandal : IIT Guwahati

I am thankful to my colleagues, seniors, and higher officials in the *D.T.E, Government of Uttar Pradesh, University of Allahabad, Ewing Christian College, Allahabad* and *IIT Guwahati* for their continued encouragement and professional support. I also acknowledge my students, especially the first batch I taught, whose curiosity and enthusiasm inspired my teaching and writing.

Finally, I express my gratitude to the global L<sup>A</sup>T<sub>E</sub>X developer community, whose dedicated efforts have made this powerful system accessible to students, educators, and researchers worldwide.

मेरे जीवन में एक समय ऐसा आया जब मैं पूर्ण वृक्क विफलता (Total Renal Failure) से पीड़ित था और जीवन मानो अंतिम किनारे पर खड़ा था। उस अत्यंत संकटपूर्ण क्षण में मेरी माता, श्रीमती मीरा सिंह, ने अपना गुर्दा दान कर मुझे जीवनदान दिया और मुझे दूसरा जन्म प्रदान किया। उनके निःस्वार्थ त्याग के बिना न तो मेरा जीवन संभव होता और न ही यह पुस्तक।

मैं अपने पिताजी, श्री कुंज बिहारी सिंह, के प्रति हृदय से कृतज्ञता व्यक्त करता हूँ, जिनके प्रति मैं अपने अस्तित्व, संस्कारों और शक्ति का ऋणी हूँ।

मैं अपने चाचा, श्री लाला राम, तथा अपनी चाची, श्रीमती सारिका सिंह, के प्रति उनकी निरंतर देखभाल, आशीर्वाद और प्रोत्साहन के लिए गहन आभार प्रकट करता हूँ।

मुझे दूसरा जीवन देने के पश्चात् मेरी माता ने मुझे मेरी पत्नी, रेशमा, को सौंपा, जिनकी अटूट शक्ति, धैर्य और स्नेह ने इस संपूर्ण यात्रा में मुझे संभाले रखा। मेरी पुत्री, अनाया सिंह, जिसे स्नेहपूर्वक कुकू कहा जाता है, मेरे जीवन की जीवनरेखा और मेरे अस्तित्व की धड़कन है। उनका प्रेम मेरी सबसे बड़ी व्यक्तिगत शक्ति है।

मैं अपनी बहनों और बहनोइयों के प्रति उनके अटूट समर्थन और स्नेह के लिए कृतज्ञ हूँ:

डॉ. मीनाक्षी सिंह : Aplinka Solutions & Technologies Pvt. Ltd.  
 डॉ. रश्मि सिंह : Institute of Seismological Research, Government of Gujarat, India  
 डॉ. रजनी सिंह : Copenhagen Business School, Denmark, Europe  
 डॉ. ऋचा गौतम : Hamdard Institute of Medical Sciences & Research, New Delhi, India  
 रोमिका सिंह : Union Bank of India, India  
 कुसुम सिंह : लखनऊ  
 आनंद गौतम : Gas Authority of India Limited, India  
 प्रतीक मंडल : Oil and Natural Gas Corporation, India  
 डॉ. आशुतोष धर द्विवेदी : Aalborg University, Denmark, Europe  
 जसबीर सिंह : राजभाषा विभाग, गृह मंत्रालय, भारत

मैं अपने चचेरे/ममेरे भाइयों के प्रति भी आभार व्यक्त करता हूँ—

सतीश कुमार : नई दिल्ली  
 कमलेश कुमार : उत्तर प्रदेश पुलिस  
 आदित्य कुमार : कानपुर, उत्तर प्रदेश  
 बोधिसत्व वैभव : नई दिल्ली  
 मेघल : प्रयागराज, उत्तर प्रदेश

जिन सभी ने मेरे जीवन के सबसे चुनौतीपूर्ण चरणों में समर्थन, संरक्षण और प्रेरणा के सतत स्रोत के रूप में दृढ़ता से मेरा साथ दिया।

Academic Writing with L<sup>A</sup>T<sub>E</sub>X के विकास में मेरे शैक्षणिक मार्गदर्शकों, सहकर्मियों और व्यावसायिक साथियों के मार्गदर्शन, प्रोत्साहन और रचनात्मक सुझावों से अत्यधिक समृद्धि प्राप्त हुई। मैं निम्नलिखित के समर्थन के प्रति अपनी हार्दिक कृतज्ञता व्यक्त करता हूँ:

डॉ. राज बहादुर सिंह	:	प्राविधिक शिक्षा विभाग, उत्तर प्रदेश सरकार, भारत
रेवत	:	Oil and Natural Gas Corporation, India
डॉ. राजेश देवराज	:	NVIDIA Corporation, India
डॉ. राहुल गंगोपाध्याय	:	Indian Institute of Technology (ISM), धनबाद, भारत
डॉ. अवनीश कुमार	:	National Institute of Technology, अगरतला, भारत
डॉ. सतीश कुमार	:	Antevia Networks, United Kingdom, Europe
हिमांशु मोर्य	:	प्राविधिक शिक्षा विभाग, उत्तर प्रदेश सरकार, भारत
अभिषेक कुमार पंकज	:	प्राविधिक शिक्षा विभाग, उत्तर प्रदेश सरकार, भारत
डॉ. नेहा मोर्य	:	नई दिल्ली, भारत
प्रद्युम्न के. सिद्धार्थ	:	नई दिल्ली, भारत

मैं अपने पीएच.डी. पर्यवेक्षकों के प्रति अपनी गहन कृतज्ञता व्यक्त करता हूँ, जिनके अमूल्य मार्गदर्शन और शैक्षणिक सहयोग ने मेरे शोधकाल को समृद्ध किया।

प्रो. संतोष बिश्वास	:	IIIT भिलाई
प्रो. जतिन्द्र कुमार डेका	:	IIIT गुवाहाटी
प्रो. अर्नब सरकार	:	IIIT खड़गपुर

मैं प्राविधिक शिक्षा विभाग, उत्तर प्रदेश सरकार, इलाहाबाद विश्वविद्यालय, इविंग क्रिश्चियन कॉलेज, इलाहाबाद तथा IIIT गुवाहाटी में अपने सहकर्मियों, वरिष्ठों और उच्चाधिकारियों का उनके निरंतर प्रोत्साहन और व्यावसायिक सहयोग के लिए आभारी हूँ। मैं अपने विद्यार्थियों का भी आभार व्यक्त करता हूँ, विशेष रूप से उस प्रथम बैच का जिसे मैंने पढ़ाया, जिनकी जिज्ञासा और उत्साह ने मेरे शिक्षण और लेखन को प्रेरित किया।

अंततः, मैं वैश्विक  $\text{KATEX}$  डेवलपर समुदाय के प्रति अपनी कृतज्ञता व्यक्त करता हूँ, जिनके समर्पित प्रयासों ने इस सशक्त प्रणाली को विश्वभर के विद्यार्थियों, शिक्षकों और शोधकर्ताओं के लिए सुलभ बनाया है।

# About the Author / लेखक के बारे में

Dr. Kunwer Mrityunjay Singh is an accomplished academician, researcher, and technologist with a Ph.D. in Computer Science and Engineering from the Indian Institute of Technology (IIT) Guwahati. He also holds an M.Tech in Computer Science and Engineering from IIT Guwahati, a B.Tech in Electronics and Communication Engineering from the University of Allahabad, and a B.Sc. degree in Physics, Chemistry, and Mathematics from Ewing Christian College, Prayagraj, Uttar Pradesh, India. His academic journey reflects a strong interdisciplinary foundation and a sustained commitment to excellence in higher education.

With over eight years of teaching experience in the Department of Technical Education, Government of Uttar Pradesh, Dr. Singh currently serves as a *Lecturer in Electronics Engineering (Gazetted)*.

In addition to his academic responsibilities, Dr. Singh has played an active role in academic administration and in advancing ICT-enabled education to benefit learners across the world. He has conceptualized and managed an online education portal within his department and led initiatives for delivering live and recorded digital lectures. He has also made key contributions to curriculum development aligned with AICTE guidelines and the National Education Policy (NEP) 2020. Furthermore, he has guided institutional initiatives related to electronic media development, creation of digital learning resources, and organization of professional training programs.

Dr. Singh's academic excellence is further reflected in his notable achievements. He has qualified various competitive examinations for government positions. In particular, he qualified the Uttar Pradesh Public Service Commission (UPPSC) Lecturer recruitment examination twice, securing second rank in one of the selections.

He also successfully qualified the prestigious Graduate Aptitude Test in Engineering (GATE) examination for three consecutive years — 2011, 2012, and 2013.

In addition, he has served in leadership roles in professional organizations and academic committees, contributing significantly to policy-level initiatives and institutional development.

Through his dual focus on cutting-edge research and student-centered teaching, Dr. Singh continues to contribute meaningfully to technical education and interdisciplinary research in India.

डॉ. कुँवर मृत्युंजय सिंह एक प्रतिष्ठित शिक्षाविद्, शोधकर्ता एवं प्रौद्योगिकी विशेषज्ञ हैं। उन्होंने भारतीय प्रौद्योगिकी संस्थान गुवाहाटी से कंप्यूटर विज्ञान एवं अभियांत्रिकी में पीएच.डी. की उपाधि प्राप्त की है। इसके अतिरिक्त उन्होंने वहीं से कंप्यूटर विज्ञान एवं अभियांत्रिकी में एम.टेक., इलाहाबाद विश्वविद्यालय से इलेक्ट्रॉनिक्स एवं संचार अभियांत्रिकी में बी.टेक., तथा इर्विंग क्रिश्चियन कॉलेज, प्रयागराज (उत्तर प्रदेश) से भौतिकी, रसायन विज्ञान एवं गणित विषयों में बी.एससी. की डिग्री प्राप्त की है। उनकी शैक्षणिक यात्रा एक सुदृढ़ अंतर्विषयक आधार तथा उच्च शिक्षा में उत्कृष्टता के प्रति निरंतर समर्पण को दर्शाती है। तकनीकी शिक्षा विभाग, उत्तर प्रदेश सरकार में आठ वर्षों से अधिक के शिक्षण अनुभव के साथ, डॉ. सिंह वर्तमान में इलेक्ट्रॉनिक्स अभियांत्रिकी में व्याख्याता (राजपत्रित) के पद पर कार्यरत हैं।

अपने शैक्षणिक दायित्वों के अतिरिक्त, डॉ. सिंह ने शैक्षणिक प्रशासन तथा सूचना एवं संचार प्रौद्योगिकी (आईसीटी) आधारित शिक्षा को बढ़ावा देने में सक्रिय भूमिका निभाई है। उन्होंने अपने विभाग में एक ऑनलाइन शिक्षा पोर्टल की परिकल्पना एवं संचालन किया तथा लाइव एवं रिकॉर्डेड डिजिटल व्याख्यानों के प्रसारण की पहल का नेतृत्व किया। उन्होंने एआईसीटीई के दिशा-निर्देशों तथा राष्ट्रीय शिक्षा नीति (एनईपी) 2020 के अनुरूप पाठ्यक्रम विकास में भी महत्वपूर्ण योगदान दिया है। इसके अतिरिक्त, उन्होंने इलेक्ट्रॉनिक मीडिया विकास, डिजिटल शिक्षण सामग्री के निर्माण तथा व्यावसायिक प्रशिक्षण कार्यक्रमों के आयोजन से संबंधित संस्थागत पहलों का मार्गदर्शन किया है।

डॉ. सिंह की शैक्षणिक उत्कृष्टता उनकी उल्लेखनीय उपलब्धियों में भी परिलक्षित होती है। उन्होंने विभिन्न सरकारी एवं प्रतिस्पर्धी सेवाओं से संबंधित परीक्षाओं को उत्तीर्ण किया है। विशेष रूप से, उन्होंने उत्तर प्रदेश लोक सेवा आयोग (UPPSC) की व्याख्याता भर्ती परीक्षा दो बार उत्तीर्ण की, जिनमें से एक अवसर पर उन्होंने द्वितीय स्थान प्राप्त किया। इसके अतिरिक्त, उन्होंने प्रतिष्ठित ग्रेजुएट एपीट्यूड टेस्ट इन इंजीनियरिंग (GATE) परीक्षा को वर्ष 2011, 2012 और 2013 में लगातार तीन बार सफलतापूर्वक उत्तीर्ण किया।

उन्होंने विभिन्न व्यावसायिक संगठनों एवं शैक्षणिक समितियों में नेतृत्वकारी भूमिकाएँ निभाते हुए नीतिगत स्तर तथा संस्थागत विकास संबंधी पहलों में भी महत्वपूर्ण योगदान दिया है।

उन्नत शोध एवं छात्र-केंद्रित शिक्षण के प्रति अपनी प्रतिबद्धता के माध्यम से, डॉ. सिंह भारत में तकनीकी शिक्षा तथा अंतर्विषयक अनुसंधान के क्षेत्र में निरंतर सार्थक योगदान दे रहे हैं।

# Contents

<b>Copyright</b>	<b>i</b>
<b>Preface</b>	<b>ii</b>
<b>In the Classroom</b>	<b>iv</b>
<b>Acknowledgments</b>	<b>v</b>
<b>About the Author / लेखक के बारे में</b>	<b>viii</b>
<b>1 Introduction to LaTeX / LaTeX परिचय</b>	<b>1</b>
1.1 What is LaTeX? / LaTeX क्या है ?	1
1.2 Markup-Based Approach / मार्कअप-आधारित दृष्टिकोण	1
1.3 History of LaTeX / LaTeX का इतिहास	1
1.4 Why use LaTeX? LaTeX का प्रयोग क्यों करें ?	2
1.5 Benefits of LaTeX over other popular tools / अन्य लोकप्रिय टूल्स की तुलना में LaTeX के लाभ	3
<b>2 Installation / इंस्टॉलेशन</b>	<b>4</b>
2.1 Choosing Between MiKTeX and TeX Live / MiKTeX और TeX Live में चयन:	4
2.1.1 MiKTeX / मिकटेक्स (MiKTeX)	4
2.1.2 TeX Live / टेक्स लाइव (TeX Live)	5
2.1.3 Considerations / विचारणीय बिंदु:	5
2.2 Install “MiKTeX” on Windows / Windows पर MiKTeX इंस्टॉल करें:	5
2.2.1 Download MiKTeX / MiKTeX डाउनलोड करें:	5
2.2.2 Run the Installer / इंस्टॉलर चलाएँ:	6
2.2.3 Installation Settings / इंस्टॉलेशन सेटिंग्स:	6
2.2.4 Download and Install Packages / पैकेज डाउनलोड और इंस्टॉल करें:	7
2.2.5 Test the Installation / इंस्टॉलेशन की जाँच करें:	7
2.3 OR Install “TeX Live” on Windows / या Windows पर TeX Live इंस्टॉल करें:	8
2.3.1 Run the Installer / इंस्टॉलर चलाएँ:	9
2.3.2 Installation Settings / इंस्टॉलेशन सेटिंग्स:	9
2.4 LaTeX Editor / LaTeX एडिटर	10
2.4.1 Install “TeXstudio” / TeXstudio इंस्टॉल करें:	11
2.5 Installing TeX Live on Linux/ Linux पर TeX Live इंस्टॉल करना	12
2.5.1 Step 1: Open the Terminal / चरण 1: टर्मिनल खोलें	13
2.5.2 Step 2: Update the System / चरण 2: सिस्टम अपडेट करें	13
2.5.3 Step 3: Install TeX Live / चरण 3: TeX Live इंस्टॉल करें	13
2.5.4 Step 4: Verify the Installation / चरण 4: इंस्टॉलेशन सत्यापित करें	13
2.5.5 Step 5: Install TeXstudio Editor (Optional) / चरण 5: TeXstudio एडिटर इंस्टॉल करें (वैकल्पिक)	15
2.6 Writing and Compiling Your First LaTeX Document Using TeXstudio / TeXstudio का उपयोग करके पहला LaTeX डॉक्यूमेंट लिखना और कंपाइल करना	16
2.6.1 Step 1: Launch TeXstudio / चरण 1: TeXstudio प्रारंभ करें	16
2.6.2 Step 2: Create a New LaTeX File / चरण 2: नई LaTeX फ़ाइल बनाएँ	16
2.6.3 Step 3: Write a Basic LaTeX Program / चरण 3: एक सरल LaTeX प्रोग्राम लिखें	16
2.6.4 Step 4: Save the File / चरण 4: फ़ाइल को सेव करें	18
2.6.5 Step 5: Compile the Document / चरण 5: डॉक्यूमेंट को कंपाइल करें	19
2.6.6 Step 6: View the Output PDF / चरण 6: PDF आउटपुट देखें	19
2.6.7 Step 7: Understanding Compile Messages / चरण 7: कंपाइलेशन संदेशों को समझना	19
2.6.8 Step 8: Modify and Recompile / चरण 8: संशोधन करें और पुनः कंपाइल करें	19

2.6.9	Workflow Summary / कार्यप्रवाह का सारांश . . . . .	20
<b>3</b>	<b>Basic Document Structure / मूल डॉक्यूमेंट संरचना</b>	<b>22</b>
3.1	LaTeX Tutorial: Document Class / LaTeX ट्यूटोरियल: डॉक्यूमेंट क्लास . . . . .	22
3.1.1	What is a Document Class? / डॉक्यूमेंट क्लास क्या होती है? . . . . .	22
3.1.2	Common Document Classes / सामान्य डॉक्यूमेंट क्लासेस . . . . .	22
3.1.3	Using a Document Class / डॉक्यूमेंट क्लास का उपयोग . . . . .	22
3.2	LaTeX Tutorial: Preamble / LaTeX ट्यूटोरियल: प्रीएम्बल . . . . .	23
3.2.1	What is Preamble? / प्रीएम्बल क्या है? . . . . .	23
3.2.2	Common Preamble Commands / सामान्य प्रीएम्बल कमांड्स . . . . .	23
3.2.3	Title, Author, and Date / शीर्षक, लेखक और तारीख . . . . .	24
3.3	LaTeX Tutorial: Sections, Subsections, and Subsubsections / LaTeX ट्यूटोरियल: सेक्शन, सब-सेक्शन और सब-सब-सेक्शन . . . . .	25
3.3.1	Sections / सेक्शन . . . . .	25
3.3.2	Subsections / सबसेक्शन . . . . .	25
3.3.3	Subsubsections / सब-सबसेक्शन . . . . .	25
3.3.4	Example / उदाहरण . . . . .	25
<b>4</b>	<b>Document Formatting / डॉक्यूमेंट का स्वरूपण</b>	<b>27</b>
4.1	Customizing Page Margins / पेज मार्जिन को अनुकूलित करना . . . . .	27
4.1.1	Using the geometry Package / geometry पैकेज का उपयोग . . . . .	27
4.2	Changing Paper Size / पेपर साइज बदलना . . . . .	28
4.2.1	Using the geometry Package / geometry पैकेज का उपयोग . . . . .	28
4.3	Changing Background Color / बैकग्राउंड रंग बदलना . . . . .	28
4.3.1	Using the pagecolor Package / pagecolor पैकेज का उपयोग . . . . .	28
4.4	Header and Footer / हेडर और फूटर . . . . .	29
4.4.1	Setting Up / सेटअप करना . . . . .	29
4.4.2	Customizing Headers / हेडर को अनुकूलित करना . . . . .	30
4.4.3	Customizing Footers / फूटर को अनुकूलित करना . . . . .	30
4.5	Font Size / फ्रॉन्ट साइज़ . . . . .	31
4.5.1	Font Size Commands in LaTeX / LaTeX में फ्रॉन्ट साइज़ कमांड्स . . . . .	31
4.5.2	Using Document Class / Document Class का प्रयोग . . . . .	32
4.5.3	Using \fontsize and \selectfont / fontsize और selectfont का उपयोग . . . . .	32
4.6	Font Styles / फ्रॉन्ट स्टाइल . . . . .	33
4.6.1	Bold Font / बोल्ड फ्रॉन्ट . . . . .	33
4.6.2	Italic Font / इटैलिक फ्रॉन्ट . . . . .	34
4.6.3	Underline Font / अंडरलाइन फ्रॉन्ट . . . . .	34
4.6.4	Emphasize Font / एम्फ़साइज़ टेक्स्ट . . . . .	35
4.6.5	Typewriter Font / टाइपराइटर फ्रॉन्ट . . . . .	35
4.7	Font Families / फ्रॉन्ट फ़ैमिली . . . . .	36
4.7.1	Serif Font Family / सेरिफ़ फ्रॉन्ट फ़ैमिली . . . . .	36
4.7.2	Sans-Serif Font Family / सैन्स-सेरिफ़ फ्रॉन्ट फ़ैमिली . . . . .	36
4.7.3	Monospace Font Family / मोनोस्पेस फ्रॉन्ट फ़ैमिली . . . . .	37
4.7.4	Using Font Packages / फ्रॉन्ट पैकेज का उपयोग . . . . .	37
4.7.5	Changing Font Color / फ्रॉन्ट का रंग बदलना . . . . .	37
4.8	Text Symbols / टेक्स्ट सिंबल . . . . .	38
4.8.1	Common Text Symbols / सामान्य टेक्स्ट सिंबल . . . . .	38
4.8.2	Typography / टाइपोग्राफी . . . . .	38
4.9	Text Alignment in LaTeX / LaTeX में टेक्स्ट अलाइनमेंट . . . . .	38
4.9.1	Center Alignment / मध्य संरेखण . . . . .	38
4.9.2	Left Alignment / बाएँ संरेखण (Left Alignment) . . . . .	39
4.9.3	Right Alignment / दाएँ संरेखण (Right Alignment) . . . . .	39

4.9.4	Left-aligned Environment / बाएँ संरेखित (Left-aligned) एनवायरनमेंट	40
4.9.5	Right-aligned Environment / दाएँ संरेखित (Right-aligned) एनवायरनमेंट	40
4.9.6	Quotations / उद्धरण (Quotations)	40
4.10	Lists / सूचियाँ	40
4.10.1	Itemize / itemize एनवायरनमेंट	41
4.10.2	Enumerate / enumerate एनवायरनमेंट	41
4.10.3	Description / description एनवायरनमेंट	42
4.11	Creating Paragraphs / पैराग्राफ बनाना	42
4.11.1	Using Blank Lines / खाली पंक्तियों का उपयोग	42
4.11.2	Using <code>\par</code> Command / <code>\par</code> कमांड का उपयोग	43
<b>5</b>	<b>Mathematics in LaTeX / LaTeX में गणित</b>	<b>44</b>
5.1	Inline and Display Math Mode / इनलाइन और डिस्प्ले मैथ मोड	44
5.1.1	Inline Math Mode / इनलाइन मैथ मोड	44
5.1.2	Display Math Mode / डिस्प्ले मैथ मोड	44
5.2	AMS Math Package / AMS मैथ पैकेज	45
5.3	Mathematical Symbols and Equations / गणितीय प्रतीक और समीकरण	45
5.3.1	Greek Letters / ग्रीक लेटर्स	45
5.3.2	Set and Relation / सेट और रिलेशन	45
5.3.3	Operators / ऑपरेटर्स	45
5.3.4	Superscripts and Subscripts / सुपरस्क्रिप्ट और सबस्क्रिप्ट	46
5.3.5	Trigonometric Functions / त्रिकोणमितीय फलन	46
5.3.6	Arrows / तीर चिह्न	46
5.4	Equation Environments / समीकरण एनवायरनमेंट	47
5.4.1	Inline Equations / इनलाइन समीकरण	47
5.4.2	Equations in Display Mode / डिस्प्ले मोड में समीकरण	47
5.5	Fractions, Roots, and Summation / भिन्न, वर्गमूल तथा योग	49
5.5.1	Fraction / भिन्न	49
5.5.2	Square Roots / वर्गमूल	50
5.5.3	Summation / संकलन (योग चिह्न)	50
5.6	Matrices / मैट्रिक्स	50
5.7	Parentheses and Brackets / कोष्ठक और ब्रैकेट	51
5.7.1	<code>bmatrix</code> (Square Brackets [ ]) / <code>bmatrix</code> (वर्ग कोष्ठक [ ])	51
5.7.2	<code>Bmatrix</code> (Curly Brackets { }) / <code>Bmatrix</code> (मध्य कोष्ठक { })	51
5.7.3	<code>vmatrix</code> (Determinant    ) / <code>vmatrix</code> (डिटरमिनेंट    )	52
5.7.4	<code>Vmatrix</code> (Norm   ) / <code>Vmatrix</code> (नॉर्म   )	52
5.8	Alignment / संरेखण (Alignment)	52
5.8.1	Using <code>align</code> with Matrices / मैट्रिक्स के साथ <code>align</code> का उपयोग	53
<b>6</b>	<b>Figures and Tables / चित्र और सारणियाँ</b>	<b>54</b>
6.1	Figures	54
6.1.1	Including Graphics / ग्राफिक्स जोड़ना	54
6.1.2	Captions / कैप्शन	55
6.1.3	Labels / लेबल	56
6.2	Sizing and Resizing Figures / चित्र का आकार बदलना	57
6.2.1	Using <code>scale</code> / <code>scale</code> का उपयोग	57
6.2.2	Using <code>width</code> / <code>width</code> का उपयोग	58
6.2.3	Using <code>height</code> / <code>height</code> का उपयोग	59
6.2.4	Subfigures / उप-चित्र (Subfigures)	60
6.3	Positioning Figures and Tables / चित्रों और सारणियों की स्थिति निर्धारित करना	60
6.4	Table and Tabular / टेबल और टैब्युलर	63
6.4.1	Using <code>tabular</code> / <code>tabular</code> का उपयोग	63
6.4.2	Using <code>table</code> with <code>tabular</code> / <code>table</code> के साथ <code>tabular</code> का उपयोग	63

6.5	Table Formatting / तालिका स्वरूपण	64
6.5.1	Left Alignment in All Columns / सभी कॉलम में बायाँ संरेखण	64
6.5.2	Center Alignment: <code>c c c</code> / केंद्र संरेखण: <code>c c c</code>	64
6.5.3	Right Alignment: <code>r r r</code> / दायाँ संरेखण: <code>r r r</code>	65
6.5.4	Mixed Alignment: <code>l c r</code> / मिश्रित संरेखण: <code>l c r</code>	65
6.5.5	Paragraph Columns: <code>p {}</code> / <code>p {}</code> कॉलम	66
6.5.6	<code>\hline</code> / <code>\hline</code>	66
6.5.7	Resizing Table / टेबल का आकार बदलना	67
6.6	Multi-row and Multi-column / मल्टी-रो और मल्टी-कॉलम	68
<b>7</b>	<b>References and Citations / संदर्भ और उद्धरण</b>	<b>70</b>
7.1	Cross-referencing	70
7.1.1	Cross-referencing Sections / अनुभागों का क्रॉस-रेफरेंस	70
7.1.2	Cross-referencing Figures / चित्रों का क्रॉस-रेफरेंस	71
7.2	Citation / उद्धरण	72
7.2.1	Step 1: Reference Data Using BibTeX / चरण 1: BibTeX द्वारा संदर्भ डेटा	72
7.2.2	Step 2: Bibliography Styles / चरण 2: bibliography शैलियाँ	72
7.2.3	Step 3: Citation Control Packages / चरण 3: उद्धरण नियंत्रण पैकेज	73
7.2.4	Overall Flow of Citation Processing / उद्धरण प्रक्रिया का समग्र प्रवाह	73
7.3	Sources and Methods for Citation in LaTeX / LaTeX में उद्धरण के स्रोत और विधियाँ	74
7.3.1	Citing Sources Using Google Scholar / Google Scholar का उपयोग करके उद्धरण	74
7.3.2	Manual Citation Using BibTeX / BibTeX द्वारा मैनुअल उद्धरण	82
7.3.3	Citing Online Websites / ऑनलाइन वेबसाइटों का उद्धरण	85
7.4	Modern Citation Using biblatex/ biblatex द्वारा आधुनिक उद्धरण	85
7.4.1	Need for biblatex/ biblatex की आवश्यकता	85
7.4.2	Key Difference from Traditional BibTeX / पारंपरिक BibTeX से मुख्य अंतर	85
7.4.3	Backend Used by biblatex/ biblatex का बैकएंड	85
7.4.4	Example of Biblatex Citation / Biblatex Citation का उदाहरण	86
7.4.5	Other biblatexCitation Commands / biblatex की अन्य उद्धरण कमांड	88
<b>8</b>	<b>LaTeX Projects for Students and Educators / छात्रों और शिक्षकों के लिए LaTeX अनुप्रयोग</b>	<b>89</b>
8.1	Write a Book Using Custom Book Class File / कस्टम बुक क्लास फ़ाइल का उपयोग करके पुस्तक लिखना	89
8.1.1	What Is a Class File <code>.cls</code> ? / क्लास फ़ाइल ( <code>.cls</code> ) क्या है?	90
8.1.2	Why Create a Custom Book Class? / कस्टम बुक क्लास क्यों बनाई जाती है?	90
8.1.3	Where the class file, chapters, main files are located? / क्लास फ़ाइल, चैप्टर और मुख्य फ़ाइलें कहाँ स्थित होती हैं?	91
8.1.4	How to write a book class file? / बुक क्लास फ़ाइल कैसे लिखी जाती है?	92
8.1.5	How to write a <code>main.tex</code> file?	96
8.1.6	How to compile and get the final book pdf? / अंतिम पुस्तक PDF कैसे तैयार करें?	98
8.2	Writing a Journal or Conference Paper Using a Given Template / दिए गए टेम्पलेट का उपयोग करके जर्नल या कॉन्फ्रेंस पेपर लिखना	99
<b>9</b>	<b>Beamer: Presentation Design / Beamer: प्रस्तुति बनाना</b>	<b>100</b>
9.1	Introduction to Beamer / Beamer का परिचय	100
9.1.1	What is Beamer? / Beamer क्या है?	100
9.1.2	Why Use Beamer for Technical Presentations / तकनीकी प्रस्तुतियों के लिए Beamer का उपयोग क्यों करें	100
9.1.3	Advantages of Beamer over PowerPoint and LibreOffice Impress / PowerPoint एवं LibreOffice Impress पर Beamer के लाभ	101
9.2	First Beamer Presentation / प्रथम Beamer प्रस्तुति	102
9.2.1	Frames and Slide Structure / Frames एवं स्लाइड संरचना	102
9.2.2	Frame Titles and Subtitles / Frame शीर्षक एवं उपशीर्षक	103
9.2.3	Multiple Frames and Frame Numbering / Multiple Frames एवं Frame Numbering	103
9.3	Writing Content in Beamer / Beamer में सामग्री लेखन	104

9.3.1	Types of Blocks / Blocks के प्रकार . . . . .	104
9.4	Themes in Beamer / Beamer में थीम्स . . . . .	105
9.4.1	Overview of Beamer Theme Architecture / Beamer Theme संरचना का अवलोकन . . . . .	105
9.4.2	Main Themes / Main Themes . . . . .	105
9.4.3	Inner Themes / Inner Themes . . . . .	107
9.4.4	Outer Themes / Outer Themes . . . . .	108
9.4.5	Color Themes / Color Themes . . . . .	109
9.4.6	Font Themes / Font Themes . . . . .	111
9.4.7	Combining and Customizing Themes / Themes का संयोजन एवं अनुकूलन . . . . .	112
9.5	Navigation and Layout . . . . .	113
9.5.1	Headlines and Footlines / Headlines एवं Footlines . . . . .	113
9.5.2	Navigation Symbols / Navigation Symbols . . . . .	114
9.5.3	Sections, Subsections, and TOC Slides / Sections, Subsections एवं TOC स्लाइड्स . . . . .	115
9.5.4	Sidebar and Miniframe Navigation / Sidebar एवं Miniframe Navigation . . . . .	115
9.6	Overlays and Animations . . . . .	117
9.6.1	Introduction to Overlays / Overlays का परिचय . . . . .	118
9.6.2	Overlay Specifications / Overlay Specifications . . . . .	118
9.6.3	The <code>\pause</code> Command / <code>\pause</code> Command . . . . .	118
9.6.4	Incremental Lists and Blocks / Incremental Lists एवं Blocks . . . . .	118
9.6.5	Best Practices for Animations / Animations के लिए सर्वोत्तम अभ्यास . . . . .	119
<b>10</b>	<b>Online Resources / ऑनलाइन संसाधन</b> . . . . .	<b>120</b>
10.1	Online LaTeX Editors / ऑनलाइन LaTeX संपादक . . . . .	120
10.1.1	Overleaf . . . . .	120
10.1.2	CoCalc . . . . .	121
10.2	Official Documentation and Learning Resources / आधिकारिक प्रलेखन एवं अध्ययन संसाधन . . . . .	121
10.2.1	CTAN (Comprehensive TeX Archive Network) / सीटैन (समग्र टेक्स अभिलेख नेटवर्क) . . . . .	121
10.3	Online Table Generators / ऑनलाइन तालिका जनरेटर . . . . .	122
10.3.1	www.tablesgenerator.com . . . . .	122
10.4	Graphics and Diagram Tools / ग्राफिक्स एवं आरेख उपकरण . . . . .	122
10.4.1	Inkscape . . . . .	122
10.4.2	TikZ . . . . .	122
10.5	Writing Journal and Conference Papers in LaTeX / LaTeX में जर्नल एवं कॉन्फ्रेंस शोध-पत्र लेखन . . . . .	123
10.5.1	Selecting the Target Journal or Conference / लक्षित जर्नल या कॉन्फ्रेंस का चयन . . . . .	123
10.5.2	Downloading the Official LaTeX Template / आधिकारिक LaTeX टेम्पलेट डाउनलोड करना . . . . .	123
10.5.3	How to Use the Template / टेम्पलेट का उपयोग कैसे करें . . . . .	125
10.5.4	Basic Structure of a Journal or Conference Paper / जर्नल या कॉन्फ्रेंस शोध-पत्र की मूल संरचना . . . . .	126
10.5.5	Important Practical Advice / महत्वपूर्ण व्यावहारिक सुझाव . . . . .	126

# 1. Introduction to LaTeX / LaTeX परिचय

## 1.1 What is LaTeX? / LaTeX क्या है ?

LaTeX (pronounced “LAY-tek” or “LAH-tek”) is a typesetting system commonly used for producing scientific and mathematical documents, technical reports, academic papers, and books. Unlike traditional word processors such as Microsoft Word, LaTeX uses a markup-based approach where the author writes structured plain text and LaTeX formats it into a professionally typeset document.

This separation between content and formatting makes LaTeX especially powerful for long documents and for writing material containing mathematical symbols, equations, and references.

LaTeX (उच्चारण “ले-टेक” या “ला-टेक”) एक टाइपसेटिंग सिस्टम है जिसका उपयोग वैज्ञानिक और गणितीय दस्तावेज़, तकनीकी रिपोर्ट, शोध-पत्र और पुस्तकों को बनाने के लिए किया जाता है। पारंपरिक वर्ड प्रोसेसर जैसे Microsoft Word की तुलना में LaTeX एक मार्कअप-आधारित प्रणाली है, जिसमें लेखक सामान्य टेक्स्ट में दस्तावेज़ की संरचना लिखता है और LaTeX उसे पेशेवर ढंग से स्वरूपित कर देता है।

सामग्री और स्वरूपण के इस अलगाव के कारण LaTeX लंबी और जटिल फ़ाइलों, विशेष रूप से गणितीय समीकरणों, वैज्ञानिक प्रतीकों और क्रॉस-रेफ़रेंस वाले दस्तावेज़ों के लिए अत्यंत उपयोगी है।

## 1.2 Markup-Based Approach / मार्कअप-आधारित दृष्टिकोण

A markup-based approach means that the author writes text along with special commands, known as markup, to describe the structure and formatting of a document. Instead of formatting text visually, the author specifies the role of each part of the content, such as headings, emphasized text, or mathematical expressions. The typesetting system then processes these commands to generate a consistently formatted and professional-looking document. In LaTeX, the emphasis is on defining structure rather than manual appearance.

मार्कअप-आधारित दृष्टिकोण का अर्थ यह है कि लेखक साधारण पाठ के साथ-साथ विशेष कमांड लिखता है, जिन्हें मार्कअप कहा जाता है, ताकि दस्तावेज़ की संरचना और स्वरूपण को परिभाषित किया जा सके। पाठ को दृश्य रूप से सजाने के बजाय, लेखक सामग्री के प्रत्येक भाग की भूमिका निर्धारित करता है, जैसे शीर्षक, उभरा हुआ पाठ या गणितीय अभिव्यक्तियाँ। इसके बाद टाइपसेटिंग सिस्टम इन कमांड्स को संसाधित करके एक समान रूप से स्वरूपित और पेशेवर दिखने वाला दस्तावेज़ तैयार करता है। LaTeX में मुख्य ज़ोर मैनुअल दिखावट के बजाय संरचना को परिभाषित करने पर होता है।

### Example:

```
\section{Introduction}
This is \textbf{bold} text and this is \textit{italic}.
```

In this example, commands such as `\section`, `\textbf`, and `\textit` act as markup. They inform LaTeX about the logical structure and formatting of the text, leaving the visual presentation to the system.

मार्कअप-आधारित दृष्टिकोण का अर्थ है कि लेखक साधारण टेक्स्ट के साथ विशेष कमांड लिखता है, जिन्हें मार्कअप कहा जाता है, ताकि डॉक्यूमेंट की संरचना और फ़ॉर्मेट को परिभाषित किया जा सके।

### उदाहरण:

```
\section{Introduction}
This is \textbf{bold} text and this is \textit{italic}.
```

इस उदाहरण में `\section`, `\textbf` और `\textit` कमांड मार्कअप के रूप में कार्य करते हैं।

## 1.3 History of LaTeX / LaTeX का इतिहास

LaTeX was created by Leslie Lamport in the early 1980s as a set of macros built on top of Donald Knuth’s TeX system. TeX, developed in the late 1970s, was designed to produce high-quality printed documents with precise typographic control, especially for mathematical material.

Lamport's macros made TeX easier to use for non-experts by providing predefined commands and structures. This combination of TeX and Lamport's macros came to be known as LaTeX. Over the years, developers around the world have contributed enhancements, packages, and tools, making LaTeX the standard for academic and scientific typesetting.

LaTeX को 1980 के दशक की शुरुआत में Leslie Lamport द्वारा विकसित किया गया था। उन्होंने Donald Knuth के TeX सिस्टम के ऊपर कुछ मैक्रोज़ बनाकर इसे सरल और अधिक प्रयोगशील बनाया। TeX, जिसे 1970 के दशक के अंत में Knuth ने विकसित किया था, उच्च-गुणवत्ता वाले मुद्रित दस्तावेज़ों और विशेष रूप से गणितीय सामग्री को सटीकता से टाइपसेट करने के लिए बनाया गया था।

Lamport के मैक्रोज़ ने TeX को उन लोगों के लिए आसान बना दिया जिन्हें गहराई से प्रोग्रामिंग का ज्ञान नहीं था। TeX और Lamport के इन मैक्रोज़ के संयोजन को LaTeX कहा गया। समय के साथ, दुनिया भर के डेवलपर्स ने कई पैकेज, सुधार और उपकरण जोड़ते हुए इसे एक मानक टाइपसेटिंग सिस्टम बना दिया।

## 1.4 Why use LaTeX? LaTeX का प्रयोग क्यों करें ?

There are several reasons why LaTeX is preferred over traditional word processors:

- **Superior typesetting quality:** Ideal for scientific formulas, equations, and structured documents.
- **Focus on content:** You write content, and LaTeX handles formatting automatically.
- **Powerful referencing:** Automatic cross-references for sections, figures, tables, and citations.
- **Version control friendly:** Documents are plain text—perfect for collaboration using Git.
- **Highly customizable:** Thousands of packages allow precise control over layout and features.
- **Free and open-source:** Works on Windows, macOS, and Linux without licensing issues.

ऐसे कई कारण हैं जिनकी वजह से LaTeX पारंपरिक वर्ड प्रोसेसर की तुलना में अधिक पसंद किया जाता है:

- उत्तम टाइपसेटिंग गुणवत्ता: वैज्ञानिक समीकरणों और जटिल दस्तावेज़ों के लिए सर्वोत्तम।
- सामग्री पर ध्यान: लेखक केवल सामग्री लिखता है—स्वरूपण LaTeX स्वयं करता है।
- मजबूत क्रॉस-रेफ़रेंसिंग: सेक्शन, चित्र, तालिका और संदर्भ स्वतः अपडेट होते हैं।
- संस्करण नियंत्रण के लिए उपयुक्त: दस्तावेज़ साधारण टेक्स्ट होते हैं, Git जैसे टूल्स के साथ आसानी से काम करते हैं।
- उच्च अनुकूलन क्षमता: हज़ारों पैकेज दस्तावेज़ के स्वरूप पर पूर्ण नियंत्रण प्रदान करते हैं।
- निःशुल्क और ओपन-सोर्स: Windows, macOS और Linux पर बिना किसी लाइसेंस के उपलब्ध।

## 1.5 Benefits of LaTeX over other popular tools / अन्य लोकप्रिय टूल्स की तुलना में LaTeX के लाभ

Table 1.1: Benefits of LaTeX over other popular tools

Feature	LaTeX	Other popular tools
Document Type	Professional typesetting system	Visual word processors
Formatting Control	Automatic and consistent	Manual and often inconsistent
Handling Large Documents	Excellent for long documents	Formatting issues in large files
Mathematical Equations	Superior and precise	Limited and less flexible
References and Citations	Fully automated (BibTeX/BibLaTeX)	Mostly manual
Cross-referencing	Accurate and automatic	Error-prone when editing
Layout Stability	Very stable layout	Layout may change unexpectedly
Focus	Content and structure based	Appearance and manual editing
Output Quality	Publication-quality PDFs	General-purpose documents
Version Control	Works well with Git	Poor version control support
Cost	Free and open-source	Paid / Internet dependent

तालिका 1.2: अन्य लोकप्रिय टूल्स की तुलना में LaTeX के लाभ

विशेषता	LaTeX	अन्य लोकप्रिय टूल्स
डॉक्यूमेंट का प्रकार	प्रोफेशनल टाइपसेटिंग सिस्टम	विजुअल वर्ड प्रोसेसर
फॉर्मेटिंग नियंत्रण	स्वचालित और एक समान	मैनुअल और अक्सर असंगत
बड़े डॉक्यूमेंट संभालना	लंबे डॉक्यूमेंट के लिए उत्कृष्ट	बड़े डॉक्यूमेंट में फॉर्मेटिंग समस्या
गणितीय समीकरण	अत्यंत सटीक और शक्तिशाली	सीमित और कम लचीला
संदर्भ और उद्धरण	पूर्णतः स्वचालित (BibTeX/BibLaTeX)	अधिकतर मैनुअल
क्रॉस-रेफरेंसिंग	सटीक और स्वचालित	एडिटिंग पर त्रुटि की संभावना
लेआउट की स्थिरता	बहुत स्थिर लेआउट	लेआउट अचानक बदल सकता है
मुख्य फोकस	कंटेंट और संरचना	दिखावट और मैनुअल एडिटिंग
आउटपुट गुणवत्ता	प्रकाशन-स्तरीय PDF	सामान्य उपयोग के डॉक्यूमेंट
वर्जन कंट्रोल	Git जैसे टूल्स के साथ उत्तम	वर्जन कंट्रोल कमजोर
लागत	मुफ्त और ओपन-सोर्स	भुगतान योग्य / इंटरनेट पर निर्भर

## 2. Installation / इंस्टॉलेशन

To install LaTeX on Windows, you need a LaTeX distribution such as MiKTeX or TeX Live. A LaTeX distribution is the core system that contains the LaTeX engine, essential packages, fonts, and tools required to compile .tex files into output formats such as PDF. You should install only one distribution—either MiKTeX or TeX Live—because installing both can cause conflicts between package managers and compilation tools. In addition to a LaTeX distribution, you also need a LaTeX editor to write and manage your documents efficiently. TeXstudio is a user-friendly editor that provides features such as syntax highlighting, auto-completion, error reporting, and an integrated PDF viewer. TeXstudio itself does not perform typesetting; instead, it uses the installed LaTeX distribution (MiKTeX or TeX Live) in the background to compile documents.

In summary, MiKTeX or TeX Live provides the LaTeX system, while TeXstudio provides a convenient interface for writing and compiling LaTeX documents.

Windows पर LaTeX इंस्टॉल करने के लिए आपको किसी LaTeX डिस्ट्रीब्यूशन की आवश्यकता होती है, जैसे MiKTeX या TeX Live। LaTeX डिस्ट्रीब्यूशन वह मुख्य प्रणाली होती है जिसमें LaTeX इंजन, आवश्यक पैकेज, फॉन्ट्स और अन्य टूल्स शामिल होते हैं, जिनकी मदद से .tex फ़ाइलों को PDF जैसे आउटपुट फ़ॉर्मेट में बदला जाता है। आपको इन दोनों में से केवल एक ही डिस्ट्रीब्यूशन इंस्टॉल करनी चाहिए—या तो MiKTeX या TeX Live—क्योंकि दोनों को एक साथ इंस्टॉल करने से पैकेज प्रबंधन और कंपाइल प्रक्रिया में टकराव (conflict) हो सकता है।

LaTeX डिस्ट्रीब्यूशन के अलावा, दस्तावेज़ लिखने और प्रबंधित करने के लिए एक LaTeX एडिटर की भी आवश्यकता होती है। TeXstudio एक उपयोगकर्ता-अनुकूल एडिटर है, जो सिंटैक्स हाइलाइटिंग, ऑटो-कम्प्लीशन, त्रुटि-संदेश और इंटीग्रेटेड PDF व्यूअर जैसी सुविधाएँ प्रदान करता है। TeXstudio स्वयं टाइपसेटिंग नहीं करता, बल्कि बैकग्राउंड में इंस्टॉल किए गए LaTeX डिस्ट्रीब्यूशन (MiKTeX या TeX Live) का उपयोग करके दस्तावेज़ को कंपाइल करता है।

संक्षेप में, MiKTeX या TeX Live LaTeX की मूल प्रणाली प्रदान करते हैं, जबकि TeXstudio LaTeX दस्तावेज़ लिखने और कंपाइल करने के लिए एक सुविधाजनक इंटरफ़ेस प्रदान करता है।

### 2.1 Choosing Between MiKTeX and TeX Live / MiKTeX और TeX Live में चयन:

When selecting a LaTeX distribution, such as MiKTeX or TeX Live, the decision often hinges on individual preferences and specific needs. Here are some key considerations:

जब MiKTeX या TeX Live जैसे LaTeX वितरण का चयन किया जाता है, तो निर्णय प्रायः व्यक्तिगत पसंद और विशेष आवश्यकताओं पर निर्भर करता है। नीचे कुछ मुख्य बिंदु दिए गए हैं:

#### 2.1.1 MiKTeX / मिकटेक्स (MiKTeX)

- **Installation and Updates:** Smaller initial download size, with on-demand package installation and a convenient package manager for updates.
- **Windows Focus:** Well-suited for Windows users, offering a seamless integration experience.
- **On-Demand Package Installation:** Benefits users with limited storage space as packages are installed on-demand.
- **इंस्टॉलेशन और अपडेट:** प्रारंभिक डाउनलोड छोटा होता है, पैकेज आवश्यकतानुसार इंस्टॉल होते हैं, और अपडेट के लिए एक सुविधाजनक पैकेज मैनेजर उपलब्ध होता है।
- **Windows केंद्रित:** Windows उपयोगकर्ताओं के लिए उपयुक्त है और सहज एकीकरण अनुभव प्रदान करता है।
- **ऑन-डिमांड पैकेज इंस्टॉलेशन:** सीमित स्टोरेज स्पेस वाले उपयोगकर्ताओं के लिए लाभकारी, क्योंकि पैकेज आवश्यकता पर ही इंस्टॉल होते हैं।

### 2.1.2 TeX Live / टेक्स लाइव (TeX Live)

- **Cross-Platform:** Cross-platform compatibility, working well on Windows, macOS, and various Linux distributions.
- **Comprehensive:** Tends to be more comprehensive, including a wide range of packages and tools.
- **Stability:** Perceived by some users as more stable and less prone to issues.
- **Community Support:** Strong community support, often preferred in academic and professional settings.
- क्रॉस-प्लेटफॉर्म: Windows, macOS और विभिन्न Linux वितरणों पर अच्छी तरह कार्य करता है।
- व्यापक: इसमें बड़ी संख्या में पैकेज और टूल शामिल होते हैं।
- स्थिरता: कुछ उपयोगकर्ताओं द्वारा इसे अधिक स्थिर और कम समस्याओं वाला माना जाता है।
- सामुदायिक समर्थन: मजबूत समुदाय समर्थन, अकादमिक और पेशेवर वातावरण में अधिक पसंद किया जाता है।

### 2.1.3 Considerations / विचारणीय बिंदु:

- **User Preference:** The choice may come down to personal preferences and the features that matter most to you.
- **System Resources:** Consider available storage space, with MiKTeX's on-demand installation being advantageous for space-conscious users.
- **Specific Requirements:** Depending on specific needs, such as packages, tools, or compatibility, one distribution might be more suitable than the other.
- व्यक्तिगत पसंद: चयन आपकी व्यक्तिगत प्राथमिकताओं और आपके लिए महत्वपूर्ण सुविधाओं पर निर्भर करता है।
- सिस्टम संसाधन: उपलब्ध स्टोरेज स्पेस पर विचार करें; कम स्पेस में MiKTeX का ऑन-डिमांड इंस्टॉलेशन लाभदायक होता है।
- विशेष आवश्यकताएँ: पैकेज, टूल या संगतता जैसी आवश्यकताओं के आधार पर एक वितरण दूसरे से अधिक उपयुक्त हो सकता है।

In summary, both MiKTeX and TeX Live are reputable LaTeX distributions, and the “better” choice depends on operating system, personal preferences, and specific requirements. Users often find both distributions reliable, so experimenting with both may help in choosing the one that aligns best with your workflow.

संक्षेप में, MiKTeX और TeX Live दोनों ही प्रतिष्ठित LaTeX वितरण हैं, और "बेहतर" विकल्प ऑपरेटिंग सिस्टम, व्यक्तिगत प्राथमिकताओं और विशेष आवश्यकताओं पर निर्भर करता है। उपयोगकर्ता आमतौर पर दोनों को विश्वसनीय पाते हैं, इसलिए दोनों का प्रयोग करना कार्यशैली के अनुसार सही चयन करने में सहायक होता है।

## 2.2 Install “MiKTeX” on Windows / Windows पर MiKTeX इंस्टॉल करें:

You can use a distribution like MiKTeX or TeX Live. Here's a step-by-step guide using MiKTeX, a popular choice: आप MiKTeX या TeX Live जैसे किसी वितरण का उपयोग कर सकते हैं। नीचे MiKTeX के माध्यम से एक चरण-दर-चरण मार्गदर्शिका दी गई है, जो एक लोकप्रिय विकल्प है:

### 2.2.1 Download MiKTeX / MiKTeX डाउनलोड करें:

1. Visit the MiKTeX website: [MiKTeX Download Page](#) as shown in Figure 2.1.
2. Download the MiKTeX installer suitable for your system (32-bit or 64-bit).
1. MiKTeX वेबसाइट पर जाएँ: [MiKTeX डाउनलोड पेज](#) जैसा कि चित्र 2.1 में दर्शाया गया है।
2. अपने सिस्टम के अनुसार उपयुक्त इंस्टॉलर डाउनलोड करें (32-बिट या 64-बिट)।

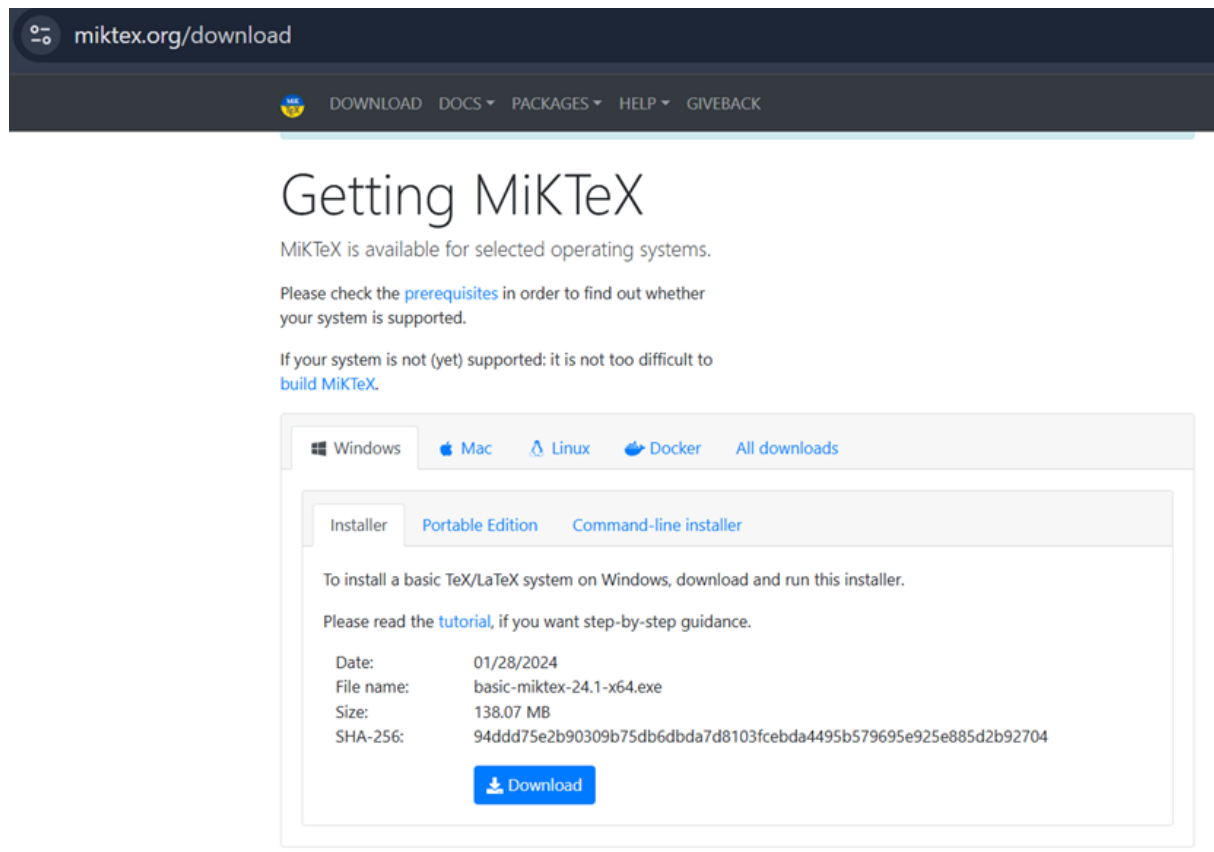


Figure 2.1: Downloading MikTeX

## 2.2.2 Run the Installer / इंस्टॉलर चलाएँ:

1. Double-click on the downloaded installer (usually a .exe file) illustrated in Figure 2.2.
  2. Choose “Install for anyone using this computer” or “Only for me” based on your preference illustrated in Figure 2.3.
  3. Follow the on-screen instructions to complete the installation.
- 1.
  2. डाउनलोड किए गए इंस्टॉलर (आमतौर पर एक .exe फ़ाइल) पर डबल-क्लिक करें, जैसा कि चित्र 2.2 में दर्शाया गया है।
  3. अपनी आवश्यकता के अनुसार “Install for anyone using this computer” या “Only for me” विकल्प चुनें, जैसा कि चित्र 2.3 में दर्शाया गया है।
  4. इंस्टॉलेशन पूरा करने के लिए ऑन-स्क्रीन निर्देशों का पालन करें।

## 2.2.3 Installation Settings / इंस्टॉलेशन सेटिंग्स:

1. During the installation, you will be prompted to choose a setup mode. The “Recommended” option is suitable for most users. Click “Next” to proceed.
2. Select a preferred paper size, and click “Next” as shown in Figure 2.4.
3. Choose a preferred installation directory or leave the default, and click “Next” as shown in Figure 2.5.

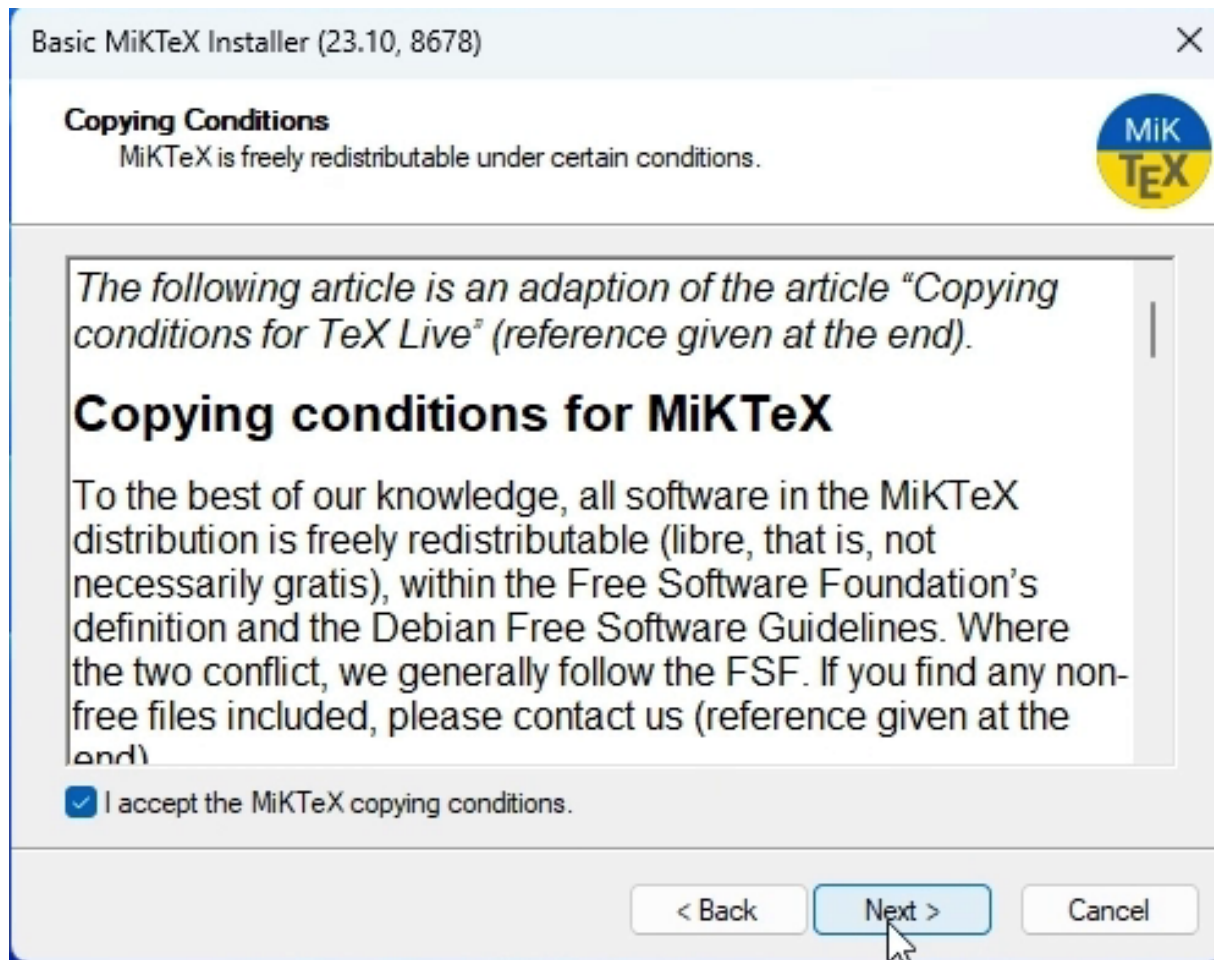


Figure 2.2: Installation Step 1

1. इंस्टॉलेशन के दौरान, आपसे सेटअप मोड चुनने को कहा जाएगा। अधिकांश उपयोगकर्ताओं के लिए "Recommended" उपयुक्त होता है। आगे बढ़ने के लिए "Next" पर क्लिक करें।
2. अपनी पसंद का पेपर साइज चुनें और "Next" पर क्लिक करें। करें, जैसा कि चित्र 2.4 में दर्शाया गया है।
3. इंस्टॉलेशन डायरेक्टरी चुनें या डिफ़ॉल्ट रहने दें, फिर "Next" पर क्लिक करें। जैसा कि चित्र 2.5 में दर्शाया गया है।

#### 2.2.4 Download and Install Packages / पैकेज डाउनलोड और इंस्टॉल करें:

1. MiKTeX will connect to the internet to download and install the necessary packages. This process may take some time based on your internet speed as shown in Figure 2.6.
2. Once the installation is complete, click "Next" and then "Finish" to exit the installer.

1. MiKTeX आवश्यक पैकेज डाउनलोड और इंस्टॉल करने के लिए इंटरनेट से जुड़ेगा। आपकी इंटरनेट गति के अनुसार इसमें समय लग सकता जैसा कि चित्र 2.6 में दर्शाया गया है।
2. इंस्टॉलेशन पूर्ण होने पर "Next" और फिर "Finish" पर क्लिक करके इंस्टॉलर बंद करें।

#### 2.2.5 Test the Installation / इंस्टॉलेशन की जाँच करें:

1. Open MiKTeX Console from the Start menu as shown in Figure 2.7.

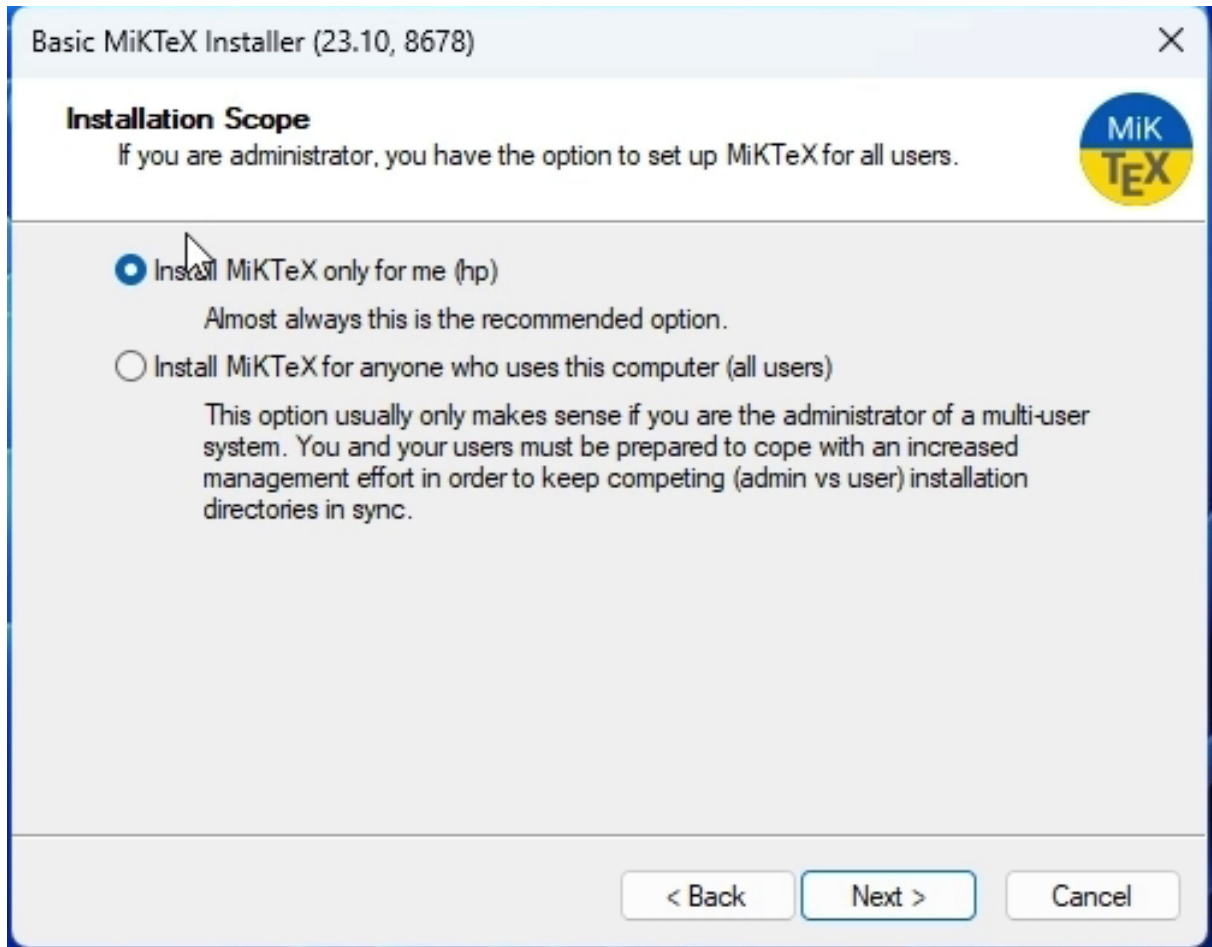


Figure 2.3: Installation Step 2

2. In the MiKTeX Console, go to the “Packages” tab and click “Refresh” to update the package database as shown in Figure 2.6.

1. Start मेनू से MiKTeX Console खोलें। जैसा कि चित्र 2.7 में दर्शाया गया है।
2. MiKTeX Console में "Packages" टैब पर जाएँ और पैकेज डेटाबेस अपडेट करने के लिए "Refresh" पर क्लिक करें। जैसा कि चित्र 2.8 में दर्शाया गया है।
3. अब आप TeXworks जैसे एडिटर का उपयोग करके अपने LaTeX डॉक्यूमेंट बना और कम्पाइल कर सकते हैं।

## 2.3 OR Install “TeX Live” on Windows / या Windows पर TeX Live इंस्टॉल करें:

### Download TeX Live / TeX Live डाउनलोड करें:

1. Visit the TeX Live website: [TeX Live Download Page](#).
2. Download the installer, usually named `install-tl-windows.exe` as shown in Figure 2.9.

1. TeX Live वेबसाइट पर जाएँ।
2. `install-tl-windows.exe` नाम की फ़ाइल डाउनलोड करें। जैसा कि चित्र 2.9 में दर्शाया गया है।

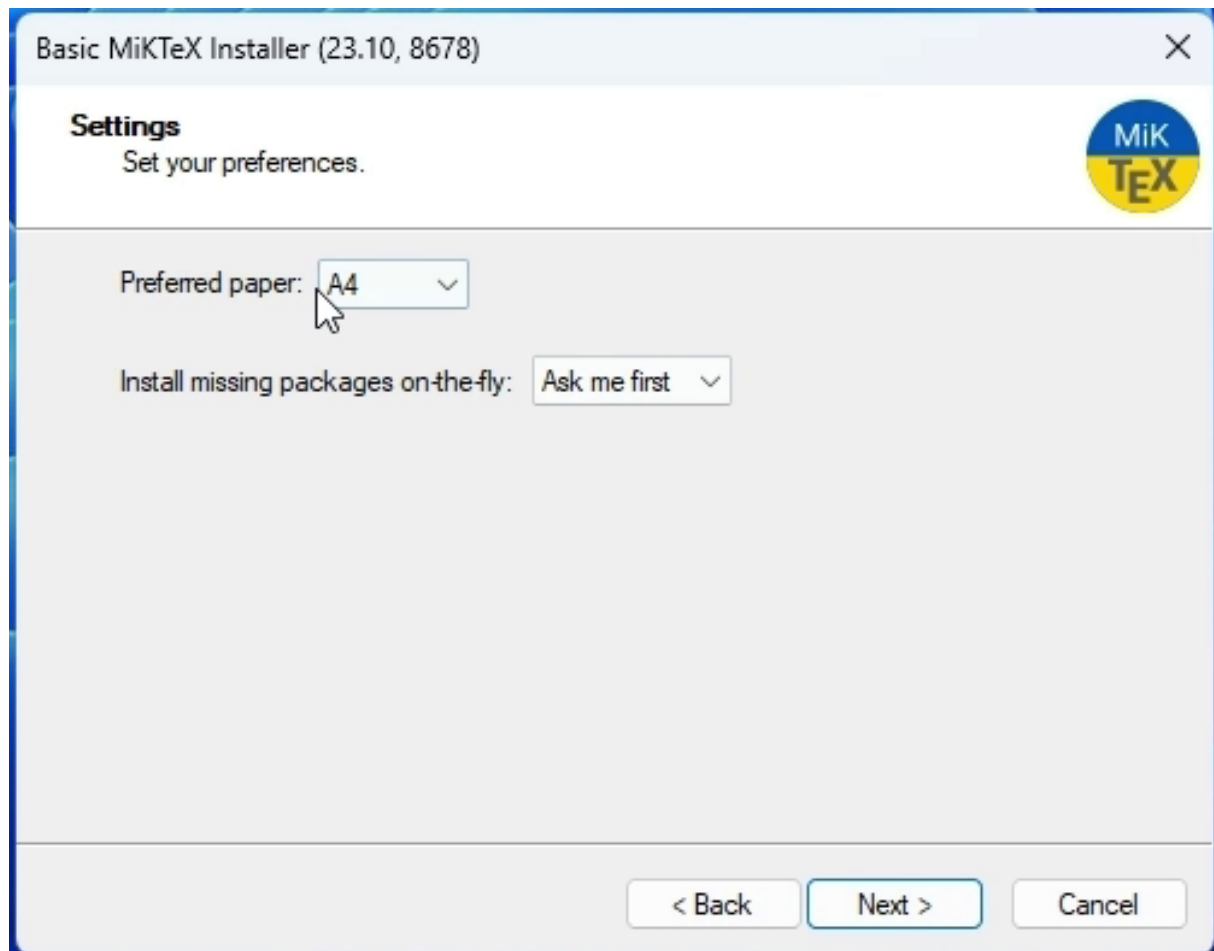


Figure 2.4: Installation Step 3

### 2.3.1 Run the Installer / इंस्टॉलर चलाएँ:

1. Double-click on the downloaded installer.
2. Choose your installation preferences (it's recommended to go with the default options for most users).
3. The installer will connect to the internet and download the necessary files.

1. डाउनलोड की गई फ़ाइल पर डबल क्लिक करें।
2. इंस्टॉलेशन विकल्प चुनें (अधिकांश उपयोगकर्ताओं के लिए डिफ़ॉल्ट सेटिंग उपयुक्त होती है)।
3. इंस्टॉलर इंटरनेट से आवश्यक फ़ाइलें डाउनलोड करेगा।

### 2.3.2 Installation Settings / इंस्टॉलेशन सेटिंग्स:

1. During the installation, you may be prompted to select a server (mirror link). The "Basic" scheme is usually sufficient for most users as shown in Figure 2.10.
2. Select a preferred paper size, and click "Next" as shown in Figure 2.11.
3. Follow the on-screen instructions to complete the installation as shown in Figure 2.12 .

1. इंस्टॉलेशन के दौरान आपसे सर्वर (मिरर लिंक) चुनने के लिए कहा जा सकता है। अधिकांश उपयोगकर्ताओं के लिए "Basic" स्कीम सामान्यतः पर्याप्त होती है, जैसा कि चित्र 2.10 में दर्शाया गया है।

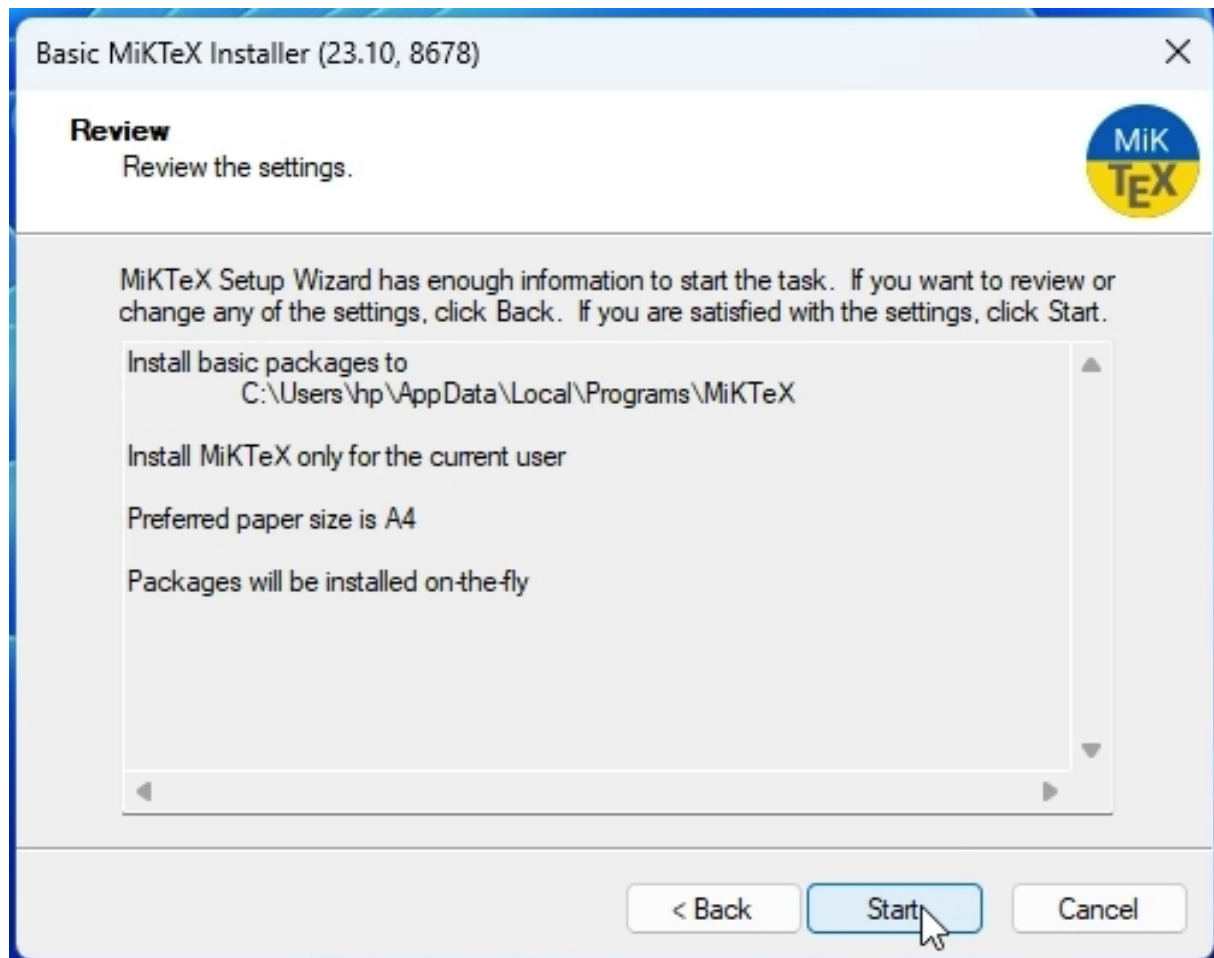


Figure 2.5: Installation Step 4

2. अपनी पसंद के अनुसार पेपर साइज चुनें और "Next" पर क्लिक करें, जैसा कि चित्र 2.11 में दिखाया गया है।
3. स्क्रीन पर दिखाई देने वाले निर्देशों का पालन करते हुए इंस्टॉलेशन प्रक्रिया को पूरा करें, जैसा कि चित्र 2.12 में दर्शाया गया है।

## 2.4 LaTeX Editor / LaTeX एडिटर

Once you have installed a LaTeX system, you will need a LaTeX editor to write and edit your documents efficiently. Here are some popular LaTeX editors to consider:

LaTeX सिस्टम इंस्टॉल करने के बाद, दस्तावेजों को प्रभावी ढंग से लिखने और संपादित करने के लिए एक LaTeX एडिटर की आवश्यकता होती है। कुछ लोकप्रिय एडिटर निम्नलिखित हैं:

1. **TeXworks:** A simple and user-friendly editor bundled with both MiKTeX and TeX Live distributions.
2. **TeXstudio:** A feature-rich, open-source editor with advanced functionality such as syntax highlighting, auto-completion, and an integrated PDF viewer. Available for Windows, macOS, and Linux.
3. **Overleaf:** An online LaTeX editor that allows collaborative writing and editing. Provides real-time preview and is accessible from any device with an internet connection.
4. **Visual Studio Code with LaTeX Workshop Extension:** Utilize Visual Studio Code as a LaTeX editor by installing the "LaTeX Workshop" extension. Offers features like syntax highlighting and PDF preview.

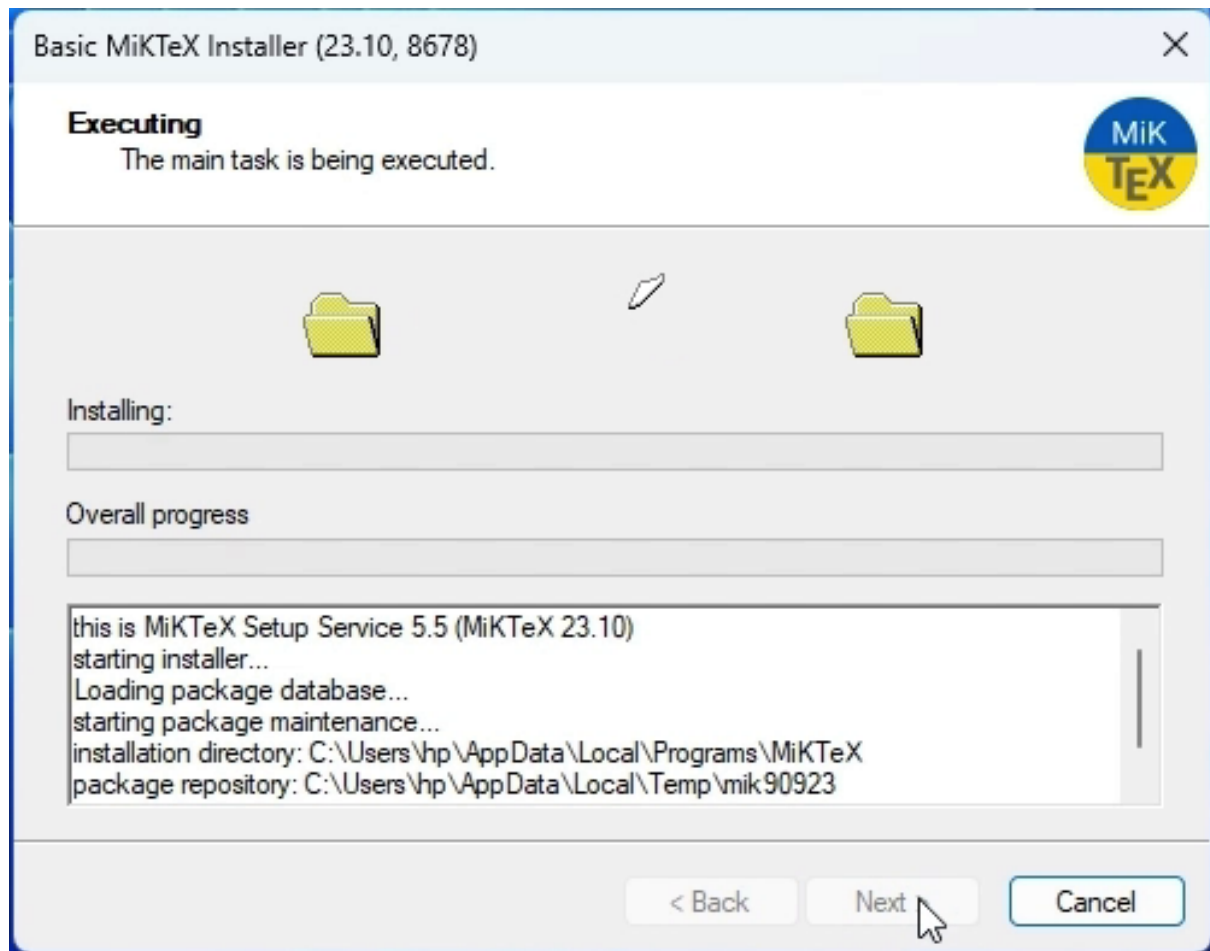


Figure 2.6: Installation Step 5

1. TeXworks: एक सरल और उपयोगकर्ता-अनुकूल एडिटर जो MiKTeX और TeX Live दोनों के साथ उपलब्ध होता है।
2. TeXstudio: एक फीचर-रिच ओपन-सोर्स एडिटर जिसमें सिंटैक्स हाइलाइटिंग, ऑटो-कम्प्लीशन और एकीकृत PDF व्यूअर जैसी उन्नत सुविधाएँ होती हैं। यह Windows, macOS और Linux के लिए उपलब्ध है।
3. Overleaf: एक ऑनलाइन LaTeX एडिटर जो सहयोगात्मक लेखन और संपादन की सुविधा देता है। यह रीयल-टाइम प्रीव्यू प्रदान करता है और इंटरनेट से जुड़ी किसी भी डिवाइस पर उपयोग किया जा सकता है।
4. Visual Studio Code (LaTeX Workshop एक्सटेंशन सहित): LaTeX Workshop एक्सटेंशन इंस्टॉल करके Visual Studio Code को LaTeX एडिटर के रूप में उपयोग किया जा सकता है। इसमें सिंटैक्स हाइलाइटिंग और PDF प्रीव्यू जैसी सुविधाएँ होती हैं।

Choose an editor based on your preferences, platform compatibility, and the level of functionality you require. Many editors offer simplicity for beginners and advanced features for experienced users.

अपनी पसंद, प्लेटफॉर्म की संगतता और आवश्यक सुविधाओं के अनुसार एडिटर चुनें। कई एडिटर नए उपयोगकर्ताओं के लिए सरलता और अनुभवी उपयोगकर्ताओं के लिए उन्नत सुविधाएँ प्रदान करते हैं।

#### 2.4.1 Install “TeXstudio” / TeXstudio इंस्टॉल करें:

TeXstudio is a free and open-source integrated writing environment designed for LaTeX, a typesetting system widely used for producing documents with complex formatting, such as academic papers and scientific articles.

TeXstudio एक निःशुल्क और ओपन-सोर्स लेखन वातावरण है जिसे LaTeX के लिए डिज़ाइन किया गया है। LaTeX व्यापक रूप से अकादमिक पेपर और वैज्ञानिक लेखों जैसे जटिल डॉक्यूमेंट तैयार करने में उपयोग किया जाता है।

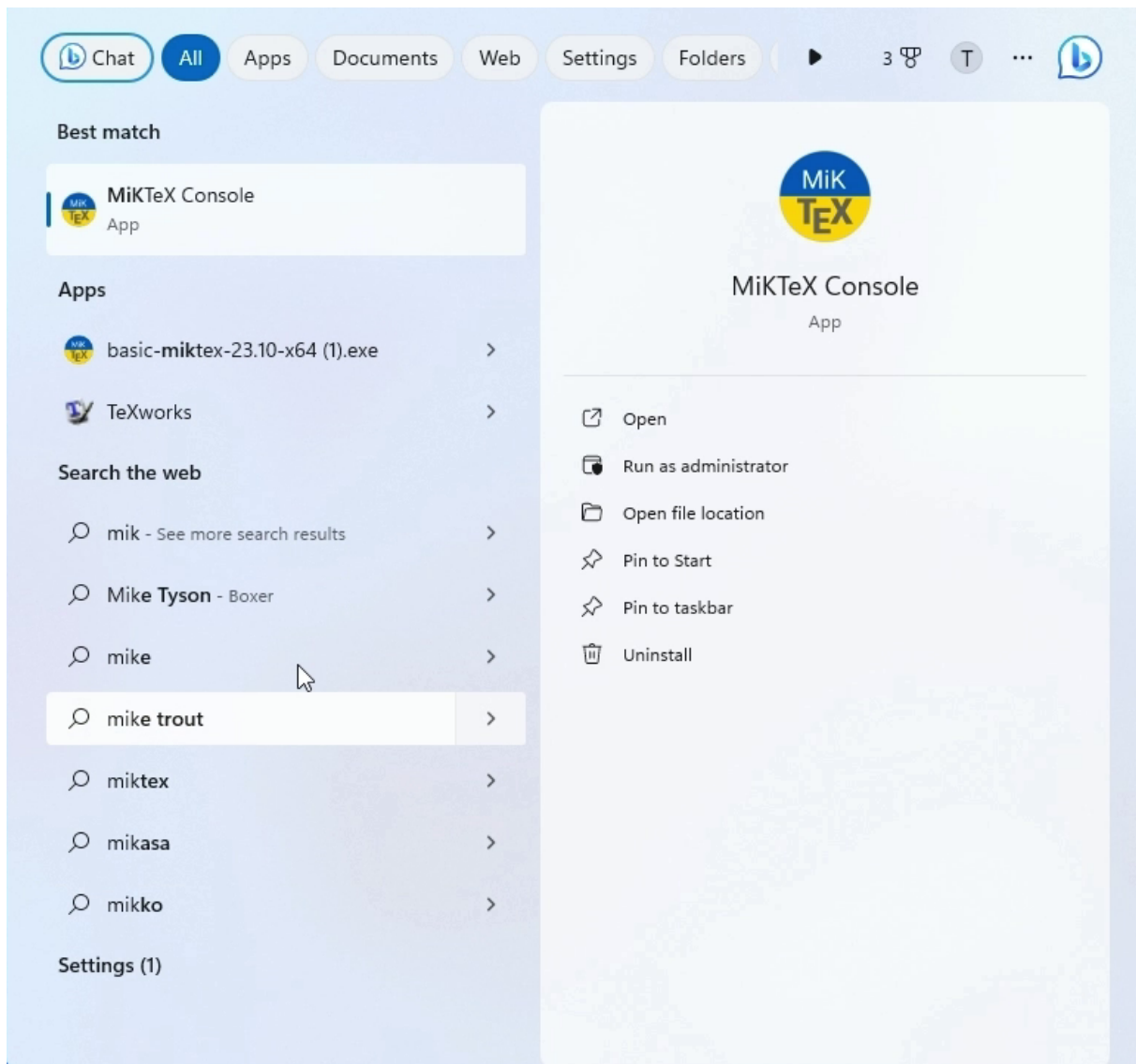


Figure 2.7: Installation Step 6

1. Download TeXstudio from the [TeXstudio website](#) as shown in Figure 2.13.
2. Install TeXstudio by running the downloaded installer.

1. [TeXstudio वेबसाइट](#) से TeXstudio डाउनलोड करें। जैसा कि चित्र 2.13 में दर्शाया गया है।
2. डाउनलोड की गई फ़ाइल चलाकर TeXstudio इंस्टॉल करें।

Now, you have a working LaTeX setup on your Windows system.

अब आपके Windows सिस्टम पर LaTeX पूरी तरह कार्य करने के लिए तैयार है।

## 2.5 Installing TeX Live on Linux/ Linux पर TeX Live इंस्टॉल करना

The easiest way to install TeX Live on Linux is by using the system package manager. This method is recommended for beginners because it requires only a few simple commands and does not need any manual configuration.

Linux पर TeX Live इंस्टॉल करने का सबसे आसान तरीका सिस्टम पैकेज मैनेजर का उपयोग करना है। यह विधि शुरुआती उपयोगकर्ताओं के लिए उपयुक्त है क्योंकि इसमें केवल कुछ सरल कमांड्स की आवश्यकता होती है और किसी भी मैनुअल कॉन्फिगरेशन की जरूरत नहीं पड़ती।

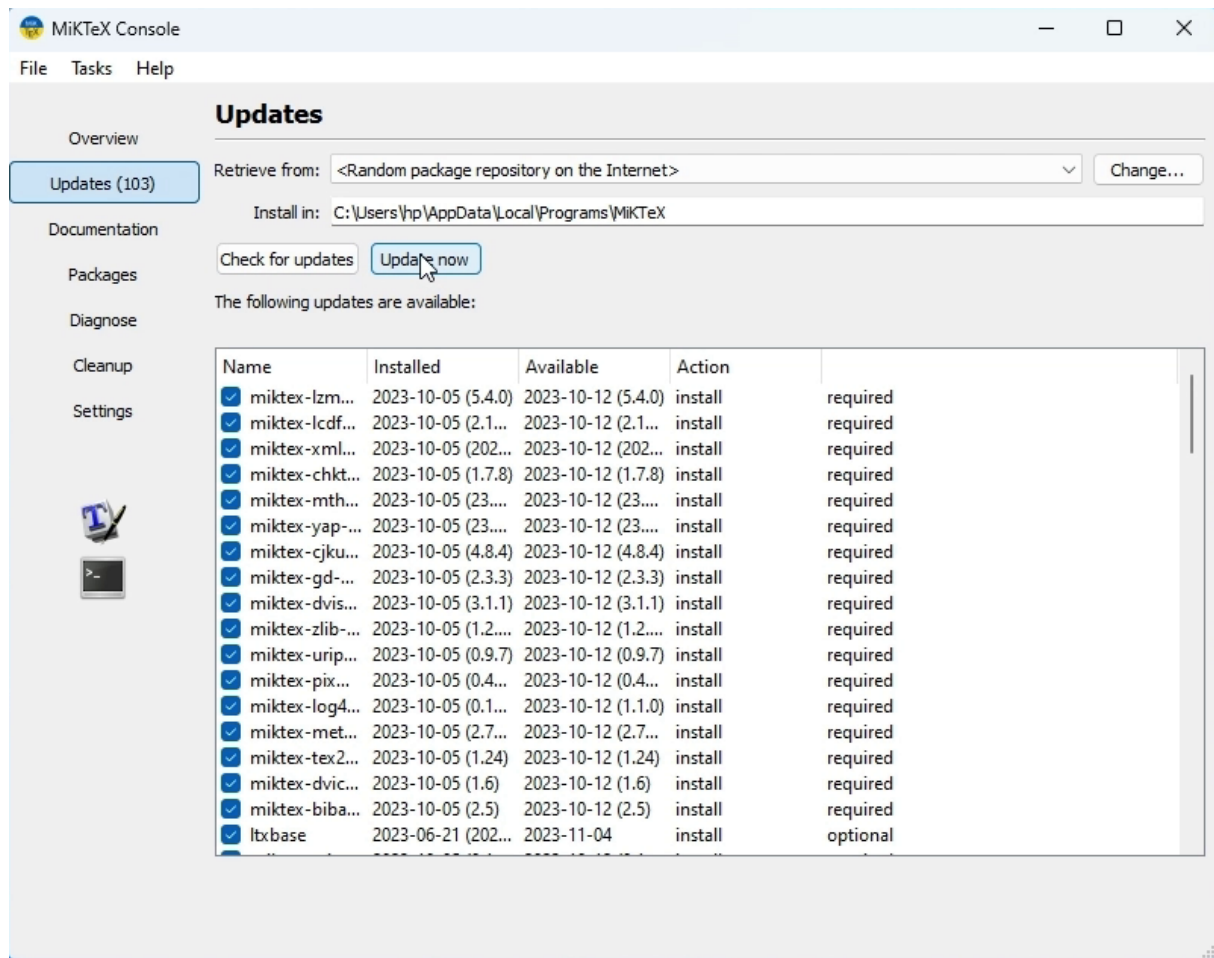


Figure 2.8: Installation Step 7

### 2.5.1 Step 1: Open the Terminal / चरण 1: टर्मिनल खोलें

Open the terminal window by pressing **Ctrl + Alt + T**.  
Ctrl + Alt + T कुंजियाँ दबाकर टर्मिनल विंडो खोलें।

### 2.5.2 Step 2: Update the System / चरण 2: सिस्टम अपडेट करें

Update the package list using the following command:

```
sudo apt update
```

निम्नलिखित कमांड का उपयोग करके पैकेज सूची को अपडेट करें:

### 2.5.3 Step 3: Install TeX Live / चरण 3: TeX Live इंस्टॉल करें

Install the complete TeX Live distribution using:

```
sudo apt install texlive-full
```

This installs all commonly used LaTeX packages and helps avoid missing-package errors during document compilation.

यह कमांड पूर्ण TeX Live डिस्ट्रीब्यूशन इंस्टॉल करती है, जिसमें अधिकांश आवश्यक LaTeX पैकेज शामिल होते हैं और डॉक्यूमेंट को कंपाइल करते समय पैकेज न मिलने की समस्या से बचाव होता है।

### 2.5.4 Step 4: Verify the Installation / चरण 4: इंस्टॉलेशन सत्यापित करें

After installation, verify that TeX Live is installed correctly by running:

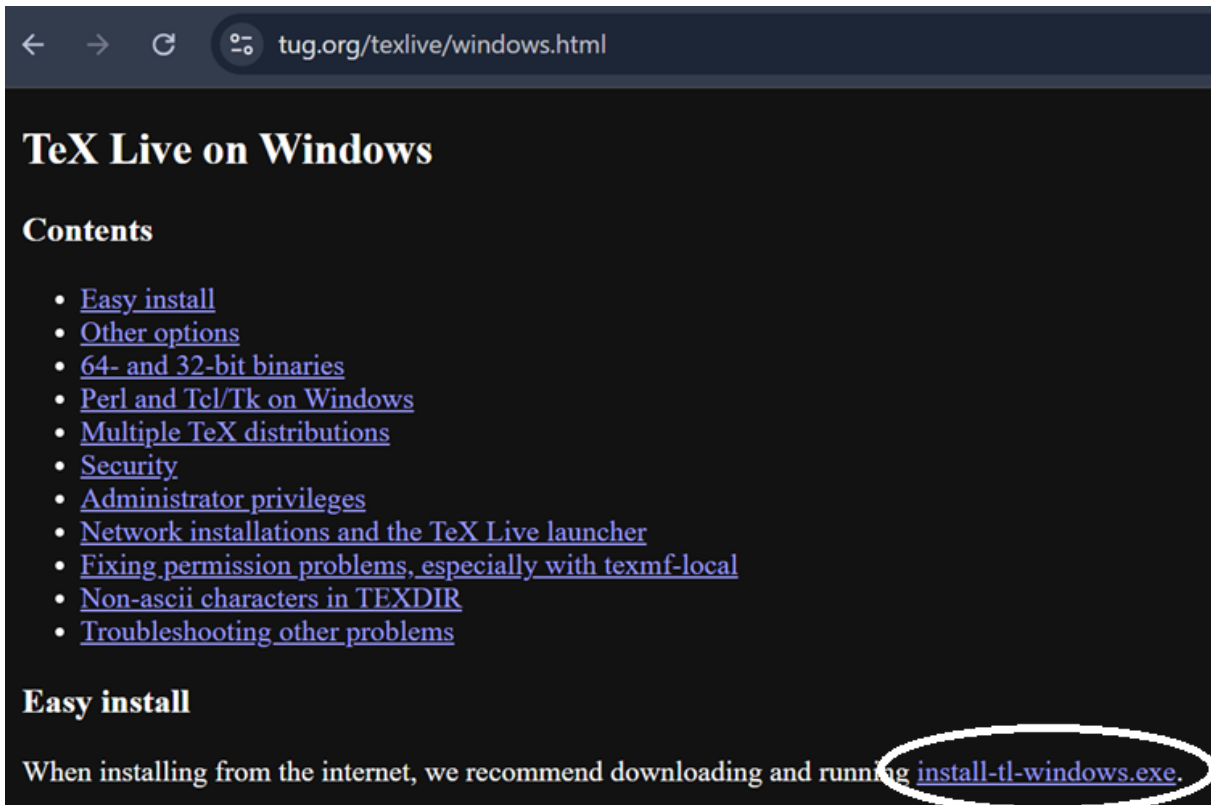


Figure 2.9: Texlive Installation Step 1

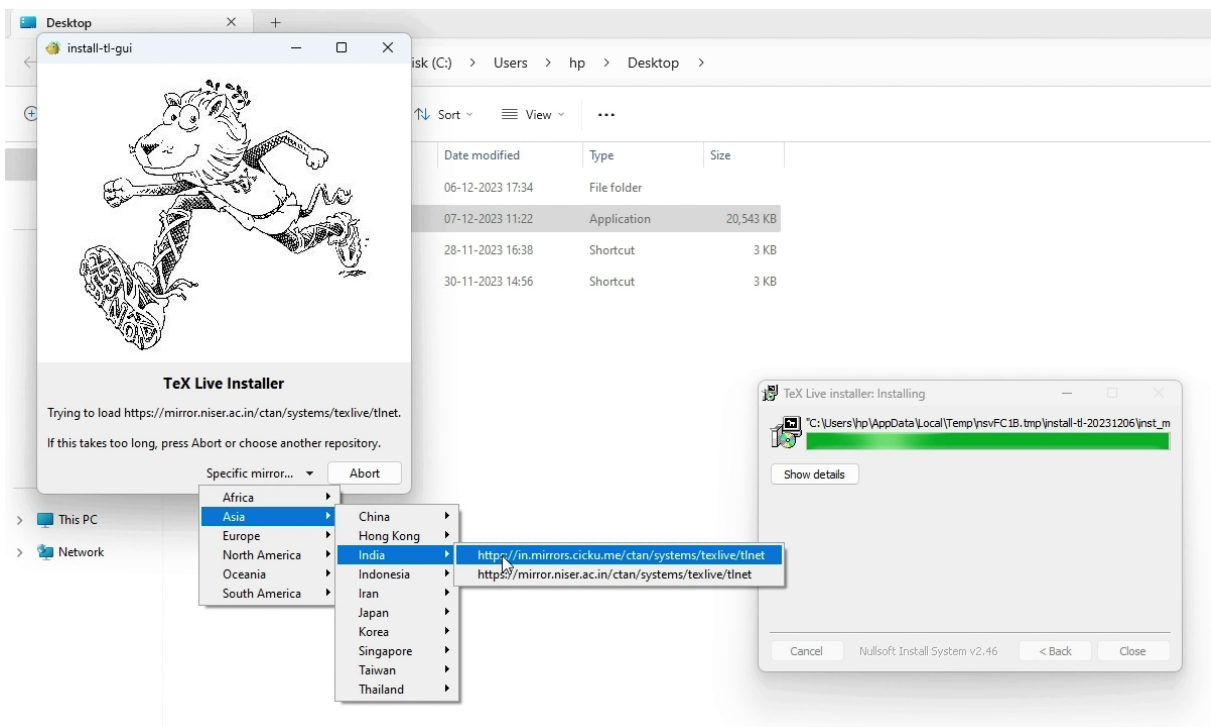


Figure 2.10: Texlive Installation Step 2

```
pdflatex --version
```

If version information is displayed, the installation is successful.

इंस्टॉलेशन के बाद यह जाँचने के लिए कि TeX Live सही प्रकार से इंस्टॉल हुआ है या नहीं, ऊपर दिया गया कमांड चलाएँ।

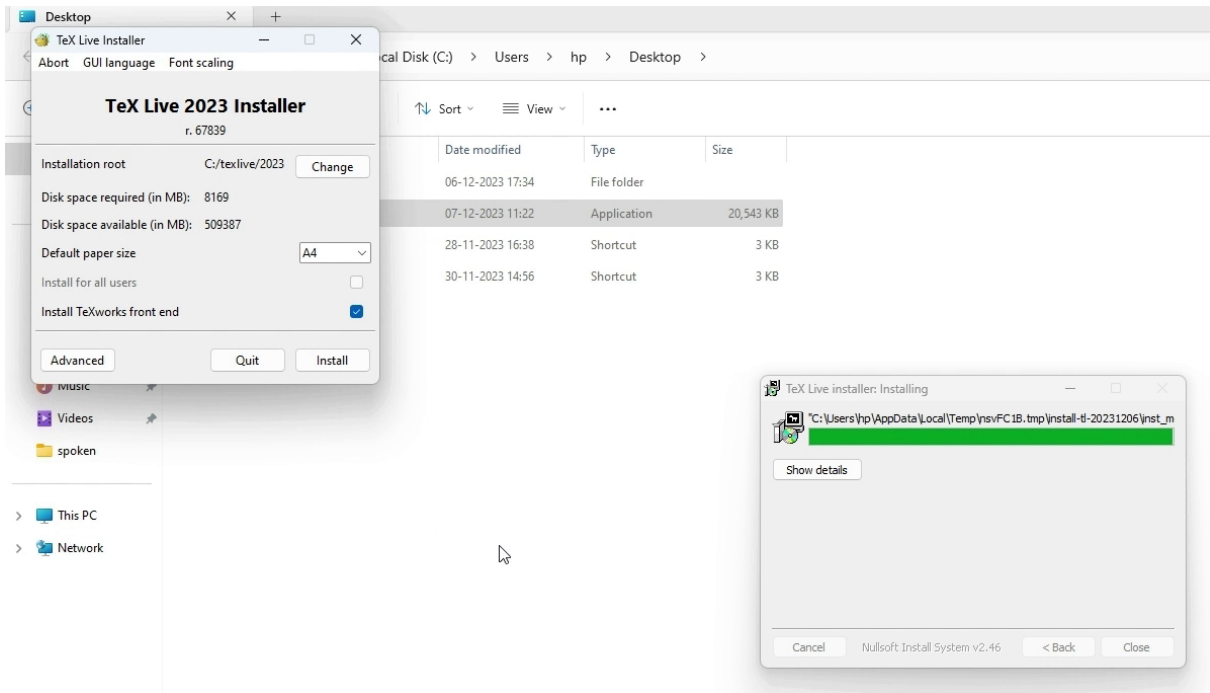


Figure 2.11: TeXlive Installation Step 3

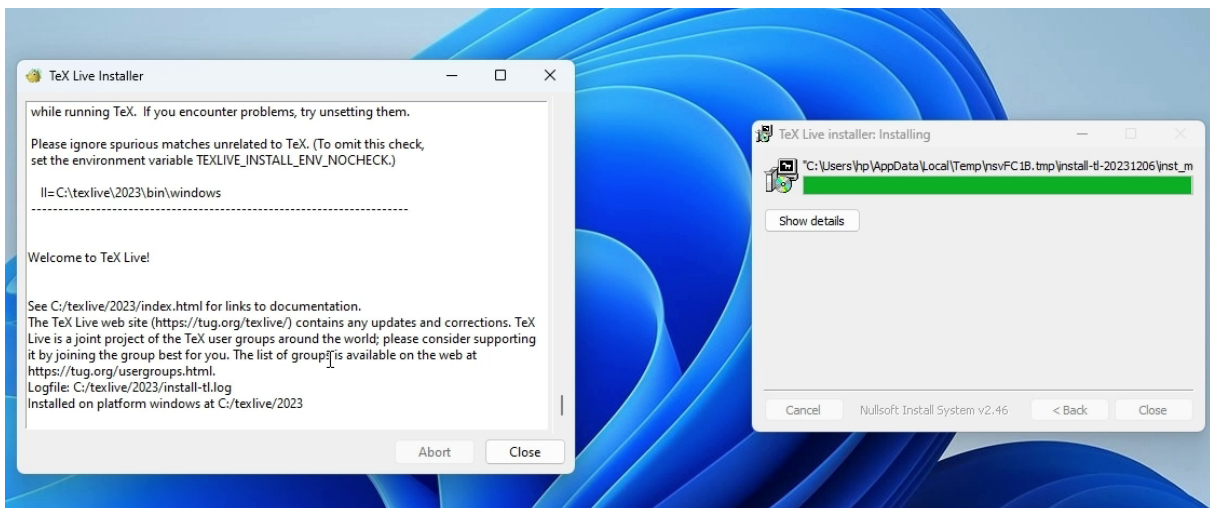


Figure 2.12: TeXlive Installation Step 4

यदि संस्करण संबंधी जानकारी दिखाई दे, तो इंस्टॉलेशन सफल है।

### 2.5.5 Step 5: Install TeXstudio Editor (Optional) / चरण 5: TeXstudio एडिटर इंस्टॉल करें (वैकल्पिक)

To write and compile LaTeX documents easily, install the TeXstudio editor using:

```
1 sudo apt install texstudio
```

TeXstudio automatically detects the installed TeX Live distribution.

LaTeX डॉक्यूमेंट आसानी से लिखने और कंपाइल करने के लिए TeXstudio एडिटर इंस्टॉल करें। TeXstudio स्वतः इंस्टॉल की गई TeX Live डिस्ट्रीब्यूशन को पहचान लेता है।

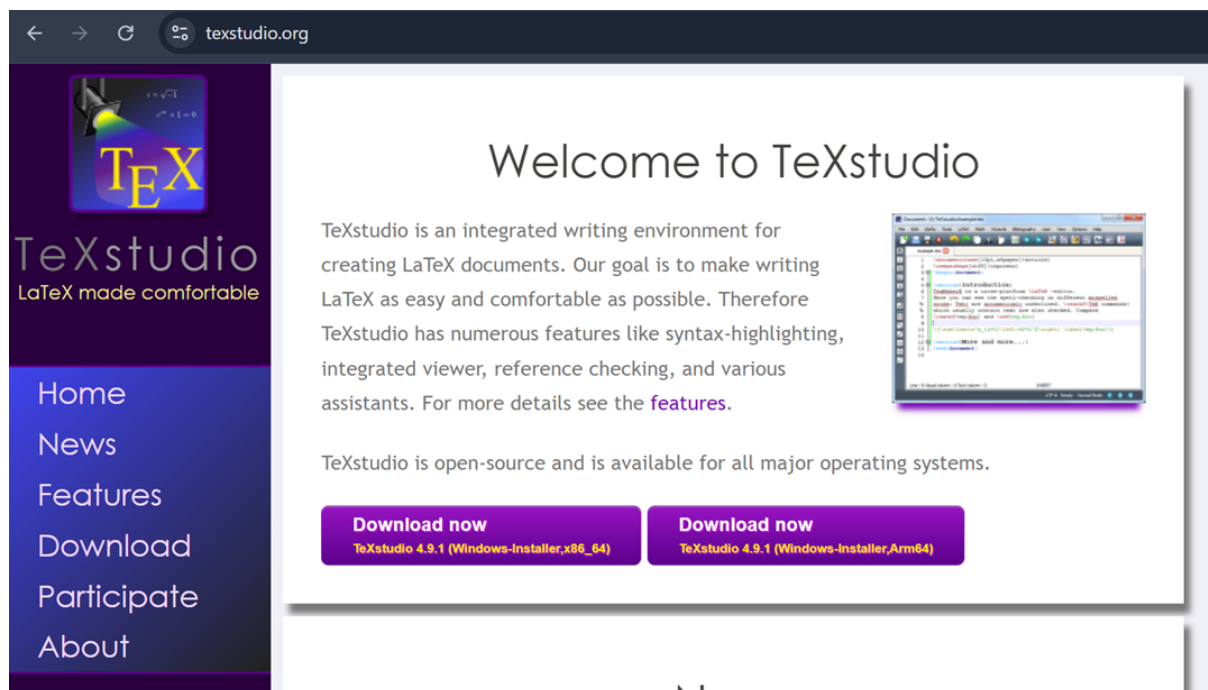


Figure 2.13: Texlive Installation Step 5

## 2.6 Writing and Compiling Your First LaTeX Document Using TeXstudio / TeXstudio का उपयोग करके पहला LaTeX डॉक्यूमेंट लिखना और कंपाइल करना

After successfully installing a LaTeX distribution (MiKTeX or TeX Live) and the TeXstudio editor, you are now ready to create, compile, and view your first LaTeX document. TeXstudio provides an integrated environment where you can write LaTeX source code, compile it, and preview the output PDF in a single interface.

MiKTeX या TeX Live जैसी LaTeX डिस्ट्रीब्यूशन तथा TeXstudio एडिटर को सफलतापूर्वक इंस्टॉल करने के बाद, अब आप अपना पहला LaTeX डॉक्यूमेंट बनाने, उसे कंपाइल करने और PDF आउटपुट देखने के लिए तैयार हैं। TeXstudio एक एकीकृत वातावरण प्रदान करता है, जिसमें LaTeX स्रोत कोड लिखना, उसे कंपाइल करना और आउटपुट PDF को एक ही इंटरफ़ेस में देखना संभव होता है।

### 2.6.1 Step 1: Launch TeXstudio / चरण 1: TeXstudio प्रारंभ करें

Open **TeXstudio** from the Start Menu or the desktop shortcut. When TeXstudio starts, the main interface appears, consisting of the editor window, menu bar, toolbar, and an integrated PDF viewer, as shown in Figure 2.14.

Start Menu या डेस्कटॉप शॉर्टकट से TeXstudio खोलें। TeXstudio प्रारंभ होने पर इसका मुख्य इंटरफ़ेस दिखाई देता है, जिसमें एडिटर विंडो, मेनू बार, टूलबार तथा एक इंटीग्रेटेड PDF व्यूअर शामिल होता है, जैसा कि Figure 2.14 में दर्शाया गया है।

### 2.6.2 Step 2: Create a New LaTeX File / चरण 2: नई LaTeX फ़ाइल बनाएँ

To create a new LaTeX document, click **File** → **New** or press **Ctrl + N**. A blank editor window will open where you can start writing LaTeX code, as shown in Figure 2.15.

नई LaTeX फ़ाइल बनाने के लिए File → New पर क्लिक करें या Ctrl + N दबाएँ। इसके बाद एक खाली एडिटर विंडो खुलेगी, जिसमें आप LaTeX कोड लिखना शुरू कर सकते हैं, जैसा कि Figure 2.15 में दर्शाया गया है।

### 2.6.3 Step 3: Write a Basic LaTeX Program / चरण 3: एक सरल LaTeX प्रोग्राम लिखें

Type the following basic LaTeX program in the editor. This represents the minimum structure of a LaTeX document, as shown in Figure 2.16.

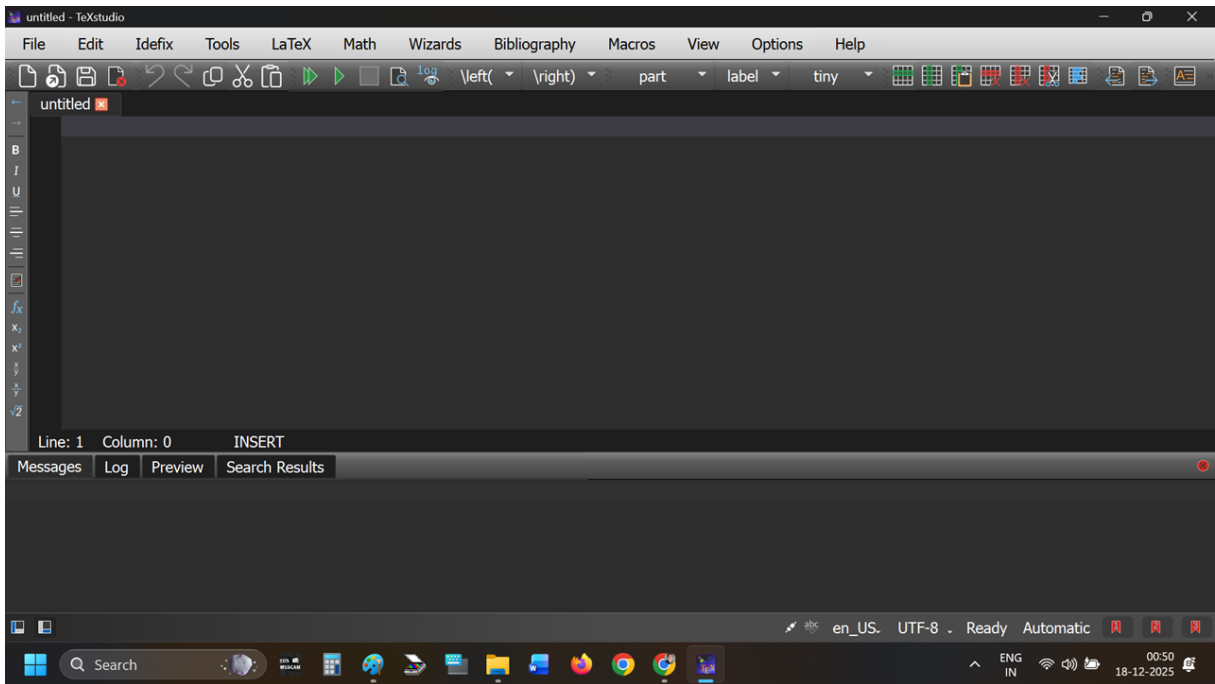


Figure 2.14: TeXstudio main interface

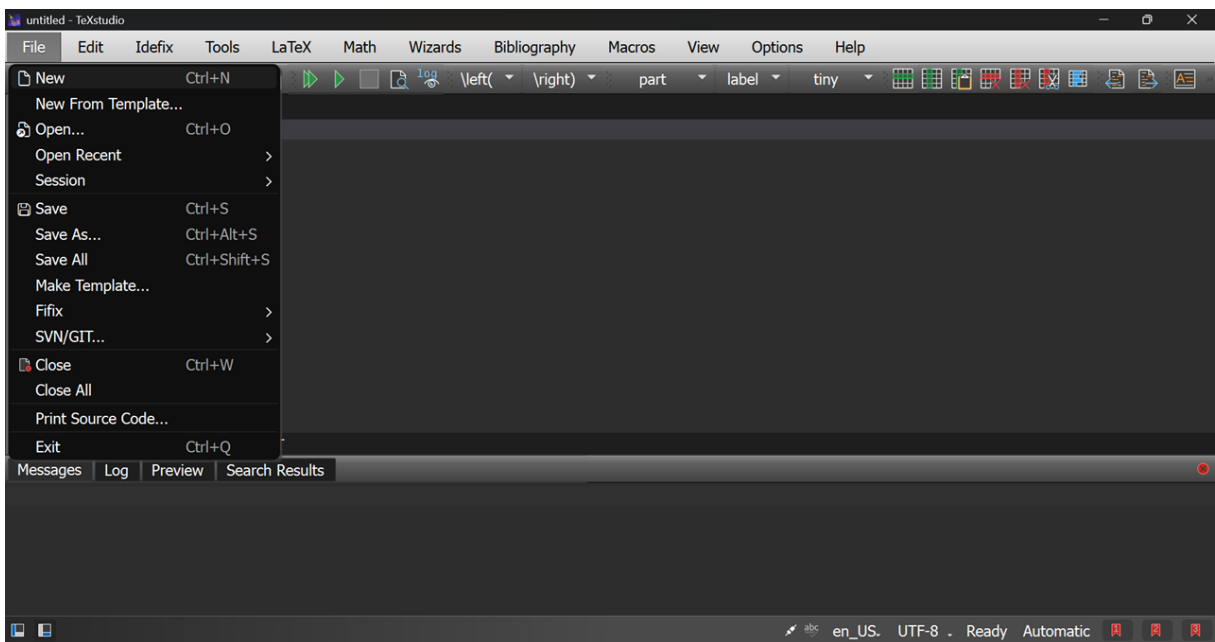


Figure 2.15: Creating a new LaTeX file in TeXstudio

एडिटर में निम्नलिखित सरल LaTeX प्रोग्राम टाइप करें। यह LaTeX डॉक्यूमेंट की न्यूनतम संरचना को दर्शाता है, जैसा कि Figure 2.16 में दिखाया गया है।

```

1 \documentclass{article}
2
3 \begin{document}
4
5 Hello, LaTeX!
6
7 This is my first LaTeX document.
8
9 \end{document}

```

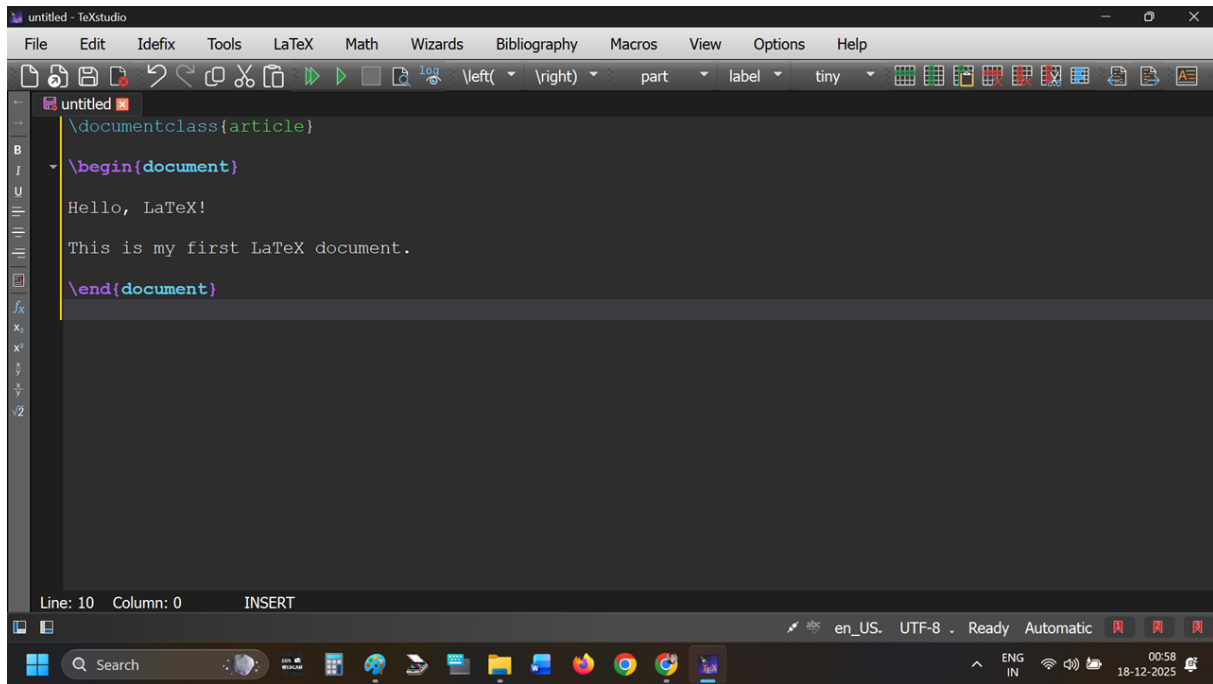
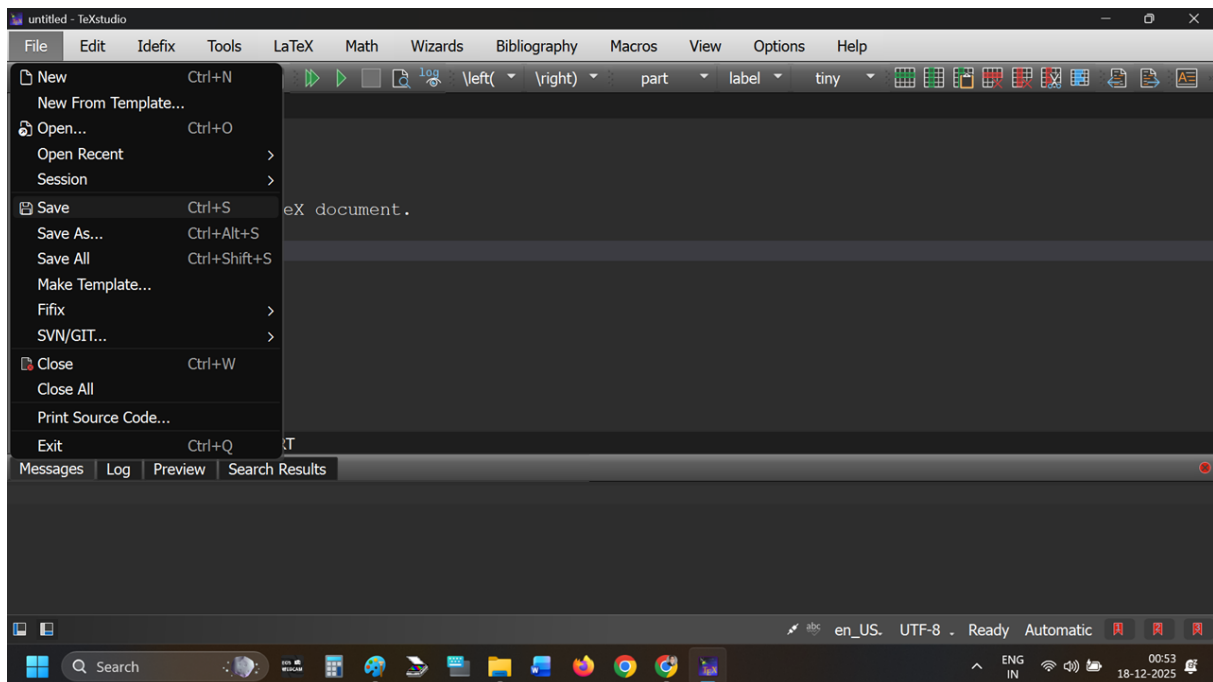


Figure 2.16: LaTeX source code written in TeXstudio

#### 2.6.4 Step 4: Save the File / चरण 4: फ़ाइल को सेव करें

Before compiling the document, save the file by clicking **File** → **Save** or pressing **Ctrl + S**. Choose a suitable folder and save the file with a `.tex` extension, for example `first_document.tex`, as shown in Figure 2.17.

डॉक्यूमेंट को कंपाइल करने से पहले फ़ाइल को सेव करना आवश्यक है। इसके लिए File → Save पर क्लिक करें या Ctrl + S दबाएँ। उपयुक्त फ़ोल्डर चुनें और फ़ाइल को `.tex` एक्सटेंशन के साथ सेव करें, जैसे `first_document.tex`, जैसा कि Figure 2.17 में दर्शाया गया है।

Figure 2.17: Saving a LaTeX file with `.tex` extension

### 2.6.5 Step 5: Compile the Document / चरण 5: डॉक्यूमेंट को कंपाइल करें

To compile the LaTeX document and generate a PDF output, click the green **Compile** or **Build & View** button in the toolbar, or press **F5**, as shown in Figure 2.18. TeXstudio uses the installed LaTeX distribution (MiKTeX or TeX Live) to process the source file.

LaTeX डॉक्यूमेंट को कंपाइल करके PDF आउटपुट प्राप्त करने के लिए टूलबार में मौजूद हरे रंग के Compile या Build & View बटन पर क्लिक करें अथवा F5 दबाएँ, जैसा कि Figure 2.18 में दर्शाया गया है। इस प्रक्रिया में TeXstudio इंस्टॉल की गई LaTeX डिस्ट्रीब्यूशन (MiKTeX या TeX Live) का उपयोग करके स्रोत फ़ाइल को प्रोसेस करता है।

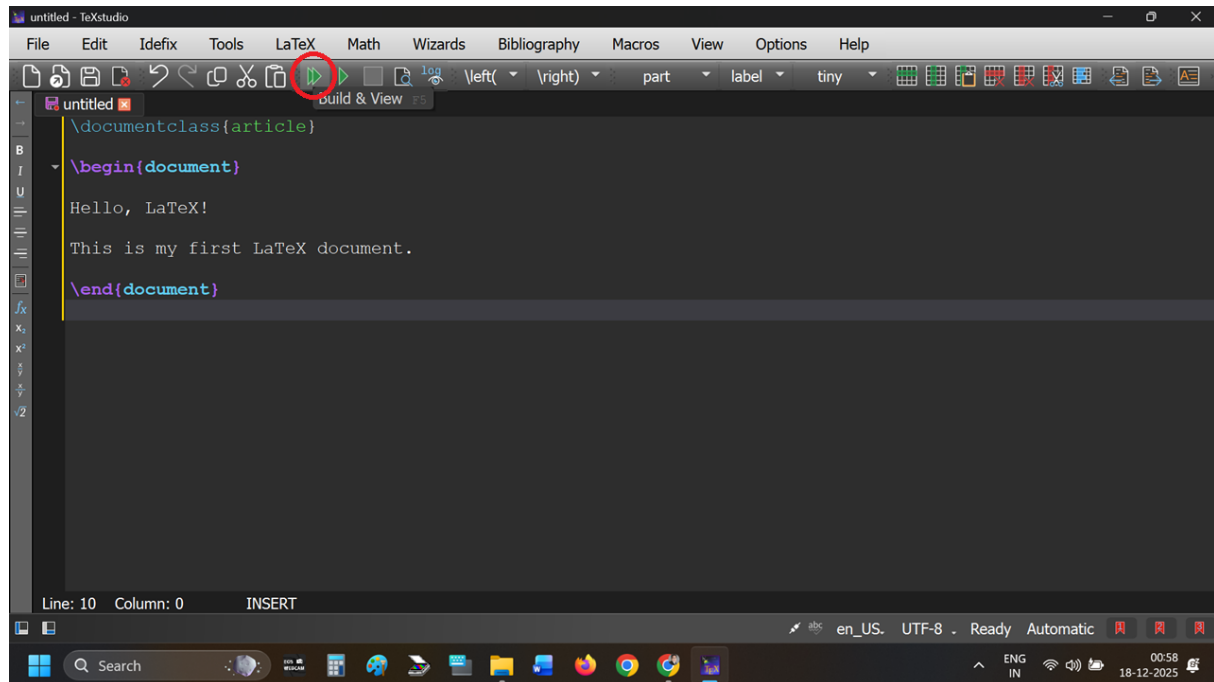


Figure 2.18: Compiling the LaTeX document in TeXstudio

### 2.6.6 Step 6: View the Output PDF / चरण 6: PDF आउटपुट देखें

If the compilation is successful, the output PDF is displayed automatically in the integrated PDF viewer panel, as shown in Figure 2.19.

यदि कंपाइलेशन सफल रहता है, तो आउटपुट PDF स्वतः TeXstudio के इंटीग्रेटेड PDF व्यूअर पैनल में दिखाई देता है, जैसा कि Figure 2.19 में दर्शाया गया है।

### 2.6.7 Step 7: Understanding Compile Messages / चरण 7: कंपाइलेशन संदेशों को समझना

During compilation, messages related to errors or warnings are shown in the log panel at the bottom of the TeXstudio window, as shown in Figure 2.20. Errors usually occur due to missing commands, typing mistakes, or incorrect syntax.

कंपाइल प्रक्रिया के दौरान यदि कोई त्रुटि (Error) या चेतावनी (Warning) होती है, तो उससे संबंधित संदेश TeXstudio विंडो के निचले भाग में स्थित लॉग पैनल में दिखाई देते हैं, जैसा कि Figure 2.20 में दर्शाया गया है। सामान्यतः ये त्रुटियाँ कमांड की कमी, टाइपिंग की गलती या गलत सिंटैक्स के कारण होती हैं।

### 2.6.8 Step 8: Modify and Recompile / चरण 8: संशोधन करें और पुनः कंपाइल करें

You can edit the document content, save the changes, and recompile the file. Each compilation updates the PDF output automatically.

आप डॉक्यूमेंट की सामग्री में संशोधन कर सकते हैं, किए गए परिवर्तनों को सेव कर सकते हैं और फ़ाइल को पुनः कंपाइल कर सकते हैं। प्रत्येक बार कंपाइल करने पर PDF आउटपुट स्वतः अपडेट हो जाता है।

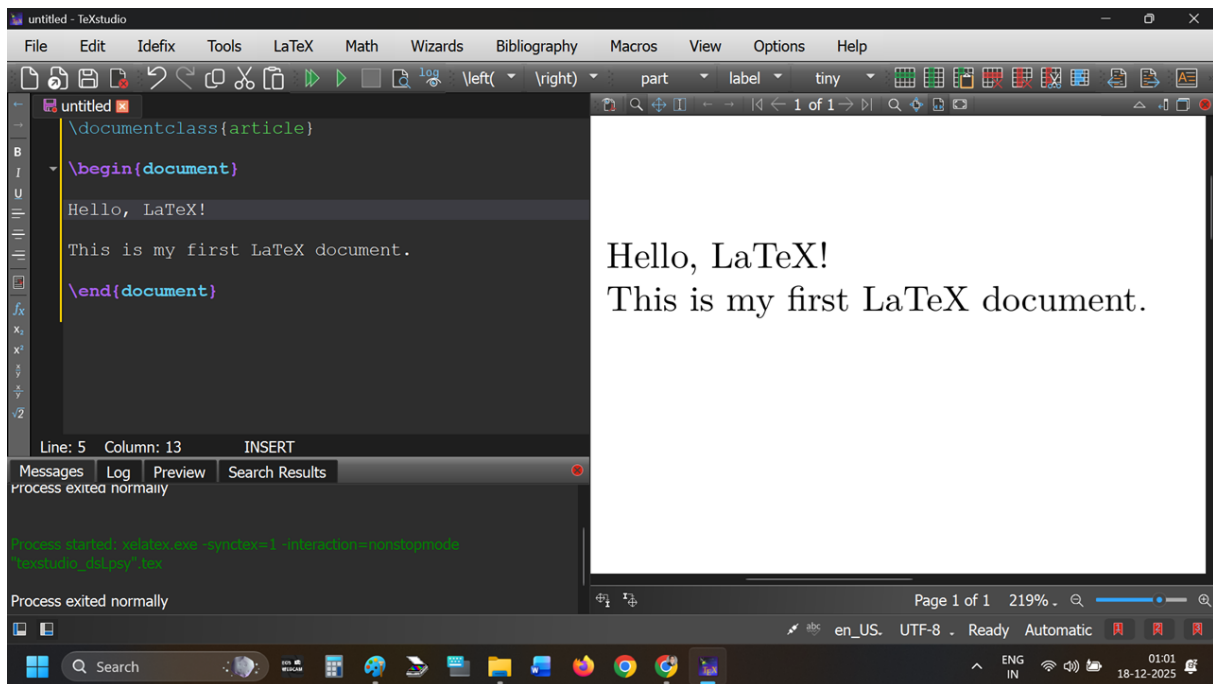


Figure 2.19: PDF output displayed in TeXstudio

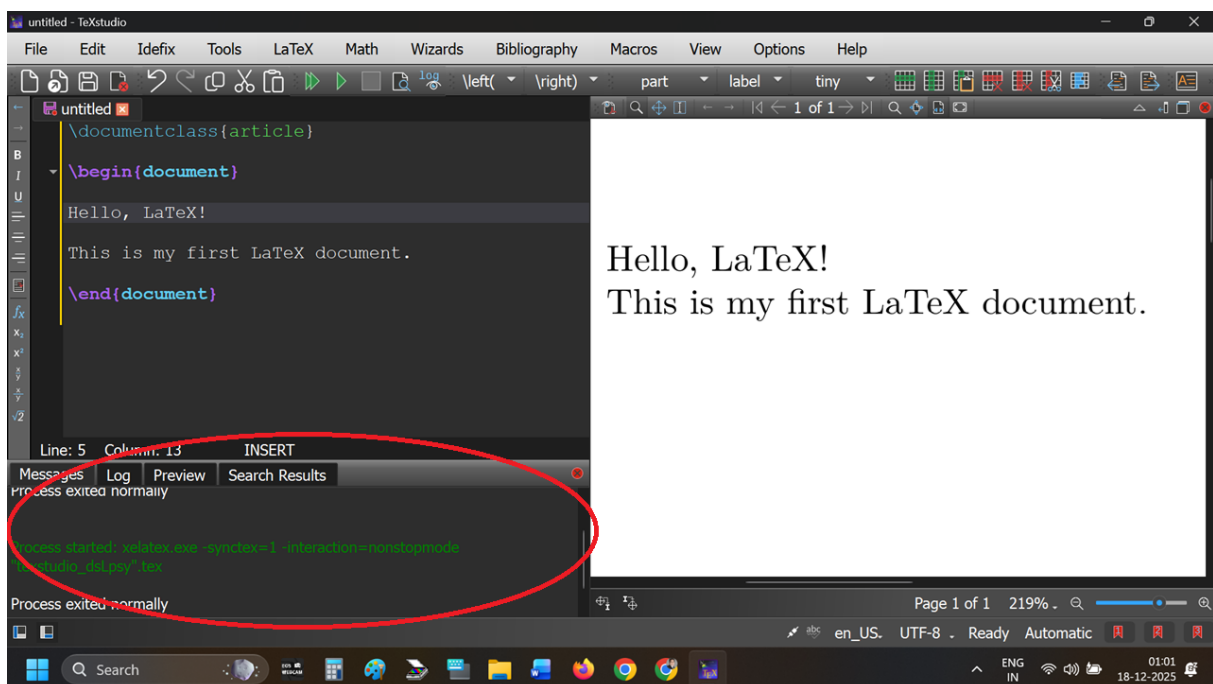


Figure 2.20: Compile messages and log panel

### 2.6.9 Workflow Summary / कार्यप्रवाह का सारांश

The complete workflow for creating documents using LaTeX and TeXstudio is summarized below:

1. Write LaTeX source code in TeXstudio
2. Save the file with a .tex extension
3. Compile the document
4. View the output as a PDF

LaTeX और TeXstudio का उपयोग करके डॉक्यूमेंट बनाने की पूरी कार्यप्रणाली नीचे संक्षेप में दी गई है:

1. TeXstudio में LaTeX स्रोत कोड लिखना
2. फ़ाइल को .tex एक्सटेंशन के साथ सेव करना
3. डॉक्यूमेंट को कंपाइल करना
4. आउटपुट को PDF के रूप में देखना

# 3. Basic Document Structure / मूल डॉक्यूमेंट संरचना

In LaTeX, the basic document structure consists of several key components: LaTeX में, डॉक्यूमेंट की मूल संरचना कई महत्वपूर्ण घटकों से मिलकर बनती है।

## 3.1 LaTeX Tutorial: Document Class / LaTeX ट्यूटोरियल: डॉक्यूमेंट क्लास

Welcome to our LaTeX tutorial series. In this tutorial, we'll discuss the **document class** in LaTeX. हमारी LaTeX ट्यूटोरियल श्रृंखला में आपका स्वागत है। इस अध्याय में हम LaTeX की डॉक्यूमेंट क्लास के बारे में चर्चा करेंगे।

### 3.1.1 What is a Document Class? / डॉक्यूमेंट क्लास क्या होती है?

A document class in LaTeX is a predefined template that determines the overall layout and formatting style of your document. It sets the basic structure and formatting rules, including margins, fonts, headers/footers, and sectioning commands.

LaTeX में डॉक्यूमेंट क्लास एक पूर्वनिर्धारित टेम्पलेट होती है जो डॉक्यूमेंट का लेआउट और फॉर्मेटिंग शैली निर्धारित करती है। इसमें मार्जिन, फ्रॉन्ट, हेडर/फूटर और सेक्शन कमांड जैसे नियम शामिल होते हैं।

### 3.1.2 Common Document Classes / सामान्य डॉक्यूमेंट क्लासेस

Here are some common document classes in LaTeX:  
LaTeX में प्रचलित कुछ डॉक्यूमेंट क्लासेस निम्नलिखित हैं:

- `article`: Suitable for shorter documents like articles, reports, and papers.
- `report`: Suitable for longer documents with chapters, such as reports and dissertations.
- `book`: Suitable for books and longer documents divided into chapters and parts.
- `letter`: Specifically designed for writing letters.
- `article`: छोटे डॉक्यूमेंट जैसे आलेख, रिपोर्ट और पेपर के लिए उपयुक्त।
- `report`: लंबे डॉक्यूमेंट जिनमें अध्याय होते हैं, जैसे रिपोर्ट व शोध कार्य।
- `book`: पुस्तकों और अध्यायों में विभाजित बड़े डॉक्यूमेंट के लिए।
- `letter`: पत्र लेखन के लिए विशेष रूप से।

### 3.1.3 Using a Document Class / डॉक्यूमेंट क्लास का उपयोग

To use a document class, simply include the following command at the beginning of your LaTeX document: डॉक्यूमेंट क्लास का उपयोग करने के लिए इसे LaTeX फ़ाइल की शुरुआत में लिखें:

```
\documentclass{<class>}
```

Replace `<class>` with the desired document class name, such as `article`, `report`, etc. `<class>` को इच्छित क्लास नाम से बदलें जैसे `article`, `report` आदि।

## Example / उदाहरण

Let's see an example of using the `article` document class:

आइए `article` डॉक्यूमेंट क्लास का उदाहरण देखें:

```
\documentclass{article}

\begin{document}

% Your content goes here...

\end{document}
```

In this example, we've declared the document class as `article`. You can now start writing your document content between `\begin{document}` and `\end{document}`.

इस उदाहरण में डॉक्यूमेंट क्लास `article` घोषित की गई है। अब आप सामग्री को `\begin{document}` और `\end{document}` के बीच लिख सकते हैं।

## 3.2 LaTeX Tutorial: Preamble / LaTeX ट्यूटोरियल: प्रीएम्बल

### 3.2.1 What is Preamble? / प्रीएम्बल क्या है?

The preamble in LaTeX is the section between `\documentclass{}` and `\begin{document}`. It contains global formatting instructions, settings, and commands that apply to the entire document.

LaTeX में प्रीएम्बल वह भाग होता है जो `\documentclass{}` और `\begin{document}` के बीच लिखा जाता है। इसमें पूरे डॉक्यूमेंट पर लागू होने वाली सेटिंग्स होती हैं।

### 3.2.2 Common Preamble Commands / सामान्य प्रीएम्बल कमांड्स

Here are some common commands and settings found in the preamble:

प्रीएम्बल में सामान्यतः निम्नलिखित कमांड प्रयोग की जाती हैं:

- `\usepackage{}`: Loads additional packages to extend LaTeX's functionality.
  - `\title{}`, `\author{}`, `\date{}`: Defines document metadata such as title, author, and date.
  - `\setlength{length}{value}`: Sets the length of various document elements such as margins, indentation, and line spacing.
  - `\newcommand{command}{}`: Defines new commands for convenience and reusability.
  - `\renewcommand{command}{}`: Modifies existing commands to customize their behavior.
- 
- `\usepackage{}`: अतिरिक्त पैकेज लोड करता है।
  - `\title{}`, `\author{}`, `\date{}`: शीर्षक, लेखक और तारीख निर्धारित करता है।
  - `\setlength{length}{value}`: मार्जिन, स्पेसिंग आदि नियंत्रित करता है।
  - `\newcommand{command}{}`: नए कमांड बनाता है।
  - `\renewcommand{command}{}`: मौजूदा कमांड संशोधित करता है।

## Example / उदाहरण

Let's see an example of a preamble:

प्रीएम्बल का एक उदाहरण देखें:

```

\documentclass{article}
\usepackage{graphicx}
\usepackage{lipsum}

\title{My Document}
\author{John Doe}
\date{\today}

\setlength{\parindent}{0pt}

\begin{document}

```

In this example, we've included the `graphicx` and `lipsum` packages, defined the document title, author, and date, and set the paragraph indentation to zero.

इस उदाहरण में `graphicx` और `lipsum` पैकेज जोड़े गए हैं, शीर्षक, लेखक और तारीख निर्धारित की गई है तथा पैराग्राफ इंडेंटेशन शून्य किया गया है।

### 3.2.3 Title, Author, and Date / शीर्षक, लेखक और तारीख

You can specify the title, author, and date of your document using the following commands in the preamble:

डॉक्यूमेंट का शीर्षक, लेखक और तारीख निम्न प्रकार से निर्धारित की जाती है:

```

\title{Your Document Title}
\author{Your Name}
\date{\today}

```

Replace `Your Document Title` with the title of your document, `Your Name` with your name or the author's name, and `\today` with a specific date or leave it as `\today` to automatically use the current date.

`Your Document Title` को अपने डॉक्यूमेंट के शीर्षक से बदलें, `Your Name` को अपने नाम या लेखक के नाम से बदलें, और `\today` को किसी विशिष्ट तिथि से बदलें। यदि आप `\today` को वैसे ही छोड़ते हैं, तो यह स्वतः वर्तमान तिथि का उपयोग करेगा।

Once you've specified the title, author, and date in the preamble, you can use the `\maketitle` command in the document body to generate the title, author, and date.

प्रीएम्बल में शीर्षक (Title), लेखक (Author) और तिथि (Date) निर्धारित करने के बाद, आप डॉक्यूमेंट के मुख्य भाग में `\maketitle` कमांड का उपयोग करके शीर्षक, लेखक और तिथि को प्रदर्शित कर सकते हैं।

```

\maketitle

```

This command generates the title, author, and date based on the information provided in the preamble.

यह कमांड प्रीएम्बल में दी गई जानकारी के आधार पर डॉक्यूमेंट का शीर्षक, लेखक और तिथि को स्वतः उत्पन्न करता है।

## Example / उदाहरण

Let's see an example of using the title, author, and date in a LaTeX document:

शीर्षक, लेखक और तारीख का उपयोग करते हुए उदाहरण देखें:

```

\documentclass{article}

\title{My First LaTeX Document}
\author{Dr. Kunwer Mrityunjay Singh}
\date{\today}

\begin{document}

\maketitle

\section{Introduction}
\lipsum[1]

\end{document}

```

In this example, we've specified the title as `My First LaTeX Document`, the author as `Dr. Kunwer Mrityunjay Singh`, and the date as `\today`. When you compile the document, the title, author, and date will be displayed at the beginning of the document.

इस उदाहरण में शीर्षक `My First LaTeX Document`, लेखक `Dr. Kunwer Mrityunjay Singh` और तारीख `\today` निर्धारित की गई है। कम्पाइल करने पर यह जानकारी डॉक्यूमेंट के शुरू में दिखाई देगी।

### 3.3 LaTeX Tutorial: Sections, Subsections, and Subsubsections / LaTeX ट्यूटोरियल: सेक्शन, सब-सेक्शन और सब-सब-सेक्शन

In this tutorial, we'll discuss how to structure your LaTeX documents using **sections**, **subsections**, and **subsubsections**.

इस भाग में डॉक्यूमेंट को सेक्शन, सबसेक्शन और सबसबसेक्शन से व्यवस्थित करना सीखेंगे।

#### 3.3.1 Sections / सेक्शन

Sections divide your document into major parts and provide a hierarchical structure.

सेक्शन आपके डॉक्यूमेंट को प्रमुख भागों में विभाजित करते हैं और उसे एक पदानुक्रमित (हायरार्किकल) संरचना प्रदान करते हैं।

#### 3.3.2 Subsections / सबसेक्शन

Subsections further divide sections into smaller parts.

सबसेक्शन (Subsections) सेक्शनों को आगे छोटे-छोटे भागों में विभाजित करते हैं।

#### 3.3.3 Subsubsections / सब-सबसेक्शन

Subsubsections provide even finer divisions within subsections.

सबसबसेक्शन (Subsubsections) सबसेक्शनों के भीतर और भी सूक्ष्म स्तर पर विभाजन प्रदान करते हैं।

#### 3.3.4 Example / उदाहरण

Let's see an example of using sections, subsections, and subsubsections in a LaTeX document:

आइए LaTeX डॉक्यूमेंट में सेक्शन, सबसेक्शन और सबसबसेक्शन के उपयोग का एक उदाहरण देखें:

```
\documentclass{article}

\begin{document}

\section{Introduction}

This is the introduction section.

\subsection{Overview}

This is the overview subsection.

\subsubsection{Background}

This is the background subsubsection.

\section{Main Content}

This is the main content section.

\subsection{Methodology}

This is the methodology subsection.

\end{document}
```

In this example, we have structured the document into sections, subsections, and subsubsections, each with descriptive titles.

इस उदाहरण में हमने डॉक्यूमेंट को सेक्शन, सबसेक्शन और सबसबसेक्शन में व्यवस्थित किया है, जहाँ प्रत्येक का अपना वर्णनात्मक शीर्षक दिया गया है।

When compiled, the document will display a table of contents with these hierarchical divisions.

कंपाइल होने पर यह डॉक्यूमेंट इन पदानुक्रमित विभाजनों के साथ विषय-सूची (Table of Contents) प्रदर्शित करेगा।

# 4. Document Formatting / डॉक्यूमेंट का स्वरूपण

Document Formatting focuses on controlling the appearance and layout of a LaTeX document to achieve a clean, professional, and readable output. In this chapter, we explore how LaTeX handles page layout and formatting, with special emphasis on adjusting page margins and other layout parameters using appropriate packages.

डॉक्यूमेंट फॉर्मेटिंग का उद्देश्य LaTeX डॉक्यूमेंट की बनावट और लेआउट को नियंत्रित करना है, ताकि दस्तावेज़ साफ़, पेशेवर और पढ़ने में सुगम दिखाई दे। इस अध्याय में हम यह समझेंगे कि LaTeX पेज लेआउट और फॉर्मेटिंग को कैसे संभालता है, विशेष रूप से उपयुक्त पैकेजों की सहायता से पेज मार्जिन और अन्य लेआउट पैरामीटर को कैसे समायोजित किया जाता है।

## 4.1 Customizing Page Margins / पेज मार्जिन को अनुकूलित करना

In this section, we'll discuss how to customize page margins in LaTeX documents.

इस अनुभाग में हम LaTeX डॉक्यूमेंट में पेज मार्जिन को अनुकूलित (Customize) करने की प्रक्रिया पर चर्चा करेंगे।

### 4.1.1 Using the geometry Package / geometry पैकेज का उपयोग

The geometry package allows you to customize various aspects of the page layout, including margins, paper size, and orientation. To customize page margins, use the geometry package with the margin option.

geometry पैकेज पेज लेआउट के विभिन्न पहलुओं को अनुकूलित करने की सुविधा देता है, जिनमें मार्जिन, पेपर साइज और ओरिएंटेशन शामिल हैं। पेज मार्जिन को अनुकूलित करने के लिए geometry पैकेज का उपयोग margini विकल्प के साथ किया जाता है।

#### Example / उदाहरण

```
\usepackage[margin=1in]{geometry}
```

This sets all margins to 1 inch. You can specify different values for the left, right, top, and bottom margins separately. यह सभी मार्जिन को 1 इंच पर सेट करता है। आप बाएँ (Left), दाएँ (Right), ऊपर (Top) और नीचे (Bottom) के मार्जिन के लिए अलग-अलग मान भी निर्धारित कर सकते हैं।

#### Example / उदाहरण

```
\usepackage[left=1in,right=1.5in,top=1.25in,bottom=1.5in]{geometry}
```

This sets the left margin to 1 inch, the right margin to 1.5 inches, the top margin to 1.25 inches, and the bottom margin to 1.5 inches.

यह बाएँ मार्जिन को 1 इंच, दाएँ को 1.5 इंच, ऊपर को 1.25 इंच और नीचे को 1.5 इंच सेट करता है।

#### Example / उदाहरण

Let's see an example of customizing page margins in a LaTeX document: आइए एक उदाहरण देखें जिसमें पेज मार्जिन बदले गए हैं:

```
\documentclass{article} % Define the document class
\usepackage[margin=1in]{geometry} % Customize page margins

\begin{document} % Begin the document

\section{Introduction} % Define a section titled "Introduction"

This is the introduction section.

\end{document}
```

In this example, we've set all margins to 1 inch using the `geometry` package. You can adjust the margin values as needed to achieve your desired layout.

इस उदाहरण में `geometry` पैकेज के माध्यम से सभी मार्जिन 1 इंच पर सेट किए गए हैं। आप अपनी आवश्यकताओं के अनुसार इन मानों को बदल सकते हैं।

## 4.2 Changing Paper Size / पेपर साइज बदलना

In this tutorial, we'll discuss how to change the paper size in LaTeX documents.

इस भाग में हम सीखेंगे कि LaTeX डॉक्यूमेंट में पेपर साइज कैसे बदला जाता है।

### 4.2.1 Using the `geometry` Package / `geometry` पैकेज का उपयोग

The `geometry` package allows you to customize various aspects of the page layout, including paper size. To change the paper size, use the `geometry` package with the `papersize` option.

`geometry` पैकेज पेज लेआउट के विभिन्न पहलुओं को अनुकूलित करने की सुविधा देता है, जिनमें पेपर साइज भी शामिल है। पेपर साइज बदलने के लिए `geometry` पैकेज का उपयोग `papersize` विकल्प के साथ किया जाता है।

#### Example / उदाहरण

```
\usepackage[papersize={width,height}]{geometry}
```

Replace `width` and `height` with the desired dimensions of the paper. You can use units such as `cm`, `in`, or `mm`.

`width` और `height` को अपने इच्छित पेपर साइज से बदलें। आप `cm`, `in` या `mm` जैसी इकाइयाँ प्रयोग कर सकते हैं।

#### Example / उदाहरण

Let's see an example of changing the paper size in a LaTeX document: पेपर साइज बदलने का एक उदाहरण देखें:

```
\documentclass{article} % Define the document class
\usepackage[papersize={8.5in,11in}]{geometry} % Change paper size to letter

\begin{document}

\section{Introduction} % Define a section titled "Introduction"

This is the introduction section.

\end{document}
```

This example demonstrates how to change the paper size of a LaTeX document using the `geometry` package. The document uses the `article` class and sets the paper size to  $8.5 \times 11$  inches (letter size). A simple section titled "Introduction" is created inside the document, showing how content is structured after setting the page layout.

यह उदाहरण `geometry` पैकेज का उपयोग करके LaTeX डॉक्यूमेंट के पेपर साइज को बदलने की प्रक्रिया को दर्शाता है। इसमें `article` डॉक्यूमेंट क्लास का प्रयोग किया गया है और पेपर साइज को  $8.5 \times 11$  इंच (लेटर साइज) पर सेट किया गया है। इसके बाद "Introduction" शीर्षक वाला एक सरल सेक्शन बनाया गया है, जिससे यह स्पष्ट होता है कि पेज लेआउट सेट करने के बाद डॉक्यूमेंट की सामग्री किस प्रकार संरचित की जाती है।

## 4.3 Changing Background Color / बैकग्राउंड रंग बदलना

In this section, we'll discuss how to change the background color in LaTeX documents.

इस अनुभाग में हम LaTeX डॉक्यूमेंट में पृष्ठभूमि (Background) का रंग बदलना सीखेंगे।

### 4.3.1 Using the `pagecolor` Package / `pagecolor` पैकेज का उपयोग

The `pagecolor` package allows you to change the background color of the entire page.

`pagecolor` पैकेज पूरे पृष्ठ का रंग बदलने की सुविधा देता है।

**Example / उदाहरण**

```
\pagecolor{blue!10} % Change background color to light blue
```

This example demonstrates how to change the background color of a page in a LaTeX document using the `\pagecolor` command. The color specification `blue!10` represents a light shade of blue, where 10 indicates the intensity of the color. Specifically, `blue!10` means that 10% blue is mixed with 90% white, resulting in a very light blue background. Lower values produce lighter colors, while higher values produce darker shades. This technique is useful for improving visual appearance, highlighting content, or designing presentation-style documents.

यह उदाहरण LaTeX डॉक्यूमेंट में पेज की पृष्ठभूमि का रंग बदलने के लिए `\pagecolor` कमांड के उपयोग को दर्शाता है। यहाँ `blue!10` एक हल्के नीले रंग को दर्शाता है। 10 रंग की तीव्रता (Intensity) को बताता है, जिसका अर्थ है कि 10% नीला रंग और 90% सफ़ेद रंग मिलाकर यह रंग बनाया गया है। कम प्रतिशत मान देने पर रंग हल्का होता है, जबकि अधिक प्रतिशत मान देने पर रंग गहरा दिखाई देता है। इस प्रकार पृष्ठभूमि का रंग बदलना डॉक्यूमेंट की दृश्य गुणवत्ता बढ़ाने, महत्वपूर्ण सामग्री को उभारने या प्रस्तुतीकरण शैली के दस्तावेज़ बनाने में उपयोगी होता है।

Replace `blue!10` with the desired color. You can specify colors using predefined color names or RGB/HTML color codes.

`blue!10` को अपने मनचाहे रंग से बदलें। आप पूर्वनिर्धारित रंग नामों या RGB/HTML कलर कोड का प्रयोग कर सकते हैं।

**Example / उदाहरण**

Let's see an example of changing the background color in a LaTeX document:

आइए LaTeX डॉक्यूमेंट में बैकग्राउंड रंग बदलने का एक उदाहरण देखें:

```
\documentclass{article} % Define the document class
\usepackage{xcolor} % Load the xcolor package for coloring text
\usepackage{lipsum} % Load the lipsum package for generating dummy text
\usepackage{pagecolor} % Load the pagecolor package for changing background color

\begin{document} % Begin the document

\pagecolor{yellow!10} % Change background color to light yellow

\section{Introduction} % Define a section titled "Introduction"

\lipsum[1-4] % Insert dummy text

\end{document}
```

In this example, we've changed the background color to light yellow using the `pagecolor` package.

इस उदाहरण में `pagecolor` पैकेज का उपयोग करके बैकग्राउंड का रंग हल्का पीला कर दिया गया है।

## 4.4 Header and Footer / हेडर और फूटर

In this section, we will learn how to customize headers and footers in LaTeX using the `fancyhdr` package.

इस भाग में हम LaTeX में `fancyhdr` पैकेज का उपयोग करके हेडर और फूटर को अनुकूलित करना सीखेंगे।

### 4.4.1 Setting Up / सेटअप करना

To get started, we need to include the `fancyhdr` package in the preamble of our document.

शुरू करने के लिए हमें डॉक्यूमेंट के प्रीएम्बल में `fancyhdr` पैकेज को शामिल करना होगा।

```
\usepackage{fancyhdr}
```

Next, we set the page style to `fancy` to enable custom headers and footers. इसके बाद कस्टम हेडर और फूटर सक्षम करने के लिए पेज स्टाइल को `fancy` पर सेट किया जाता है।

```
\pagestyle{fancy}
```

## 4.4.2 Customizing Headers / हेडर को अनुकूलित करना

We can customize the headers using the following commands:

निम्नलिखित कमांड का उपयोग करके हम हेडर को अनुकूलित कर सकते हैं:

```
\fancyhead[LE,RO]{\slshape \leftmark}
\fancyhead[RE,LO]{\slshape \rightmark}
```

This code sets the headers on even pages (LE for Left Even, RO for Right Odd) to display the chapter title in slanted font on the left (`\leftmark`) and the section title in slanted font on the right (`\rightmark`). Similarly, it sets the headers on odd pages (RE for Right Even, LO for Left Odd) to display the section title on the left and the chapter title on the right.

यह कोड सम पृष्ठों (LE = Left Even, RO = Right Odd) पर बाएँ ओर तिरछे फ़ॉन्ट में अध्याय का शीर्षक (`\leftmark`) और दाएँ ओर सेक्शन का शीर्षक (`\rightmark`) दिखाता है। इसी प्रकार विषम पृष्ठों (RE = Right Even, LO = Left Odd) पर बाएँ ओर सेक्शन का और दाएँ ओर अध्याय का शीर्षक प्रदर्शित होता है।

## 4.4.3 Customizing Footers / फूटर को अनुकूलित करना

Similarly, we can customize the footers using the following command:

इसी प्रकार निम्नलिखित कमांड का उपयोग करके फूटर को भी अनुकूलित किया जा सकता है:

```
\fancyfoot[CE,CO]{\thepage}
```

This code configures the footer to display the page number using the `\thepage` command at the center (C) of both even (E) and odd (O) pages. As a result, the page number appears centered at the bottom of the page on even pages as well as on odd pages, ensuring a consistent and balanced footer layout throughout the document.

यह कोड फूटर को इस प्रकार कॉन्फ़िगर करता है कि पेज नंबर को `\thepage` कमांड की सहायता से सम (Even) और विषम (Odd) दोनों पृष्ठों के निचले भाग के मध्य (Center) में प्रदर्शित किया जाए। परिणामस्वरूप, सम पृष्ठों पर भी और विषम पृष्ठों पर भी पेज नंबर पृष्ठ के निचले भाग में केंद्र में दिखाई देता है, जिससे पूरे डॉक्यूमेंट में फूटर का लेआउट एक समान और संतुलित बना रहता है।

### Example / उदाहरण

Let's put everything together with a simple example:

आइए एक सरल उदाहरण के साथ पूरे कोड को देखें:

```
\documentclass{article} % Document class declaration: article class for basic document
  formatting

\usepackage{fancyhdr} % Including the fancyhdr package for custom headers and footers

\pagestyle{fancy} % Setting the page style to 'fancy' for custom headers and footers

\fancyhf{} % Clearing the default header and footer settings

\fancyhead[L]{\leftmark} % Setting the left header content to the chapter information (\
  leftmark)
\fancyhead[R]{\rightmark} % Setting the right header[R] content to the section information (\
  rightmark)
\fancyfoot[C]{\thepage} % Setting the centered footer content to the page number

\begin{document} % Beginning of the document

\section{Introduction} % Starting a new section titled 'Introduction'
This is the introduction section.

\newpage % Starting a new page

\section{Main Body} % Starting a new section titled 'Main Body'
This is the main body section.

\newpage % Starting a new page

\section{Conclusion} % Starting a new section titled 'Conclusion'
This is the conclusion section.
```

```
\end{document} % End of the document
```

Compile this code, and you'll have a document with customized headers and footers.

इस कोड को कम्पाइल करने पर आपको कस्टम हेडर और फूटर वाला डॉक्यूमेंट प्राप्त होगा।

This set of commands uses the `fancyhdr` package to customize the header and footer of a LaTeX document.

The package is loaded using `\usepackage{fancyhdr}`, and the page style is set to `fancy` so that custom header and footer settings take effect. The command `\fancyhf{}` clears all default header and footer content. The left header is configured with `\fancyhead[L]{\leftmark}` to display section-related information, while the right header uses `\fancyhead[R]{\rightmark}` to show additional section details.

The footer is set using `\fancyfoot[C]{\thepage}` to display the page number at the center of the bottom of each page. As output, every page shows updated section information in the header and a centered page number in the footer, resulting in a clean, consistent, and professional document layout.

इन सभी कमांड्स का उपयोग `fancyhdr` पैकेज की सहायता से LaTeX डॉक्यूमेंट में कस्टम हेडर और फूटर बनाने के लिए किया जाता है। `\usepackage{fancyhdr}` द्वारा पैकेज को लोड किया जाता है और `\pagestyle{fancy}` से कस्टम हेडर-फूटर लेआउट को सक्रिय किया जाता है। `\fancyhf{}` कमांड डिफॉल्ट हेडर और फूटर की सभी सामग्री को हटाकर एक साफ़ लेआउट प्रदान करती है। इसके बाद `\fancyhead[L]{\leftmark}` के माध्यम से हेडर के बाएँ भाग में सेक्शन से संबंधित जानकारी प्रदर्शित होती है, जबकि `\fancyhead[R]{\rightmark}` द्वारा हेडर के दाएँ भाग में अतिरिक्त सेक्शन जानकारी दिखाई जाती है। अंत में `\fancyfoot[C]{\thepage}` कमांड के माध्यम से प्रत्येक पेज के निचले भाग के मध्य में पेज नंबर प्रदर्शित किया जाता है। इसके आउटपुट के रूप में हर पेज के ऊपर सेक्शन की जानकारी स्वतः अपडेट होती रहती है और नीचे केंद्र में पेज नंबर दिखाई देता है, जिससे पूरा डॉक्यूमेंट सुव्यवस्थित, एकरूप और पेशेवर दिखाई देता है।

## 4.5 Font Size / फ़ॉन्ट साइज़

Font size is an essential aspect of document design in LaTeX. This section will guide you through the various commands and techniques for adjusting font sizes in your document.

LaTeX में फ़ॉन्ट का आकार डॉक्यूमेंट डिज़ाइन का एक महत्वपूर्ण पक्ष है। यह भाग आपको फ़ॉन्ट साइज़ बदलने से संबंधित विभिन्न कमांड और विधियाँ सिखाएगा।

### 4.5.1 Font Size Commands in LaTeX / LaTeX में फ़ॉन्ट साइज़ कमांड्स

Font size commands in LaTeX are used to change the size of text for emphasis, titles, notes, or special formatting. The following example demonstrates how different font size commands affect text appearance.

LaTeX में फ़ॉन्ट साइज़ कमांड्स का उपयोग टेक्स्ट के आकार को बदलने के लिए किया जाता है। इनका प्रयोग शीर्षक, विशेष नोट्स, उदाहरण या किसी महत्वपूर्ण जानकारी को उभारने के लिए किया जाता है। नीचे दिया गया उदाहरण विभिन्न फ़ॉन्ट साइज़ कमांड्स के प्रभाव को दर्शाता है।

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}

\begin{document}

This is normal text.

\tiny This is Tiny text.

\scriptsize This is Scriptsize text.

\footnotesize This is Footnotesize text.

\small This is Small text.

\large This is Large text.

\LARGE This is LARGE text.

\Huge This is Huge text.

\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

This is normal text.

This is Tiny text.

This is Scriptsize text.

This is Footnotesize text.

This is Small text.

This is Large text.

This is LARGE text.

This is Huge text.

**4.5.2 Using Document Class / Document Class का प्रयोग****Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass[12pt]{article}
\begin{document}
This is a document with 12pt font size.
\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

This is a document with 12pt font size.

This LaTeX code creates a simple document using the `article` document class with a base font size of 12 points. The option `[12pt]` specifies that all normal text in the document will be displayed in 12-point font size instead of the default 10pt. The commands `\begin{document}` and `\end{document}` define the beginning and end of the document content. Any text written between these commands appears as the main content of the document and is rendered using the specified font size when compiled.

यह LaTeX कोड `article` डॉक्यूमेंट क्लास का उपयोग करके एक सरल डॉक्यूमेंट तैयार करता है, जिसमें आधार फ्रॉन्ट साइज 12 पॉइंट निर्धारित किया गया है। `[12pt]` विकल्प यह सुनिश्चित करता है कि डॉक्यूमेंट में प्रयुक्त सामान्य टेक्स्ट डिफॉल्ट 10pt के स्थान पर 12pt आकार में प्रदर्शित हो। `\begin{document}` और `\end{document}` कमांड्स डॉक्यूमेंट की सामग्री की शुरुआत और अंत को दर्शाती हैं। इनके बीच लिखा गया टेक्स्ट डॉक्यूमेंट की मुख्य सामग्री होता है और कंपाइल करने पर यह 12pt फ्रॉन्ट साइज में दिखाई देता है।

**4.5.3 Using `\fontsize` and `\selectfont` / `fontsize` और `selectfont` का उपयोग****Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}
\begin{document}
\fontsize{14pt}{16pt}\selectfont This is a 14pt text with 16pt leading.
```

```
\fontsize{18pt}{22pt}\selectfont This is an 18pt text with 22pt leading.
\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

This is a 14pt text with 16pt leading.  
This is an 18pt text with 22pt leading.

This LaTeX example demonstrates how to manually control font size and line spacing using the `\fontsize` command.

The command `\fontsize{14pt}{16pt} \selectfont` sets the text size to 14 points and the line spacing (leading) to 16 points.

Similarly, `\fontsize{18pt}{22pt}\selectfont` increases the font size to 18 points with a line spacing of 22 points. The `\selectfont` command is mandatory, as it applies the specified font size and line spacing settings. This approach is useful when precise typographic control is required beyond the predefined font size commands provided by LaTeX.

यह उदाहरण `\fontsize` कमांड का उपयोग करके फ्रॉन्ट साइज और लाइन स्पेसिंग को मैनुअली नियंत्रित करने की विधि को दर्शाता है।

`\fontsize{14pt}{16pt}\selectfont` कमांड 14pt फ्रॉन्ट साइज और 16pt लाइन स्पेसिंग (लीडिंग) निर्धारित करती है। इसी प्रकार, `\fontsize{18pt}{22pt}\selectfont` कमांड 18pt फ्रॉन्ट साइज और 22pt लाइन स्पेसिंग सेट करती है। यहाँ `\selectfont` कमांड आवश्यक होती है, क्योंकि यह निर्धारित किए गए फ्रॉन्ट साइज और लाइन स्पेसिंग को लागू करती है। यह विधि तब उपयोगी होती है जब डॉक्यूमेंट में टेक्स्ट के स्वरूप पर सटीक और अधिक नियंत्रण की आवश्यकता होती है।

## 4.6 Font Styles / फ्रॉन्ट स्टाइल

### 4.6.1 Bold Font / बोल्ड फ्रॉन्ट

To make text bold in LaTeX, you have two main options:

LaTeX में टेक्स्ट को बोल्ड करने के दो मुख्य तरीके हैं:

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}
\begin{document}
\textbf{This text is bold.}
\bfseries {This text is also bold.}
\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

**This text is bold.**  
**This text is also bold.**

- `\textbf{}`: This command is used to make the enclosed text bold.
- `\bfseries`: This command switches the font to bold for the entire scope until the end of the current group or environment.

- `\textbf{}` कमांड का उपयोग केवल दिए गए टेक्स्ट को बोल्ड करने के लिए किया जाता है।
- `\bfseries` कमांड पूरे समूह या एनवायरनमेंट की सीमा तक फ्रॉन्ट को बोल्ड कर देता है।

## 4.6.2 Italic Font / इटैलिक फ्रॉन्ट

To make text italic in LaTeX, similarly, you have two main options:

इसी प्रकार LaTeX में टेक्स्ट को इटैलिक बनाने के दो मुख्य तरीके हैं:

- `\textit{}`: This command is used to make the enclosed text italic.
- `\itshape`: This command switches the font to italic for the entire scope until the end of the current group or environment.
- `\textit{}` केवल चयनित शब्दों को इटैलिक बनाता है, जबकि
- `\itshape` पूरे ब्लॉक का फ्रॉन्ट इटैलिक कर देता है।

**Example / उदाहरण**

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}
\begin{document}
\textit{This text is italicized.}
{\itshape This text is also italicized.}
\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

*This text is italicized.*  
*This text is also italicized.*

## 4.6.3 Underline Font / अंडरलाइन फ्रॉन्ट

LaTeX does not have built-in support for underlining text out of the box. However, you can achieve underlining using the `ulem` package. After including the package, you can use the `\uuline{}` command to underline text.

LaTeX में डिफ़ॉल्ट रूप से टेक्स्ट को अंडरलाइन करने की सुविधा नहीं होती। हालाँकि आप `ulem` पैकेज की सहायता से अंडरलाइन प्रभाव प्राप्त कर सकते हैं जोड़ने के बाद आप टेक्स्ट को अंडरलाइन करने के लिए `\uuline{}` कमांड का उपयोग कर सकते हैं।

**Example / उदाहरण**

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}
\usepackage[normalem]{ulem}
\begin{document}
\uuline{This text is underlined.}
\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

This text is underlined.

#### 4.6.4 Emphasize Font / एम्फ़साइज़ टेक्स्ट

To emphasize text in LaTeX, you can use the `\emph{}` command. The appearance of emphasized text typically depends on the document class and may be italicized or appear bold, depending on the context.

LaTeX में टेक्स्ट को उभारने के लिए `\emph{}` कमांड का उपयोग किया जाता है। एम्फ़साइज़ किए गए टेक्स्ट का रूप डॉक्यूमेंट क्लास पर निर्भर करता है और वह इटैलिक या बोल्ड दिखाई दे सकता है।

**Example / उदाहरण**

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}
\begin{document}
\emph{This text is emphasized.}
\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

*This text is emphasized.*

#### 4.6.5 Typewriter Font / टाइपराइटर फ़ॉन्ट

To display text in a typewriter or monospaced font, you can use the `\texttt{}` command.

टाइपराइटर या मोनोस्पेस फ़ॉन्ट दिखाने के लिए `\texttt{}` कमांड का प्रयोग किया जाता है।

**Example / उदाहरण**

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}
\begin{document}
\texttt{This text is in a typewriter font.}
\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

This text is in a typewriter font.

## 4.7 Font Families / फ़ॉन्ट फ़ैमिली

In LaTeX, you can customize the appearance of your text by using different font families. Common font families include serif, sans-serif, and monospace. This tutorial will cover how to use these font families in your LaTeX documents.

LaTeX में आप विभिन्न फ़ॉन्ट फ़ैमिली का उपयोग करके टेक्स्ट का रूप बदल सकते हैं। सामान्य फ़ॉन्ट फ़ैमिली हैं: serif, sans-serif और monospace। इस भाग में आप इन फ़ॉन्टों का प्रयोग करना सीखेंगे।

### 4.7.1 Serif Font Family / सेरिफ़ फ़ॉन्ट फ़ैमिली

Serif fonts have small lines or strokes attached to the ends of letters. They are commonly used for body text in printed materials.

Serif फ़ॉन्ट में अक्षरों के सिरों पर छोटी रेखाएँ या घुमाव होते हैं। इनका उपयोग आमतौर पर पुस्तकों और प्रिंटेड सामग्री में किया जाता है।

**Example / उदाहरण**

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}
\begin{document}
{\rmfamily This is serif text.}
\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

This is serif text.

### 4.7.2 Sans-Serif Font Family / सैन्स-सेरिफ़ फ़ॉन्ट फ़ैमिली

Sans-serif fonts do not have the small lines or strokes attached to the ends of letters. They are often used for headings and titles in printed materials and are popular for online content.

Sans-serif फ़ॉन्ट में अक्षरों के सिरों पर छोटी रेखाएँ नहीं होती हैं। इनका उपयोग आमतौर पर शीर्षक और ऑनलाइन सामग्री में किया जाता है।

**Example / उदाहरण**

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}
\begin{document}
{\sffamily This is sans-serif text.}
\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

This is sans-serif text.

### 4.7.3 Monospace Font Family / मोनोस्पेस फ़ॉन्ट फ़ैमिली

Monospace fonts have fixed-width characters where each character occupies the same amount of horizontal space. They are commonly used for code snippets and typewriter text.

Monospace फ़ॉन्ट में प्रत्येक अक्षर समान चौड़ाई घेरता है। इसका उपयोग प्रायः कोड और टाइपराइटर टेक्स्ट के लिए किया जाता है।

**Example / उदाहरण**

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}

\begin{document}

{\ttfamily This is monospace text.}

\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

This is monospace text.

### 4.7.4 Using Font Packages / फ़ॉन्ट पैकेज का उपयोग

To use a different font family, include the corresponding font package in the preamble of your document. Here are examples of using different font families:

किसी अन्य फ़ॉन्ट फ़ैमिली का उपयोग करने के लिए उसे संबंधित फ़ॉन्ट पैकेज के माध्यम से प्रीएम्बल में जोड़ें। नीचे विभिन्न फ़ॉन्ट पैकेज के उदाहरण दिए गए हैं:

```
\usepackage{lmodern}      % Latin Modern (default)
\usepackage{tgadventor}   % TeX Gyre Adventor
\usepackage{tgheros}      % TeX Gyre Heros
\usepackage{tgcursor}     % TeX Gyre Cursor
```

Feel free to explore and include other font packages based on your preferences.

अपनी पसंद के अनुसार अन्य फ़ॉन्ट पैकेज को भी आजमाएँ और शामिल करें।

### 4.7.5 Changing Font Color / फ़ॉन्ट का रंग बदलना

You can change the color of text using the `xcolor` package.

आप `xcolor` पैकेज का उपयोग करके टेक्स्ट का रंग बदल सकते हैं।

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}
\usepackage{xcolor}

\begin{document}
```

```
\textcolor{red}{Your Text}
\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**



## 4.8 Text Symbols / टेक्स्ट सिंबल

LaTeX supports various text symbols. Here are some examples:

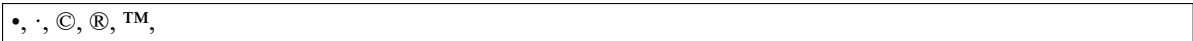
LaTeX विभिन्न प्रकार के टेक्स्ट सिंबल को सपोर्ट करता है। नीचे कुछ उदाहरण दिए गए हैं:

### 4.8.1 Common Text Symbols / सामान्य टेक्स्ट सिंबल

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}
\begin{document}
\documentclass{article}
\begin{document}
\textbullet, \textperiodcentered, \textcopyright, \textregistered, \texttrademark,
\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**




### 4.8.2 Typography / टाइपोग्राफी

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}
\begin{document}
\textbackslash, \textbar, \textunderscore, \textellipsis
\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**



## 4.9 Text Alignment in LaTeX / LaTeX में टेक्स्ट अलाइनमेंट

### 4.9.1 Center Alignment / मध्य संरेखण

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}

\begin{document}

\begin{center}
This text is centered.
\end{center}

\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

This text is centered.

## 4.9.2 Left Alignment / बाएँ संरेखण (Left Alignment)

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}

\begin{document}

\raggedright
This text is left-aligned.

\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

This text is left-aligned.

## 4.9.3 Right Alignment / दाएँ संरेखण (Right Alignment)

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}

\begin{document}

\raggedleft
This text is right-aligned.

\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

This text is right-aligned.

#### 4.9.4 Left-aligned Environment / बाएँ संरेखित (Left-aligned) एनवायरनमेंट

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}

\begin{document}

\begin{flushleft}
This is a left-aligned block of text.
It will continue until the end of the environment.
\end{flushleft}

\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

This is a left-aligned block of text. It will continue until the end of the environment.

#### 4.9.5 Right-aligned Environment / दाएँ संरेखित (Right-aligned) एनवायरनमेंट

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}

\begin{document}

\begin{flushright}
This is a right-aligned block of text.
It will continue until the end of the environment.
\end{flushright}

\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

This is a right-aligned block of text. It will continue until the end of the environment.

#### 4.9.6 Quotations / उद्धरण (Quotations)

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}
\begin{document}

  \begin{quote}
    This is a short quote.
  \end{quote}

\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

This is a short quote.

### 4.10 Lists / सूचियाँ

In LaTeX, you can create lists using the `itemize`, `enumerate`, and `description` environments. These environments allow you to organize and present information in a structured manner.

LaTeX में आप सूचियाँ बनाने के लिए `itemize`, `enumerate`, और `description` एनवायरनमेंट का उपयोग कर सकते हैं। ये एनवायरनमेंट जानकारी को सुव्यवस्थित रूप में प्रस्तुत करने में सहायक होते हैं।

### 4.10.1 Itemize / itemize एनवायरनमेंट

The itemize environment is used to create bulleted lists.

itemize एनवायरनमेंट बुलेट वाली सूची (bulleted list) बनाने के लिए उपयोग किया जाता है।

**Example / उदाहरण**

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}
\begin{document}
\begin{itemize}
  \item First item
  \item Second item
  \item Third item
\end{itemize}
\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

- First item
- Second item
- Third item

### 4.10.2 Enumerate / enumerate एनवायरनमेंट

The enumerate environment is used to create numbered lists.

enumerate एनवायरनमेंट क्रमांकित सूची (numbered list) बनाने के लिए उपयोग किया जाता है।

**Example / उदाहरण**

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}
\documentclass{article}
\begin{document}
\begin{enumerate}
  \item First item
  \item Second item
  \item Third item
\end{enumerate}
\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

1. First item
2. Second item
3. Third item

**4.10.3 Description / description एनवायरनमेंट**

The description environment is used to create lists with descriptive labels.

description एनवायरनमेंट विवरणात्मक लेबल्स के साथ सूची बनाने के लिए उपयोग किया जाता है।

**Example / उदाहरण****Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}
\begin{document}
\begin{description}
  \item[First] First item
  \item[Second] Second item
  \item[Third] Third item
\end{description}
\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

- First** First item
- Second** Second item
- Third** Third item

**4.11 Creating Paragraphs / पैराग्राफ बनाना**

In LaTeX, paragraphs are units of text separated by blank lines or by the `\par` command. They are the basic building blocks of text in a document.

LaTeX में पैराग्राफ ऐसे टेक्स्ट भाग होते हैं जिन्हें खाली लाइन या `\par` कमांड द्वारा अलग किया जाता है। ये डॉक्यूमेंट के मूल निर्माण खंड (building blocks) होते हैं।

**4.11.1 Using Blank Lines / खाली पंक्तियों का उपयोग**

To create a new paragraph in LaTeX, you simply leave a blank line between blocks of text:

LaTeX में नया पैराग्राफ बनाने के लिए टेक्स्ट के दो भागों के बीच एक खाली पंक्ति छोड़नी होती है:

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}

\begin{document}

This is the first paragraph. It ends with a blank line.

This is the second paragraph.

\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

This is the first paragraph. It ends with a blank line.  
This is the second paragraph.

**4.11.2 Using \par Command / \par कमांड का उपयोग**

Alternatively, you can use the `\par` command to explicitly indicate the end of a paragraph:  
वैकल्पिक रूप से, किसी पैराग्राफ के अंत को स्पष्ट करने के लिए `\par` कमांड का भी उपयोग किया जा सकता है:

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}

\begin{document}

This is the first paragraph. It ends with \par blank line.

\par This is the second paragraph.

\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

This is the first paragraph. It ends with  
blank line.  
This is the second paragraph.

# 5. Mathematics in LaTeX / LaTeX में गणित

## 5.1 Inline and Display Math Mode / इनलाइन और डिस्प्ले मैथ मोड

### 5.1.1 Inline Math Mode / इनलाइन मैथ मोड

Inline math mode is used for including mathematical expressions within the text. To enter inline math mode, use the dollar sign (\$) both at the beginning and end of the mathematical expression.

इनलाइन मैथ मोड का उपयोग टेक्स्ट के बीच गणितीय अभिव्यक्तियाँ लिखने के लिए किया जाता है। इसे उपयोग करने के लिए गणितीय अभिव्यक्ति की शुरुआत और अंत में डॉलर चिन्ह (\$) लगाएँ।

### 5.1.2 Display Math Mode / डिस्प्ले मैथ मोड

Display math mode is used for displaying larger mathematical expressions on separate lines. To enter display math mode, use double dollar signs (\$\$) or the `equation` environment.

डिस्प्ले मैथ मोड का उपयोग बड़े गणितीय समीकरणों को अलग पंक्ति में दिखाने के लिए किया जाता है। इसे उपयोग करने के लिए दो डॉलर चिन्ह (\$\$) या `equation` एनवायरनमेंट का प्रयोग करें।

For example, the Pythagorean theorem can be displayed as:

उदाहरण के लिए, पाइथागोरस प्रमेय को इस प्रकार लिखा जा सकता है:

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}
\begin{document}
$$c = \sqrt{a^2 + b^2}$$
\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

$$c = \sqrt{a^2 + b^2}$$

Alternatively, you can use the `equation` environment for the same result:

इसी परिणाम के लिए आप `equation` एनवायरनमेंट का भी उपयोग कर सकते हैं:

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}
\begin{document}
\begin{equation}
c = \sqrt{a^2 + b^2}
\end{equation}
\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

$$c = \sqrt{a^2 + b^2} \tag{5.1}$$

Display math mode is particularly useful for equations that require more space and need to be emphasized.

डिस्टले मैथ मोड विशेष रूप से उन समीकरणों के लिए उपयोगी होता है जिनमें अधिक स्थान की आवश्यकता होती है या जिन्हें विशेष रूप से दर्शाना हो।

## 5.2 AMS Math Package / AMS मैथ पैकेज

The AMS (American Mathematical Society) math package (`amsmath`) is commonly used to enhance the functionality of LaTeX for mathematical typesetting. It provides additional commands and environments for handling mathematical content.

AMS (American Mathematical Society) द्वारा बनाया गया मैथ पैकेज `amsmath` LaTeX में गणितीय टाइपसेटिंग की क्षमता बढ़ाता है। इसमें अतिरिक्त कमांड और एनवायरनमेंट दिए गए हैं जो गणितीय सामग्री को आसानी से नियंत्रित करने में मदद करते हैं।

## 5.3 Mathematical Symbols and Equations / गणितीय प्रतीक और समीकरण

### 5.3.1 Greek Letters / ग्रीक लेटर्स

Input (LaTeX Code) / इनपुट (LaTeX कोड)

```
\documentclass{article}
\begin{document}
\[
\alpha, \beta, \gamma, \delta, \epsilon, \zeta, \eta, \theta, \iota, \kappa,
\lambda, \mu, \nu, \xi, \pi, \rho, \sigma, \tau, \upsilon, \phi, \chi, \psi, \omega
\]
\end{document}
```

Output (PDF Result) / आउटपुट (PDF परिणाम)

$$\alpha, \beta, \gamma, \delta, \epsilon, \zeta, \eta, \theta, \iota, \kappa, \lambda, \mu, \nu, \xi, \pi, \rho, \sigma, \tau, \upsilon, \phi, \chi, \psi, \omega$$

### 5.3.2 Set and Relation / सेट और रिलेशन

Input (LaTeX Code) / इनपुट (LaTeX कोड)

```
\documentclass{article}
\begin{document}
\[
\in, \subset, \approx, \leq, \geq, \neq
\]
\end{document}
```

Output (PDF Result) / आउटपुट (PDF परिणाम)

$$\in, \subset, \approx, \leq, \geq, \neq$$

### 5.3.3 Operators / ऑपरेटर्स

Input (LaTeX Code) / इनपुट (LaTeX कोड)

```
\documentclass{article}
\begin{document}
\[
\sum, \prod, \int, \lim, \sum_{i=1}^n a_i,
\prod_{i=1}^n b_i
\]
```

```
\]
\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

$$\sum, \prod, \int, \lim, \sum_{i=1}^n a_i, \prod_{i=1}^n b_i$$

### 5.3.4 Superscripts and Subscripts / सुपरस्क्रिप्ट और सबस्क्रिप्ट

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}
\begin{document}
$a^2, x_1$
\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

$a^2, x_1$

### 5.3.5 Trigonometric Functions / त्रिकोणमितीय फलन

Use commands like `\sin`, `\cos`, and `\tan` for trigonometric functions. त्रिकोणमितीय फलनों के लिए `\sin`, `\cos` और `\tan` कमांड का उपयोग किया जाता है।

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}
\begin{document}
$\sin(\theta), \cos(\theta), \tan(\theta)$
\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

$\sin(\theta), \cos(\theta), \tan(\theta)$

### 5.3.6 Arrows / तीर चिह्न

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}
\begin{document}
\[
\leftarrow, \rightarrow, \leftrightarrows, \uparrow, \downarrow,
\Uparrow, \Downarrow
\]
\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

$$\leftarrow, \rightarrow, \leftrightarrow, \uparrow, \downarrow, \Uparrow, \Downarrow$$
**5.4 Equation Environments / समीकरण एनवायरनमेंट****5.4.1 Inline Equations / इनलाइन समीकरण**

Inline equations are enclosed with a single dollar sign ( $\$...\$$ ) or the  $\backslash(\dots\backslash)$  command.

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}
\begin{document}
The quadratic formula is  $x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$ .
\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

The quadratic formula is  $x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$ .

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}
\begin{document}
Another way to write the quadratic formula is
 $(x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a})$ .
\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

Another way to write the quadratic formula is  $x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$ .

**5.4.2 Equations in Display Mode / डिस्प्ले मोड में समीकरण**

Equations in Display Mode are written in a math environment, such as `equation`, `align`, or `gather`.

डिस्प्ले मोड में समीकरण को मैथ एनवायरनमेंट में लिखा जाता है जैसे `equation`, `align`, या `gather`।

**Equation Environment / समीकरण एनवायरनमेंट**

The equation environment is used for a single, centered equation. It is typically used for important standalone equations.

`equation` एनवायरनमेंट एक अकेले, केंद्रित (centered) समीकरण के लिए उपयोग किया जाता है। यह सामान्यतः महत्वपूर्ण स्वतंत्र समीकरणों के लिए प्रयोग होता है।

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```

\documentclass{article}
\begin{document}

  \begin{equation}
    E = mc^2
  \end{equation}

\end{document}

```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

$$E = mc^2 \quad (5.2)$$

**Align Environment / अलाइन एनवायरनमेंट**

The align environment is used to display and align multiple equations in a structured manner. It is commonly used when several equations need to be aligned at a specific point, most often at the equals sign, which improves readability and presents mathematical expressions in a clear and organized form.

align एनवायरनमेंट का उपयोग एक से अधिक समीकरणों को सुव्यवस्थित रूप से प्रदर्शित करने और उन्हें एक ही बिंदु पर संरेखित (Align) करने के लिए किया जाता है। इसका प्रयोग सामान्यतः तब किया जाता है जब कई समीकरणों को बराबर के चिह्न (Equals sign) पर संरेखित करना हो, जिससे गणितीय अभिव्यक्तियाँ अधिक स्पष्ट, पढ़ने में आसान और व्यवस्थित दिखाई देती हैं।

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```

\documentclass{article}
\usepackage{amsmath}

\begin{document}

  \begin{align}
    F &= ma \\
    E &= \frac{1}{2}mv^2
  \end{align}

\end{document}

```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

$$F = ma \quad (5.3)$$

$$E = \frac{1}{2}mv^2 \quad (5.4)$$

**Gather Environment / गैदर एनवायरनमेंट**

The gather environment is used to display multiple equations one below the other without aligning them at any specific point. Each equation is centered independently, making this environment suitable when alignment at the equals sign is not required and the equations are logically related but visually independent.

gather एनवायरनमेंट का उपयोग एक से अधिक समीकरणों को बिना किसी विशेष बिंदु पर संरेखित किए, एक के नीचे एक प्रदर्शित करने के लिए किया जाता है। इस एनवायरनमेंट में प्रत्येक समीकरण अपने आप केंद्र में दिखाई देता है। जब समीकरणों को बराबर के चिह्न पर संरेखित करने की आवश्यकता न हो और वे केवल क्रमवार दिखाने हों, तब gather एनवायरनमेंट का उपयोग किया जाता है।

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```

\documentclass{article}
\usepackage{amsmath}

\begin{document}

  \begin{gather}
    a = b + c \\
    d = e + f + d
  \end{gather}

\end{document}

```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

$$a = b + c \quad (5.5)$$

$$d = e + f + d \quad (5.6)$$

**Difference Between equation, align, and gather / equation, align और gather के बीच अंतर**

The equation, align, and gather environments are used to display mathematical equations in LaTeX, but they serve different purposes. The equation environment is used to display a single equation with automatic numbering. The align environment is used when multiple equations need to be aligned at a specific point, usually at the equals sign, making it ideal for step-by-step derivations. The gather environment is used to display multiple equations one below the other without any alignment, where each equation is centered independently.

LaTeX में equation, align और gather एनवायरनमेंट का उपयोग समीकरणों को प्रदर्शित करने के लिए किया जाता है, लेकिन इनका उद्देश्य अलग-अलग होता है। equation एनवायरनमेंट का उपयोग एकल समीकरण को क्रमांक (Number) के साथ प्रदर्शित करने के लिए किया जाता है। align एनवायरनमेंट तब उपयोगी होता है जब एक से अधिक समीकरणों को किसी विशेष बिंदु, सामान्यतः बराबर के चिह्न, पर संरेखित करना हो। gather एनवायरनमेंट का उपयोग कई समीकरणों को बिना संरेखण के, एक के नीचे एक केंद्र में प्रदर्शित करने के लिए किया जाता है।

**Summary Table / सारांश तालिका**

Environment	Purpose	Alignment
equation	Single equation	Centered
align	Multiple equations	Aligned (usually at =)
gather	Multiple equations	Centered, no alignment

**5.5 Fractions, Roots, and Summation / भिन्न, वर्गमूल तथा योग****5.5.1 Fraction / भिन्न**

Use `\frac{}{}` for fractions, `\sqrt{}` for square roots, and `\sum` for summation.

भिन्न (fractions) के लिए `\frac{}{}`, वर्गमूल (square root) के लिए `\sqrt{}`, और योग (summation) के लिए `\sum` कमांड का उपयोग किया जाता है।

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```

\documentclass{article}
\usepackage{amsmath}

\begin{document}

  \[
    \frac{n!}{k!(n-k)!} = \binom{n}{k}
  \]

\end{document}

```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

$$\frac{n!}{k!(n-k)!} = \binom{n}{k}$$

### 5.5.2 Square Roots / वर्गमूल

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```

\documentclass{article}
\usepackage{amsmath}

\begin{document}
\[
\sqrt{a^2 + b^2} = c
\]
\end{document}

```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

$$\sqrt{a^2 + b^2} = c$$

### 5.5.3 Summation / संकलन (योग चिन्ह)

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```

\documentclass{article}
\usepackage{amsmath}

\begin{document}

\[
\sum_{i=1}^n i = \frac{n(n+1)}{2}
\]

\end{document}

```

**Output****Output (PDF Result) / आउटपुट (PDF परिणाम)**

$$\sum_{i=1}^n i = \frac{n(n+1)}{2}$$

## 5.6 Matrices / मैट्रिक्स

The 'matrix' environment is used for creating basic matrices. Use the & symbol to separate elements in a row and \\ to move to the next row.

'matrix' एनवायरनमेंट का उपयोग साधारण मैट्रिक्स बनाने के लिए किया जाता है। प्रत्येक पंक्ति में तत्वों को अलग करने के लिए & और अगली पंक्ति में जाने के लिए \\ का उपयोग करें।

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```

\documentclass{article}
\usepackage{amsmath}

\begin{document}

\[
\begin{matrix}
\end{matrix}

```

```

1 & 2 \\
3 & 4
\end{matrix}
\]

\end{document}

```

**Output****Output (PDF Result) / आउटपुट (PDF परिणाम)**

$\begin{matrix} 1 & 2 \\ 3 & 4 \end{matrix}$
--

**5.7 Parentheses and Brackets / कोष्ठक और ब्रैकेट**

You can use different types of parentheses and brackets with the ‘bmatrix’, ‘Bmatrix’, ‘vmatrix’, and ‘Vmatrix’ environments.

आप विभिन्न प्रकार के कोष्ठक और ब्रैकेट का उपयोग ‘bmatrix’, ‘Bmatrix’, ‘vmatrix’, और ‘Vmatrix’ एनवायरनमेंट के साथ कर सकते हैं।

**5.7.1 bmatrix (Square Brackets [ ]) / bmatrix (वर्ग कोष्ठक [ ])****Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```

\documentclass{article}
\usepackage{amsmath}

\begin{document}

\[
\begin{bmatrix}
1 & 2 \\
3 & 4
\end{bmatrix}
\end{bmatrix}

\end{document}

```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$
--

**5.7.2 Bmatrix (Curly Brackets { }) / Bmatrix (मध्य कोष्ठक { })****Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```

\documentclass{article}
\usepackage{amsmath}

\begin{document}

\[
\begin{Bmatrix}
1 & 0 \\
0 & 1
\end{Bmatrix}
\end{Bmatrix}

\end{document}

```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

$$\begin{Bmatrix} 1 & 0 \\ 0 & 1 \end{Bmatrix}$$

**5.7.3 vmatrix (Determinant | |) / vmatrix (डिटरमिनेंट | |)****Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}
\usepackage{amsmath}

\begin{document}

\[
\begin{vmatrix}
2 & 3 \\
1 & 4
\end{vmatrix}
\]

\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

$$\begin{vmatrix} 2 & 3 \\ 1 & 4 \end{vmatrix}$$

**5.7.4 Vmatrix (Norm ||) / Vmatrix (नॉर्म ||)****Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}
\usepackage{amsmath}

\begin{document}

\[
\begin{Vmatrix}
1 & -1 \\
2 & 3
\end{Vmatrix}
\]

\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

$$\begin{Vmatrix} 1 & -1 \\ 2 & 3 \end{Vmatrix}$$

**5.8 Alignment / संरेखण (Alignment)**

The 'align' environment is used to align multiple mathematical elements vertically, especially when working with matrices or equations.

'align' एनवायरनमेंट का उपयोग कई गणितीय तत्वों को ऊर्ध्वाधर रूप से एक सीध में व्यवस्थित (align) करने के लिए किया जाता है, विशेष रूप से मैट्रिक्स या समीकरणों के साथ।

### 5.8.1 Using align with Matrices / मैट्रिक्स के साथ align का उपयोग

Input (LaTeX Code) / इनपुट (LaTeX कोड)

```
\documentclass{article}
\usepackage{amsmath}

\begin{document}

\begin{align*}
&\begin{bmatrix}
1 & 2 \\
3 & 4
\end{bmatrix} \\
&\begin{Bmatrix}
5 & 6 \\
7 & 8
\end{Bmatrix}
\end{align*}

\end{document}
```

Output (PDF Result) / आउटपुट (PDF परिणाम)

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \\ \begin{Bmatrix} 5 & 6 \\ 7 & 8 \end{Bmatrix}$$

# 6. Figures and Tables / चित्र और सारणियाँ

In LaTeX, figures and tables are essential elements for presenting visual and tabular information in documents. This tutorial covers including graphics, captions and labels, positioning figures and tables, and creating tables.

LaTeX में चित्र (figures) और सारणियाँ (tables) डॉक्यूमेंट में दृश्य और तालिकात्मक जानकारी प्रस्तुत करने के महत्वपूर्ण घटक होते हैं। इस भाग में हम ग्राफ़िक्स जोड़ना, कैप्शन और लेबल लगाना, चित्र और सारणी की स्थिति नियंत्रित करना, और टेबल बनाना सीखेंगे।

## 6.1 Figures

### 6.1.1 Including Graphics / ग्राफ़िक्स जोड़ना

To include graphics in LaTeX, you can use the `\includegraphics` command from the `graphicx` package.

LaTeX में चित्र जोड़ने के लिए `graphicx` पैकेज का उपयोग किया जाता है। इस पैकेज के अंतर्गत `\includegraphics` कमांड से इमेज जोड़ सकते हैं।

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}
\usepackage{graphicx}

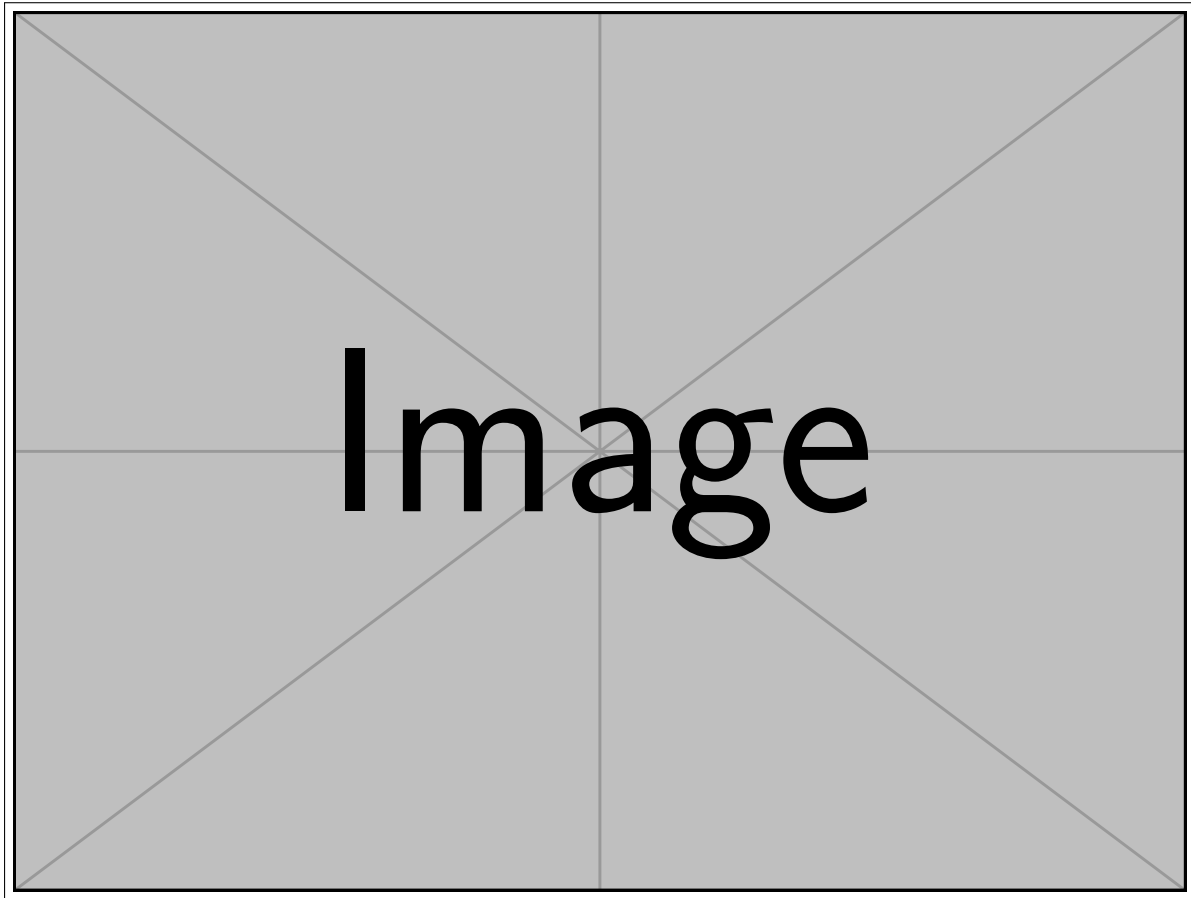
\begin{document}

\begin{figure}

\includegraphics[width=\textwidth]{example-image}

\end{figure}

\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)****6.1.2 Captions / कैप्शन****Input (LaTeX Code) / इनपुट (LaTeX कोड)**

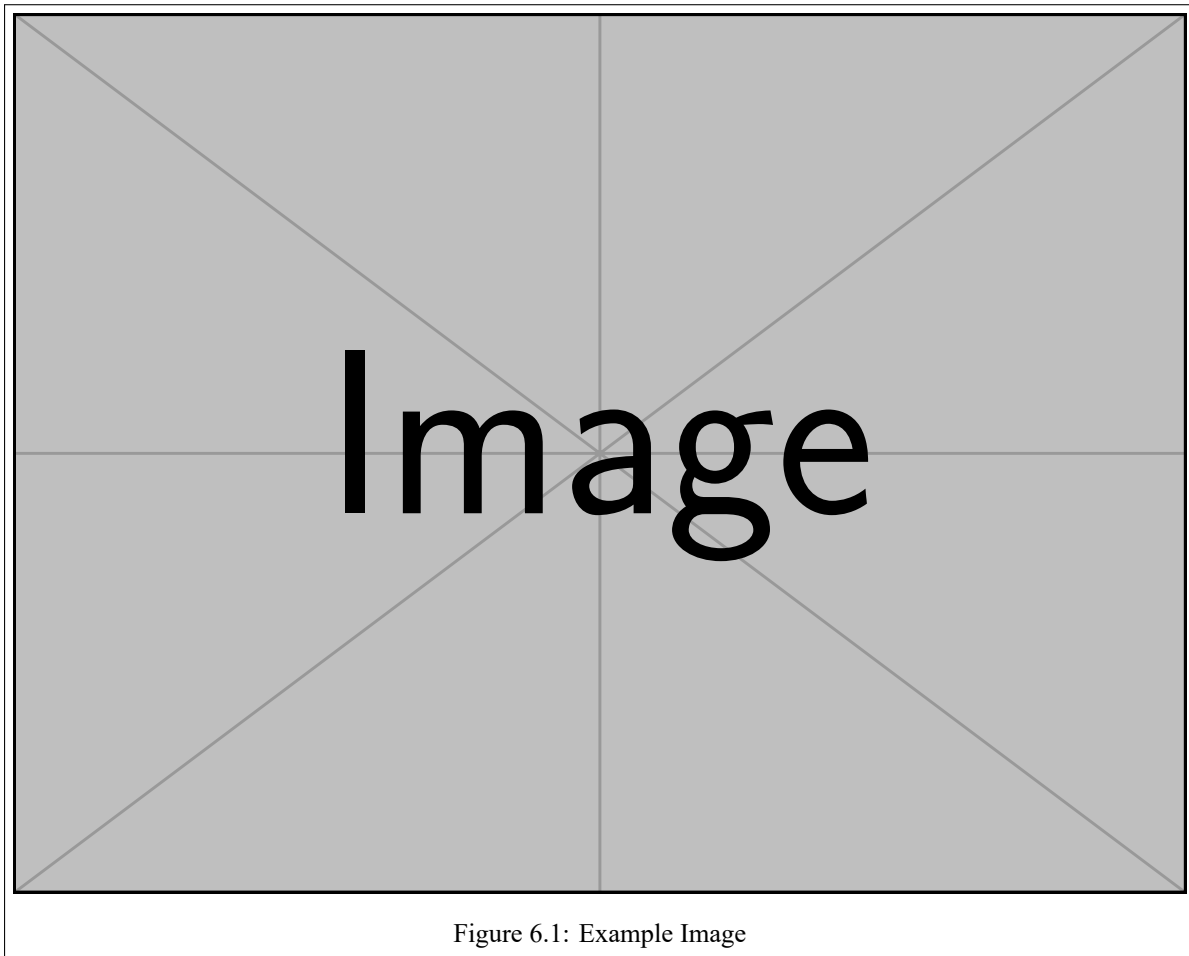
```
\documentclass{article}
\usepackage{caption}
\usepackage{graphicx}
\begin{document}

\begin{figure}

  \includegraphics[width=\textwidth]{example-image}
  \caption{Example Image}

\end{figure}

\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)****6.1.3 Labels / लेबल**

The `\label` command is used in LaTeX to assign a unique identification name to a numbered element such as a figure.

`\label` कमांड का उपयोग LaTeX में किसी क्रमांकित तत्व (जैसे चित्र) को एक विशिष्ट पहचान नाम देने के लिए किया जाता है।

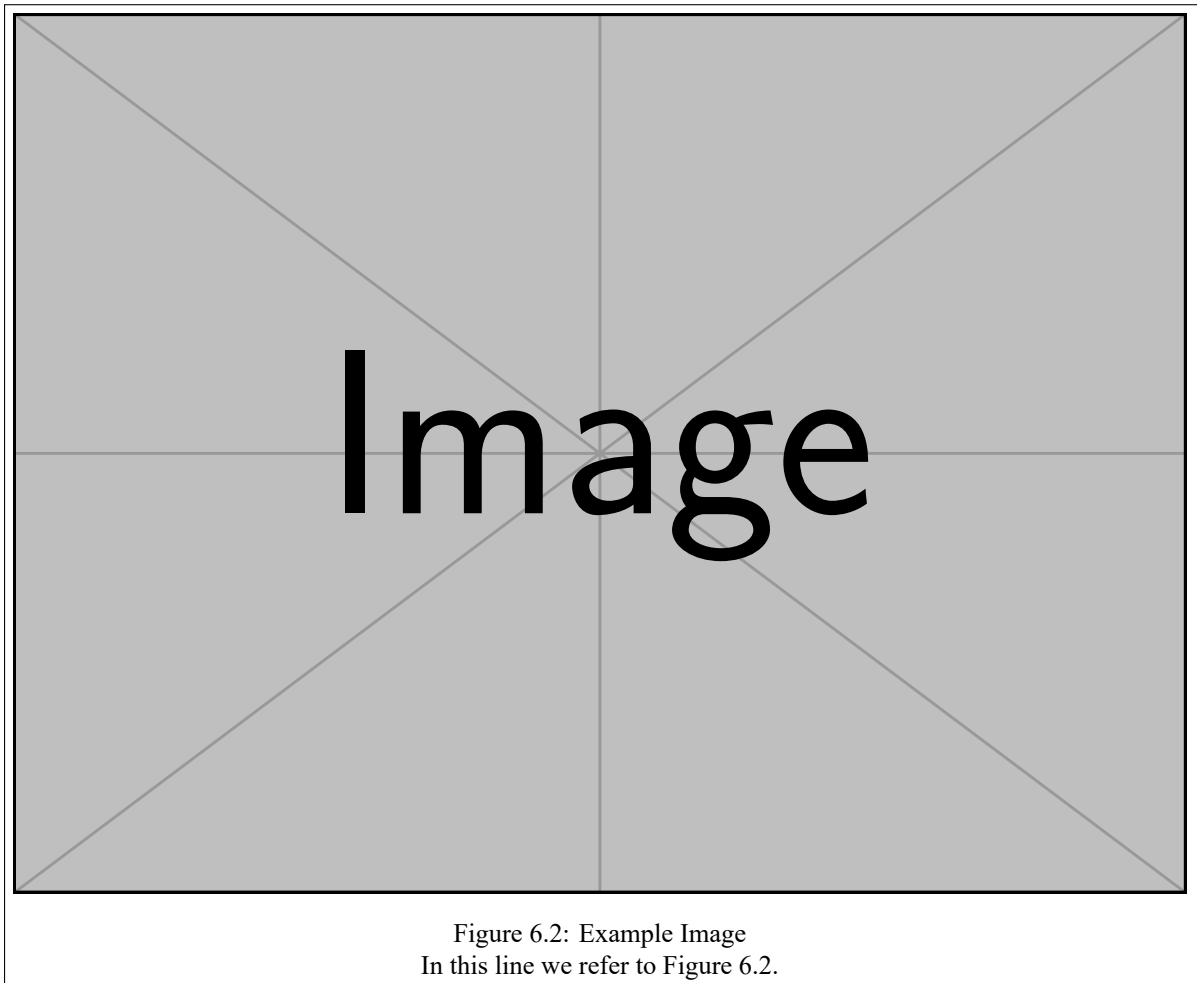
**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}
\usepackage{caption}
\usepackage{graphicx}

\begin{document}

\begin{figure}
  \centering
  \includegraphics[width=\textwidth]{example-image}
  \caption{Example Image}
  \label{fig:example}
\end{figure}
```

In this line we refer to Figure `\ref{fig:example}`.  
`\end{document}`

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

The command `\label{fig:example}` assigns an internal name to the figure and links it with the figure number generated by LaTeX.

`\label{fig:example}` कमांड चित्र को एक आंतरिक नाम प्रदान करती है और उसे LaTeX द्वारा दिए गए चित्र क्रमांक से जोड़ती है।

*The `\label` command should always be written immediately after the `\caption` command.*

`\label` कमांड को हमेशा `\caption` कमांड के ठीक नीचे लिखा जाना चाहिए।

## 6.2 Sizing and Resizing Figures / चित्र का आकार बदलना

In LaTeX, the size of a figure can be adjusted using options such as `scale`, `width`, and `height` in the `\includegraphics` command.

LaTeX में चित्र का आकार `scale`, `width` और `height` विकल्पों द्वारा बदला जा सकता है।

### 6.2.1 Using `scale` / `scale` का उपयोग

The `scale` option resizes the image proportionally.

`scale` विकल्प चित्र को समान अनुपात में बड़ा या छोटा करता है।

#### Input (LaTeX Code) / इनपुट (LaTeX कोड)

```
\documentclass{article}
\usepackage{caption}
\usepackage{graphicx}
\begin{document}
```

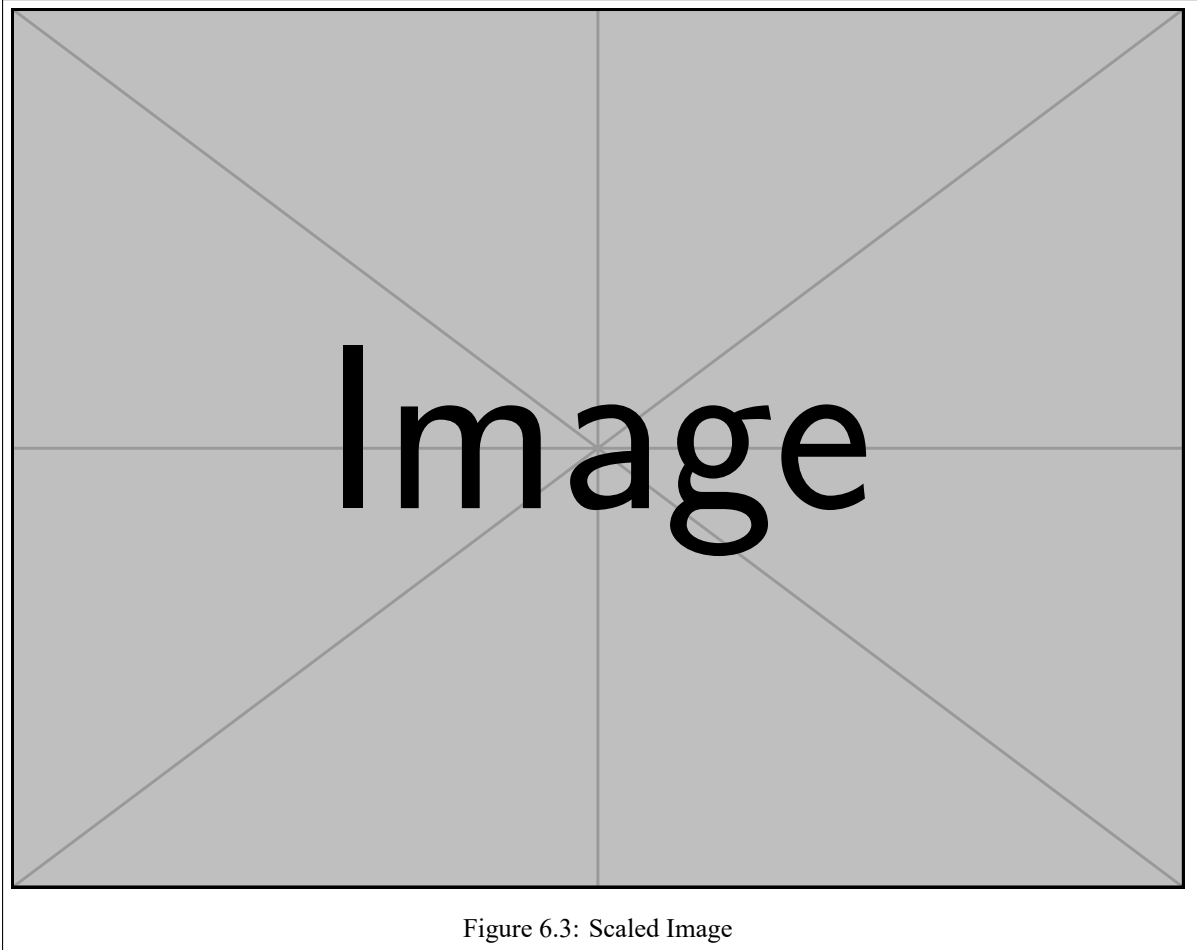
```

\begin{figure}
  \centering
  \includegraphics[width=\textwidth]{example-image}
  \caption{Scaled Image}
\end{figure}

\end{document}

```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**



The value 0.4 reduces the image to 40% of its original size.

0.4 मान चित्र को उसके मूल आकार के 40 प्रतिशत में बदल देता है।

### 6.2.2 Using width / width का उपयोग

The width option sets the width of the image relative to the text width.

width विकल्प चित्र की चौड़ाई को टेक्स्ट की चौड़ाई के अनुसार निर्धारित करता है।

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```

\documentclass{article}
\usepackage{caption}
\usepackage{graphicx}
\begin{document}

\begin{figure}
  \centering
  \includegraphics[width=\textwidth]{example-image}
  \caption{Image with Fixed Width}
\end{figure}

```

```
\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

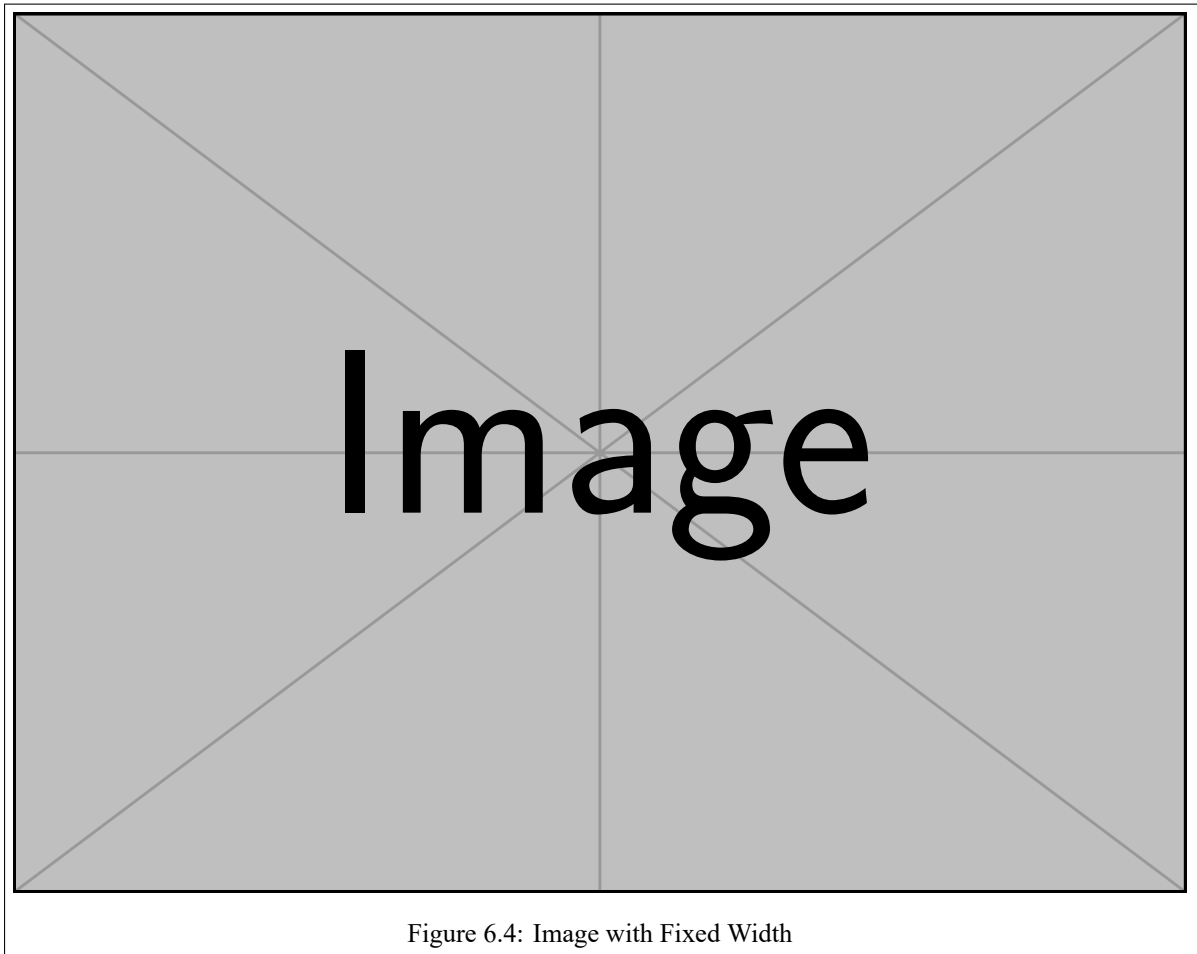


Figure 6.4: Image with Fixed Width

The option `width=\textwidth` sets the image width to half of the text width.

`width=0.5\textwidth` विकल्प चित्र की चौड़ाई को टेक्स्ट क्षेत्र के आधे के बराबर कर देता है।

### 6.2.3 Using `height` / `height` का उपयोग

The `height` option sets a fixed height for the image.

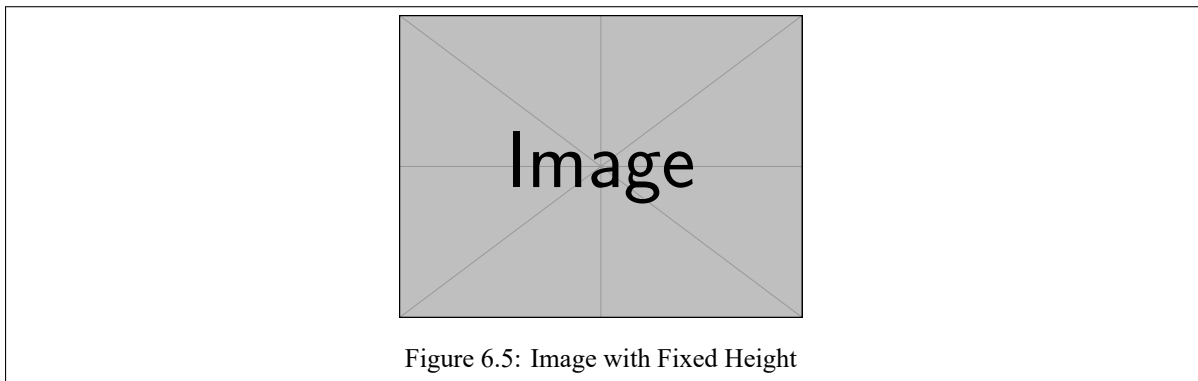
`height` विकल्प चित्र की ऊँचाई निर्धारित करता है।

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}
\usepackage{caption}
\usepackage{graphicx}
\begin{document}

\begin{figure}
  \centering
  \includegraphics[height=4cm]{example-image}
  \caption{Image with Fixed Height}
\end{figure}

\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

The value 4cm sets the image height to 4 centimeters.  
4cm मान चित्र की ऊँचाई 4 सेंटीमीटर निर्धारित करता है।

**6.2.4 Subfigures / उप-चित्र (Subfigures)**

To include subfigures, use the subcaption package and the subfigure environment:  
उप-चित्र जोड़ने के लिए subcaption पैकेज और subfigure एनवायरनमेंट का उपयोग किया जाता है:

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}
\usepackage{caption}
\usepackage{graphicx}
\usepackage{subcaption}
%\usepackage{subfigure}
\begin{document}

\begin{figure}[h]
\centering
\begin{subfigure}[b]{0.4\textwidth}
\includegraphics[width=\textwidth]{example-image-a}
\caption{Subfigure A}
\label{subfig:a}
\end{subfigure}

\begin{subfigure}[b]{0.4\textwidth}
\includegraphics[width=\textwidth]{example-image-b}
\caption{Subfigure B}
\label{subfig:b}
\end{subfigure}
\caption{Subfigures Example}
\label{fig:subfigures}
\end{figure}

\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)****6.3 Positioning Figures and Tables / चित्रों और सारणियों की स्थिति निर्धारित करना**

Figures and tables can be positioned using optional arguments in the figure and table environments. Positional arguments in LaTeX are optional arguments used to specify the preferred placement of figures and tables within the document. They provide instructions to LaTeX on where to place the elements relative to the text. चित्र और तालिकाओं की स्थिति figure और table एनवायरनमेंट में दिए गए वैकल्पिक आर्गुमेंट (optional arguments) के माध्यम से निर्धारित की जा सकती है। LaTeX में positional arguments ऐसे वैकल्पिक आर्गुमेंट होते हैं जो यह निर्धारित करते हैं कि चित्र और सारणी टेक्स्ट के सापेक्ष कहाँ दिखाई देंगे।

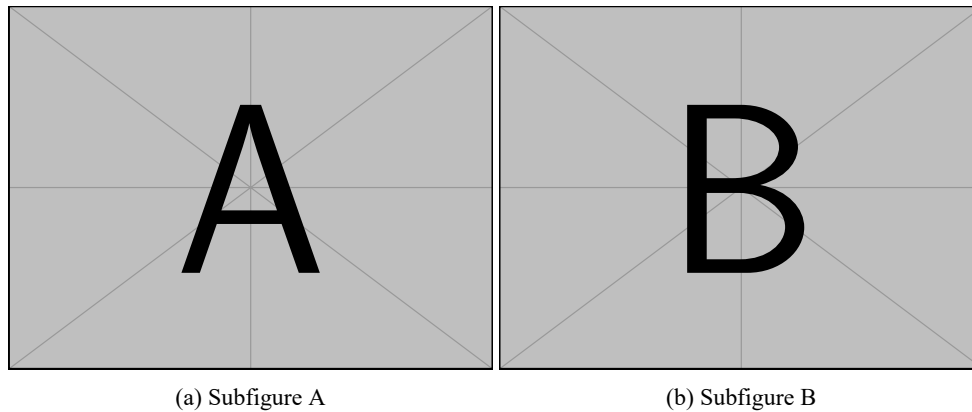


Figure 6.6: Subfigures Example

The common positional arguments are:

सामान्य रूप से प्रयुक्त पोज़िशनल आर्गुमेंट निम्नलिखित हैं:

1. **h**: Place the figure or table "here", i.e., as close as possible to the current position in the text.
2. **t**: Place the figure or table at the "top" of a page.
3. **b**: Place the figure or table at the "bottom" of a page.
4. **p**: Place the figure or table on a separate "page" containing only floats.

1. h: चित्र या तालिका को वर्तमान स्थान के ठीक पास रखने के लिए।
2. t: चित्र या तालिका को पेज के ऊपर रखने के लिए।
3. b: चित्र या तालिका को पेज के नीचे रखने के लिए।
4. p: चित्र या तालिका को केवल प्लोट वाली अलग पेज पर रखने के लिए।

These arguments can be combined to provide more specific placement instructions.

For example, [ht] specifies that LaTeX should try to place the figure or table either "here" or at the "top" of a page.

इन आर्गुमेंट को आपस में जोड़कर अधिक सटीक स्थिति निर्धारित की जा सकती है।

उदाहरण के लिए, [ht] का अर्थ है कि LaTeX पहले "यहाँ" रखने की कोशिश करेगा और यदि संभव न हो तो पेज के ऊपर रखेगा।

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}
\usepackage{caption}
\usepackage{graphicx}

\begin{document}

\begin{figure}[htbp]
  \centering
  \includegraphics[width=0.5\textwidth]{example-image}
  \caption{Example Image with Custom Positioning}
  \label{fig:custom}
\end{figure}

\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

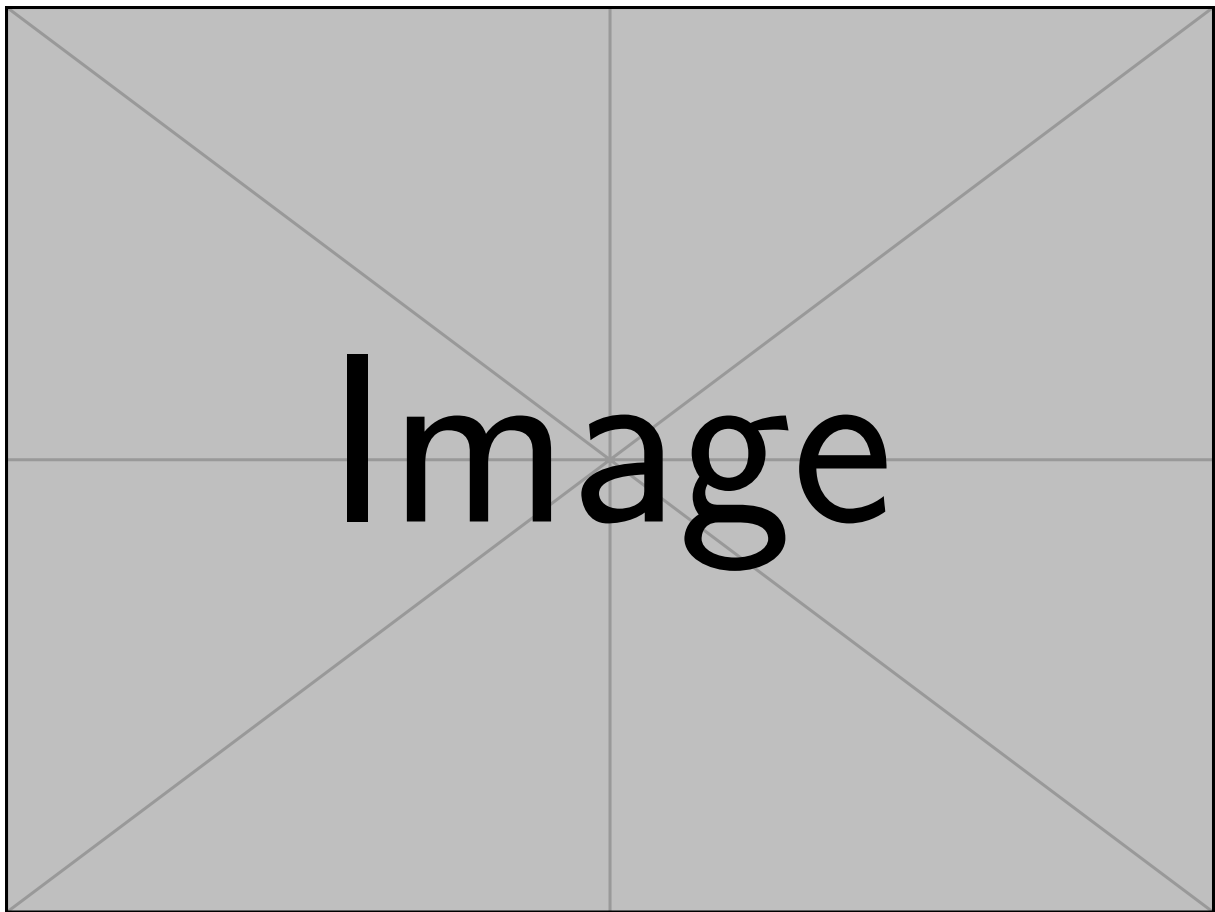


Figure 6.7: Example Image with Custom Positioning

## 6.4 Table and Tabular / टेबल और टैब्युलर

The `table` environment is used to create a floating table with caption and label, while the `tabular` environment is used to arrange data in rows and columns.

`table` एन्वायरनमेंट का उपयोग कैप्शन और लेबल के साथ फ्लोटिंग टेबल बनाने के लिए किया जाता है, जबकि `tabular` एन्वायरनमेंट का उपयोग पंक्तियों और स्तंभों में डेटा व्यवस्थित करने के लिए किया जाता है।

### 6.4.1 Using `tabular` / `tabular` का उपयोग

The `tabular` environment is responsible only for the table structure.

`tabular` एन्वायरनमेंट केवल टेबल की संरचना के लिए उत्तरदायी होता है।

#### Input (LaTeX Code) / इनपुट (LaTeX कोड)

```
\documentclass{article}

\begin{document}

\begin{tabular}{|c|c|c|}
\hline
S.No & Name & Marks \\
\hline
1 & A & 85 \\
2 & B & 90 \\
3 & C & 88 \\
\hline
\end{tabular}

\end{document}
```

#### Output (PDF Result) / आउटपुट (PDF परिणाम)

S.No	Name	Marks
1	A	85
2	B	90
3	C	88

The `tabular` environment creates rows and columns but does not provide caption or numbering.

`tabular` एन्वायरनमेंट पंक्तियाँ और स्तंभ बनाता है, लेकिन कैप्शन या क्रमांक प्रदान नहीं करता।

### 6.4.2 Using `table with tabular` / `table` के साथ `tabular` का उपयोग

The `table` environment is used to add caption, label, and automatic numbering to a table.

`table` एन्वायरनमेंट का उपयोग टेबल को कैप्शन, लेबल और स्वतः क्रमांकन देने के लिए किया जाता है।

#### Input (LaTeX Code) / इनपुट (LaTeX कोड)

```
\documentclass{article}

\begin{document}

\begin{table}
\centering
\begin{tabular}{|c|c|c|}
\hline
S.No & Name & Marks \\
\hline
1 & A & 85 \\
2 & B & 90 \\
3 & C & 88 \\
\hline
\end{tabular}
\caption{Student Marks}
\label{tab:marks}
\end{table}

\end{document}
```

```
\end{table}
\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

S.No	Name	Marks
1	A	85
2	B	90
3	C	88

Table 6.1: Student Marks

The `table` environment acts as a container for `tabular` and provides caption, label, and numbering. The `\label` command should always be written after the `\caption` command. `table` एन्वायरनमेंट `tabular` के लिए कंटेनर का कार्य करता है और कैप्शन, लेबल तथा क्रमांक प्रदान करता है। `\label` कमांड को हमेशा `\caption` कमांड के नीचे लिखा जाना चाहिए।

## 6.5 Table Formatting / तालिका स्वरूपण

### 6.5.1 Left Alignment in All Columns / सभी कॉलम में बायाँ संरेखण

The `l` column specifier aligns the content to the left.  
`l` कॉलम स्पेसिफायर सामग्री को बाईं ओर संरेखित करता है।

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}
\begin{document}
\begin{tabular}{|l|l|l|}
\hline
Name & Subject & Marks \\
\hline
A & Math & 85 \\
B & Physics & 90 \\
C & Chemistry & 88 \\
\hline
\end{tabular}
\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

Name	Subject	Marks
A	Math	85
B	Physics	90
C	Chemistry	88

All three columns use the `l` specifier, so the content in each column is left aligned.  
तीनों कॉलम में `l` का उपयोग किया गया है, इसलिए प्रत्येक कॉलम की सामग्री बाईं ओर संरेखित है।

### 6.5.2 Center Alignment: `c c c` / केंद्र संरेखण: `c c c`

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```

\begin{tabular}{|c|c|c|}
\hline
Name & Subject & Marks \\
\hline
A & Math & 85 \\
B & Physics & 90 \\
C & Chemistry & 88 \\
\hline
\end{tabular}

```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

Name	Subject	Marks
A	Math	85
B	Physics	90
C	Chemistry	88

All columns use c, so the content is center aligned.

सभी कॉलम में c का उपयोग किया गया है, इसलिए सामग्री केंद्र में संरेखित है।

### 6.5.3 Right Alignment: r r r / दायीं संरेखण: r r r

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```

\begin{tabular}{|r|r|r|}
\hline
Name & Subject & Marks \\
\hline
A & Math & 85 \\
B & Physics & 90 \\
C & Chemistry & 88 \\
\hline
\end{tabular}

```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

Name	Subject	Marks
A	Math	85
B	Physics	90
C	Chemistry	88

All columns use r, so the content is right aligned.

सभी कॉलम में r का उपयोग किया गया है, इसलिए सामग्री दाईं ओर संरेखित है।

### 6.5.4 Mixed Alignment: l c r / मिश्रित संरेखण: l c r

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```

\documentclass{article}

\begin{document}

\begin{tabular}{|l|c|r|}
\hline
Name & Subject & Marks \\
\hline
A & Math & 85 \\
B & Physics & 90 \\
C & Chemistry & 88 \\
\hline
\end{tabular}

\end{document}

```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

	Name	Subject	Marks
	A	Math	85
	B	Physics	90
	C	Chemistry	88

The first column is left aligned, the second is center aligned, and the third is right aligned.

पहला कॉलम बाईं ओर, दूसरा केंद्र में और तीसरा दाईं ओर संरेखित है।

**6.5.5 Paragraph Columns: p{} / p{} कॉलम**

The p{} column type is used to set a fixed column width and allows text wrapping.

p{} कॉलम प्रकार निश्चित चौड़ाई निर्धारित करता है और टेक्स्ट को स्वतः अगली पंक्ति में ले जाता है।

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}

\begin{document}

\begin{tabular}{|p{3cm}|p{4cm}|}
\hline
Name & Description \\
\hline
A & Excellent performance in mathematics subject \\
B & Strong understanding of physics concepts \\
\hline
\end{tabular}

\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

	Name	Description
	A	Excellent performance in mathematics subject
	B	Strong understanding of physics concepts

The p{} column allows fixed width and automatic text wrapping.

p{} कॉलम निश्चित चौड़ाई प्रदान करता है और टेक्स्ट को स्वतः लपेटता है।

**6.5.6 \hline / \hline**

The \hline command is used in the tabular environment to draw a horizontal line across the table. \hline कमांड का उपयोग tabular एन्वायरनमेंट में टेबल के अंदर क्षैतिज रेखा खींचने के लिए किया जाता है।

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}

\begin{document}

\begin{tabular}{|l|c|r|}
\hline
Name & Subject & Marks \\
\hline
A & Math & 85 \\
B & Physics & 90 \\
C & Chemistry & 88 \\
\hline
\end{tabular}

\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

Name	Subject	Marks
A	Math	85
B	Physics	90
C	Chemistry	88

The `\hline` command draws a horizontal line above the current row. It is commonly used:

- after `\begin{tabular}` to draw the top border,
- after the header row to separate headings from data,
- before `\end{tabular}` to draw the bottom border.

`\hline` कमांड वर्तमान पंक्ति के ऊपर एक क्षैतिज रेखा बनाती है। इसका उपयोग सामान्यतः:

- टेबल की ऊपरी सीमा बनाने के लिए,
- हेडिंग और डेटा को अलग करने के लिए,
- टेबल की निचली सीमा बनाने के लिए किया जाता है।

**6.5.7 Resizing Table / टेबल का आकार बदलना**

Sometimes a table is wider than the text area. In such cases, the table size can be adjusted using resizing commands. कभी-कभी टेबल की चौड़ाई टेक्स्ट क्षेत्र से अधिक हो जाती है। ऐसे मामलों में टेबल का आकार बदलने के लिए `resizing` कमांड का उपयोग किया जाता है।

The `\resizebox` command is used to resize a table to a specified width and height.

`\resizebox` कमांड का उपयोग टेबल की चौड़ाई और ऊँचाई निर्धारित करने के लिए किया जाता है।

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```

\documentclass{article}
\usepackage{graphicx} % REQUIRED for \resizebox

\begin{document}

\begin{table}
\centering
\resizebox{0.8\textwidth}{!}{
\begin{tabular}{|l|c|r|}
\hline
Name & Subject & Marks \\
\hline
Aditi & Math & 85 \\
Rahul & Physics & 90 \\
Neha & Chemistry & 88 \\
\hline
\end{tabular}
}
\caption{Resized Table}
\end{table}

\end{document}

```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

Name	Subject	Marks
Aditi	Math	85
Rahul	Physics	90
Neha	Chemistry	88

Table 6.2: Resized Table

The value `0.8\textwidth` resizes the table to 80% of the text width while keeping the height proportional. `0.8\textwidth` मान टेबल की चौड़ाई को टेक्स्ट क्षेत्र के 80% तक सीमित कर देता है तथा ऊँचाई स्वतः समायोजित होती है।

The symbol `!` preserves the aspect ratio of the table. `!` चिन्ह टेबल के अनुपात (aspect ratio) को बनाए रखता है।

**6.6 Multi-row and Multi-column / मल्टी-रो और मल्टी-कॉलम**

You can use the `table` and `tabular` environments to create a structured table with caption and numbering. The `\multirow` command, provided by the `multirow` package, is used to span a single cell across multiple rows. The `\multicolumn` command is used to merge multiple columns into a single cell with specified alignment. The `\cline` command is used to draw a horizontal line across selected columns. The `\caption` command adds a title to the table, and the `\label` command assigns a unique identifier to the table.

आप कैप्शन और क्रमांकन के साथ संरचित टेबल बनाने के लिए `table` और `tabular` एन्वायरनमेंट का उपयोग कर सकते हैं। `multirow` पैकेज द्वारा प्रदान की गई `\multirow` कमांड का उपयोग एक सेल को कई पंक्तियों में फैलाने के लिए किया जाता है। `\multicolumn` कमांड का उपयोग कई कॉलम को एक ही सेल में जोड़ने के लिए किया जाता है। `\cline` कमांड चयनित कॉलम के नीचे क्षैतिज रेखा खींचने के लिए उपयोग की जाती है। `\caption` कमांड टेबल को शीर्षक प्रदान करती है और `\label` कमांड टेबल को एक विशिष्ट पहचान देती है।

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}
\usepackage{multirow} % REQUIRED for \multirow

\begin{document}

\begin{table}[h]
  \centering
  \begin{tabular}{|c|c|c|}
    \hline
    \multirow{2}{*}{Name} & \multicolumn{2}{c|}{Scores} \\
    \cline{2-3}
    & Math & Physics \\
    \hline
    Alice & 90 & 85 \\
    Bob & 75 & 92 \\
    \hline
  \end{tabular}
  \caption{Multi-row and Multi-column Table}
  \label{tab:multi}
\end{table}

\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

Name	Scores	
	Math	Physics
Alice	90	85
Bob	75	92

Table 6.3: Multi-row and Multi-column Table

# 7. References and Citations / संदर्भ और उद्धरण

References and citations play a crucial role in technical and academic documents. They help acknowledge the original sources of information, maintain academic integrity, and allow readers to locate the referenced material easily. In LaTeX, references and citations are managed systematically, ensuring consistency, accuracy, and automatic formatting throughout the document.

References और Citations तकनीकी तथा शैक्षणिक दस्तावेजों का एक महत्वपूर्ण भाग होते हैं। इनके माध्यम से उपयोग किए गए मूल स्रोतों को उचित श्रेय दिया जाता है, शैक्षणिक ईमानदारी बनाए रखी जाती है तथा पाठकों को संदर्भित सामग्री तक आसानी से पहुँचने में सहायता मिलती है। LaTeX में References और Citations को व्यवस्थित तरीके से प्रबंधित किया जाता है, जिससे पूरे दस्तावेज़ में स्वचालित, सुसंगत और त्रुटिरहित संदर्भ प्रस्तुत किए जा सकते हैं।

## 7.1 Cross-referencing

### 7.1.1 Cross-referencing Sections / अनुभागों का क्रॉस-रेफरेंस

Labeling and referencing sections is a common practice to maintain consistency in your document.

Use the `\label` command to assign a label to a section, and `\ref` to reference it:

डॉक्यूमेंट में समानता और स्पष्टता बनाए रखने के लिए अनुभागों को लेबल करना और संदर्भित करना एक सामान्य प्रक्रिया है।

किसी अनुभाग का लेबल बनाने के लिए `\label` और उसे संदर्भित करने के लिए `\ref` कमांड का उपयोग करें:

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}
\begin{document}

\section{First Section}\label{sec:first}
  Here is the content of first section.

\section{Second Section}
  Here is the content of Second section where I am referring First Section \ref{sec:first}
}

\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**

**Output**

## 1 First Section

Here is the content of first section.

## 2 Second Section

Here is the content of Second section where I am referring First Section 1

### 7.1.2 Cross-referencing Figures / चित्रों का क्रॉस-रेफरेंस

Similarly, you can label and reference figures. Use `\label` to assign a label to a figure, and `\ref` to reference it: इसी प्रकार, आप चित्रों को भी लेबल कर सकते हैं और उनका संदर्भ दे सकते हैं। चित्र का लेबल बनाने के लिए `\label` और उसे संदर्भित करने के लिए `\ref` कमांड का उपयोग करें:

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{article}
\usepackage{graphicx}

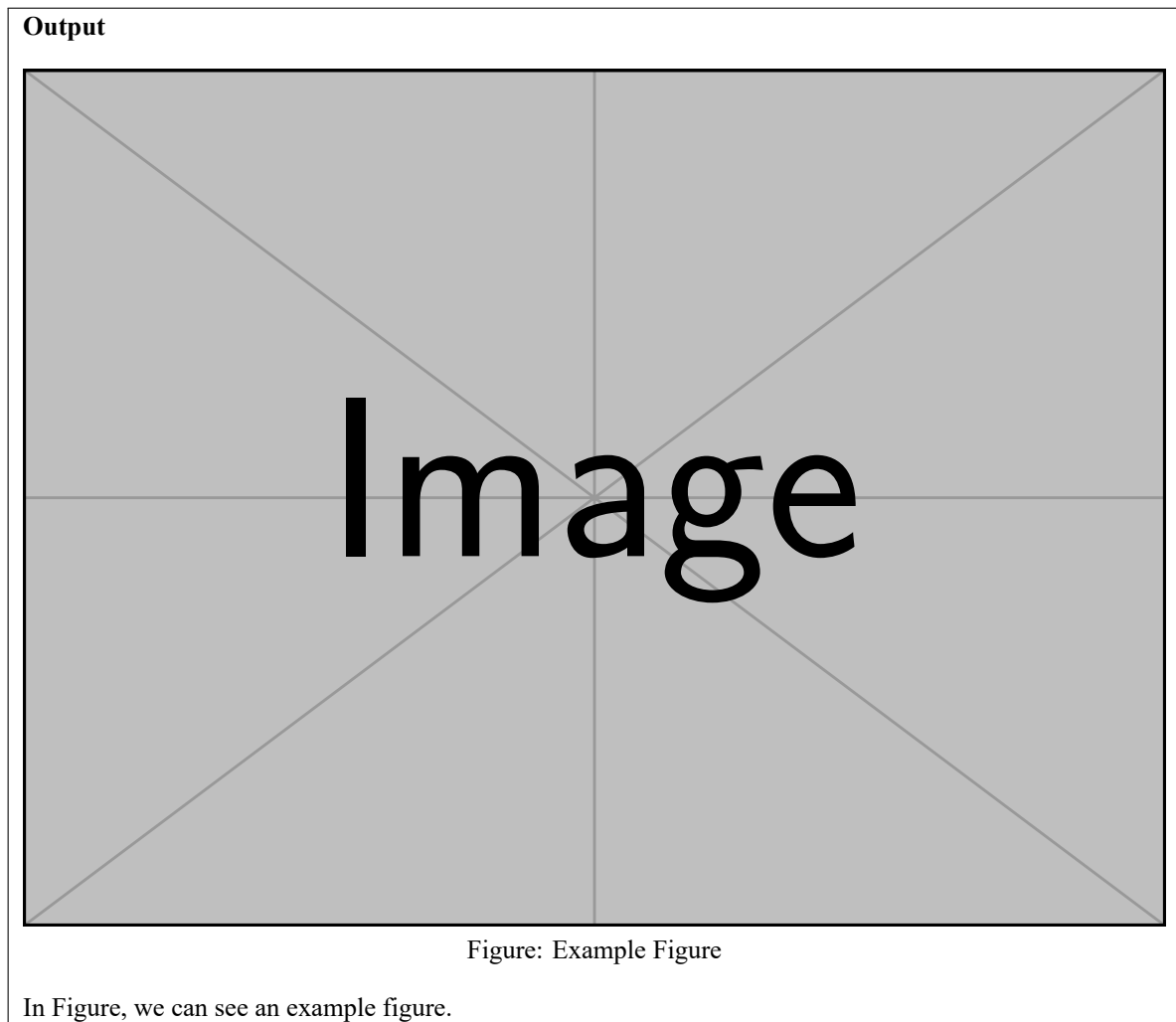
\begin{document}

  \begin{figure}
    \centering
    \includegraphics[width=\linewidth]{example-image}
    \caption{Example Figure}\label{fig:example}
  \end{figure}
```

In Figure~\ref{fig:example}, we can see an example figure.

```
\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**



## 7.2 Citation / उद्धरण

Understanding references and citations in LaTeX requires clarity about three different components: reference data storage, bibliography formatting, and citation control. These components work together to produce correctly formatted citations and reference lists.

LaTeX में संदर्भ और उद्धरण को सही ढंग से समझने के लिए तीन मुख्य घटकों को जानना आवश्यक है: संदर्भ डेटा का संग्रहण, संदर्भ सूची का फॉर्मेट, और दस्तावेज़ के भीतर उद्धरणों का नियंत्रण। ये तीनों घटक मिलकर उचित उद्धरण और संदर्भ सूची तैयार करते हैं।

### 7.2.1 Step 1: Reference Data Using BibTeX / चरण 1: BibTeX द्वारा संदर्भ डेटा

BibTeX is a reference management system used to store bibliographic information in a separate .bib file. This file contains details such as author names, title of the work, publication year, and journal or book information. BibTeX itself does not decide how references look; it only supplies the stored data to LaTeX.

BibTeX एक संदर्भ प्रबंधन प्रणाली है, जिसका उपयोग .bib फ़ाइल में बिलियोग्राफ़िक जानकारी संग्रहित करने के लिए किया जाता है। इस फ़ाइल में लेखक का नाम, शीर्षक, प्रकाशन वर्ष, और जर्नल या पुस्तक से संबंधित जानकारी होती है। BibTeX स्वयं संदर्भों का रूप निर्धारित नहीं करता, बल्कि केवल डेटा LaTeX को प्रदान करता है।

### 7.2.2 Step 2: Bibliography Styles / चरण 2: bibliography शैलियाँ

Bibliography styles define how the reference list appears in the final document. Common styles such as plain, unsrt, and ieetr control the ordering, numbering, and formatting of references. These styles are selected using the

`\bibliographystyle` command.

bibliography शैलियाँ यह निर्धारित करती हैं कि अंतिम दस्तावेज़ में संदर्भ सूची किस प्रकार दिखाई देगी। plain, unsrt, और ieetr जैसी शैलियाँ संदर्भों के क्रम, संख्या और फॉर्मेट को नियंत्रित करती हैं। इन शैलियों का चयन `\bibliographystyle` कमांड द्वारा किया जाता है।

LaTeX provides several predefined bibliography styles that control how references are formatted in the document. LaTeX में कई पूर्वनिर्धारित संदर्भ शैलियाँ उपलब्ध होती हैं, जो यह निर्धारित करती हैं कि संदर्भ सूची किस प्रकार प्रदर्शित होगी।

Some commonly used `\bibliographystyle` options are listed below.

नीचे कुछ सामान्य रूप से प्रयुक्त `\bibliographystyle` विकल्प दिए गए हैं:

- `plain` — Produces numeric citations with references sorted alphabetically by author name. संख्यात्मक उद्धरण प्रदान करता है तथा संदर्भों को लेखक के नाम के अनुसार वर्णानुक्रम में व्यवस्थित करता है।
- `unsrt` — Produces numeric citations with references listed in the order of citation. संख्यात्मक उद्धरण प्रदान करता है और संदर्भों को उद्धरण के क्रम में सूचीबद्ध करता है।
- `alpha` — Uses alphanumeric labels based on author name and year. लेखक के नाम और वर्ष के आधार पर अल्फ़ान्यूमेरिक लेबल का उपयोग करता है।
- `abbrv` — Similar to `plain`, but uses abbreviated author names and journal titles. `plain` के समान होता है, लेकिन लेखक के नाम और जर्नल शीर्षक संक्षिप्त रूप में प्रदर्शित करता है।
- `ieetr` — Formats references according to IEEE standards and is widely used in engineering publications. IEEE मानकों के अनुसार संदर्भों को फॉर्मेट करता है और अभियान्त्रिकी प्रकाशनों में व्यापक रूप से उपयोग किया जाता है।
- `plainnat` — Used with `natbib` for author–year citation style. `natbib` के साथ लेखक–वर्ष उद्धरण शैली के लिए उपयोग किया जाता है।
- `apalike` — Produces author–year style citations similar to APA or Harvard format. APA या Harvard शैली के समान लेखक–वर्ष उद्धरण प्रदान करता है।

The choice of bibliography style depends on the document type and publication requirements.

संदर्भ शैली का चयन दस्तावेज़ के प्रकार और प्रकाशन की आवश्यकताओं पर निर्भर करता है।

### 7.2.3 Step 3: Citation Control Packages / चरण 3: उद्धरण नियंत्रण पैकेज

Citation control packages determine how citations appear within the document text. Packages such as `cite` improve numeric citations by sorting and compressing citation numbers, while `natbib` provides advanced commands for both numeric and author–year citation styles. These packages work together with BibTeX and bibliography styles to control in-text citations.

उद्धरण नियंत्रण पैकेज यह निर्धारित करते हैं कि दस्तावेज़ के भीतर उद्धरण किस प्रकार प्रदर्शित होंगे। `cite` पैकेज संख्यात्मक उद्धरणों को क्रमबद्ध और संक्षिप्त बनाता है, जबकि `natbib` पैकेज संख्यात्मक तथा लेखक--वर्ष उद्धरणों के लिए उन्नत कमांड प्रदान करता है। ये पैकेज BibTeX और bibliography शैलियों के साथ मिलकर इन-टेक्स्ट उद्धरणों को नियंत्रित करते हैं।

In addition to `cite` and `natbib`, several other packages are available in LaTeX that control how citations appear in the document.

`cite` और `natbib` के अतिरिक्त, LaTeX में कई अन्य पैकेज भी उपलब्ध हैं, जो दस्तावेज़ में उद्धरणों की प्रस्तुति को नियंत्रित करते हैं।

1. `cite` : Improves numeric citations by sorting and compressing citation numbers.

संख्यात्मक उद्धरणों को क्रमबद्ध करता है तथा उद्धरण संख्याओं को संक्षिप्त रूप में प्रदर्शित करता है।

2. `natbib` : Supports numeric as well as author–year citations and provides commands such as `\citet{}` and `\citep{}`.

संख्यात्मक तथा लेखक--वर्ष उद्धरण शैलियों का समर्थन करता है और `\citet{}` तथा `\citep{}` जैसी कमांड प्रदान करता है।

3. `apacite` : Designed specifically for APA citation style and commonly used in psychology and social sciences.

APA उद्धरण शैली के लिए विशेष रूप से बनाया गया पैकेज, जिसका उपयोग मनोविज्ञान और सामाजिक विज्ञान में किया जाता है।

4. `harvard` : Provides author–year citation commands and is used for Harvard-style referencing.

Harvard शैली के अनुसार लेखक--वर्ष उद्धरण कमांड प्रदान करता है।

5. `chicago` : Supports Chicago-style citations, including note-based and author–year formats.

Chicago शैली के उद्धरणों का समर्थन करता है, जिसमें नोट-आधारित तथा लेखक--वर्ष दोनों फॉर्मेट शामिल हैं।

6. `jurabib` : A specialized citation package for legal documents with complex citation rules.

कानूनी दस्तावेज़ों के लिए विशेष उद्धरण पैकेज, जो जटिल कानूनी उद्धरण नियमों का समर्थन करता है।

### 7.2.4 Overall Flow of Citation Processing / उद्धरण प्रक्रिया का समग्र प्रवाह

The overall flow of reference and citation processing in LaTeX can be summarized as follows:

1. References are stored in a `.bib` file using BibTeX.
2. BibTeX supplies the reference data to LaTeX.
3. A bibliography style formats the reference list.
4. Citation packages control how citations appear in the text.

LaTeX में संदर्भ और उद्धरण की पूरी प्रक्रिया को निम्नलिखित रूप में समझा जा सकता है:

1. संदर्भों को BibTeX द्वारा `.bib` फ़ाइल में संग्रहित किया जाता है।
2. BibTeX यह डेटा LaTeX को प्रदान करता है।
3. bibliography शैली संदर्भ सूची को फॉर्मेट करती है।
4. उद्धरण नियंत्रण पैकेज दस्तावेज़ के भीतर उद्धरणों को नियंत्रित करते हैं।

BibTeX, bibliography styles, and citation control packages each have a distinct role. Confusion arises when these components are mixed without understanding their purpose. Students should remember that BibTeX stores reference data, bibliography styles define the appearance of references, and citation packages control how citations are displayed within the text.

BibTeX, bibliography शैलियाँ और उद्धरण नियंत्रण पैकेज तीनों की भूमिका अलग-अलग होती है। इन घटकों के उद्देश्य को समझे बिना उन्हें मिलाने से भ्रम उत्पन्न होता है। विद्यार्थियों को यह याद रखना चाहिए कि BibTeX संदर्भ डेटा संग्रहित करता है, bibliography शैलियाँ संदर्भों का रूप निर्धारित करती हैं, और उद्धरण नियंत्रण पैकेज दस्तावेज़ के भीतर उद्धरणों की प्रस्तुति को नियंत्रित करते हैं।

## 7.3 Sources and Methods for Citation in LaTeX / LaTeX में उद्धरण के स्रोत और विधियाँ

### 7.3.1 Citing Sources Using Google Scholar / Google Scholar का उपयोग करके उद्धरण

Google Scholar is one of the most commonly used platforms for searching academic articles and generating citations. It allows users to search research papers using keywords and provides citation details in multiple formats, including BibTeX. This makes Google Scholar a convenient and reliable starting point for creating references in LaTeX documents.

Google Scholar अकादमिक लेखों को खोजने और उद्धरण तैयार करने के लिए सबसे अधिक उपयोग किए जाने वाले प्लेटफॉर्म में से एक है। यह उपयोगकर्ताओं को कीवर्ड्स के माध्यम से शोध पत्र खोजने की सुविधा देता है और BibTeX सहित विभिन्न फॉर्मेट में उद्धरण विवरण प्रदान करता है। इस कारण LaTeX दस्तावेज़ों के लिए संदर्भ तैयार करने का यह एक सुविधाजनक और विश्वसनीय प्रारंभिक माध्यम है।

Now let us understand the procedure as shown in the following steps.

अब निम्नलिखित चरणों के माध्यम से इस प्रक्रिया को समझते हैं।

#### Step 1: Opening LaTeX Folder / चरण 1: LaTeX फ़ोल्डर खोलना

Open the LaTeX folder in which you are working. As shown in Figure 7.1, a folder named LaTeX Example is opened by double-clicking on it.

उस LaTeX फ़ोल्डर को खोलें जिसमें आप कार्य कर रहे हैं। Figure 7.1 में LaTeX Example नामक फ़ोल्डर को डबल-क्लिक करके खोलते हुए दिखाया गया है।

#### Step 2: Creating a New Text File / चरण 2: नयी टेक्स्ट फ़ाइल बनाना

After opening the LaTeX folder, right-click inside the folder, move the cursor to New, and then click on Text Document, as shown in Figure 7.2. You can use any text editor of your choice to create a text file. In this example, the Notepad text editor is used.

LaTeX फ़ोल्डर खोलने के बाद, फ़ोल्डर के अंदर राइट-क्लिक करें। इसके बाद कर्सर को New विकल्प पर ले जाएँ और फिर Text Document पर क्लिक करें, जैसा कि Figure 7.2 में दर्शाया गया है। Text फ़ाइल बनाने के लिए आप अपनी पसंद के किसी भी टेक्स्ट एडिटर का उपयोग कर सकते हैं। इस उदाहरण में Notepad टेक्स्ट एडिटर का उपयोग किया गया है।

#### Step 3: Renaming the Text File / चरण 3: टेक्स्ट फ़ाइल का नाम बदलना

As illustrated in Figure 7.3, a text file named New Text Document.txt is created, where .txt is the file extension. Now, rename this file according to your choice. For simplicity, the file name is changed to reference. After selecting the file name, change the file extension from .txt to .bib. While changing the file name and file extension, a warning prompt box appears, as shown in Figure 7.4. After confirming the change, the bibliography file is successfully created with the name reference.bib, as shown in Figure 7.5.

Figure 7.3 में दर्शाए अनुसार, New Text Document.txt नाम की एक टेक्स्ट फ़ाइल बनाई जाती है, जिसमें .txt फ़ाइल एक्सटेंशन होता है। अब इस फ़ाइल का नाम अपनी आवश्यकता के अनुसार बदलें। सरलता के लिए, फ़ाइल का नाम reference रखा गया है। फ़ाइल का नाम चुनने के बाद, फ़ाइल एक्सटेंशन को .txt से बदलकर .bib कर दें। फ़ाइल का नाम और एक्सटेंशन बदलते समय एक चेतावनी प्रॉम्प्ट बॉक्स दिखाई देता है, जैसा कि Figure 7.4 में दिखाया गया है। परिवर्तन की पुष्टि करने के बाद, reference.bib नाम की bibliography फ़ाइल सफलतापूर्वक बन जाती है, जैसा कि Figure 7.5 में दर्शाया गया है।

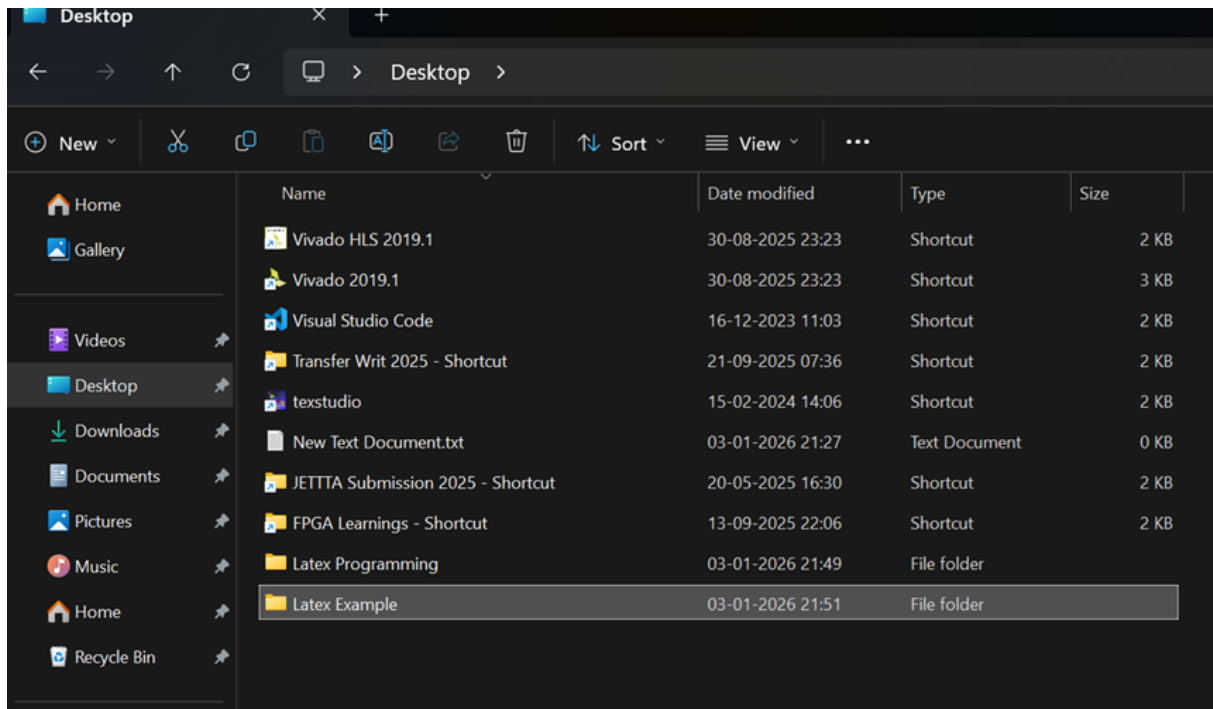


Figure 7.1: Latex Folder

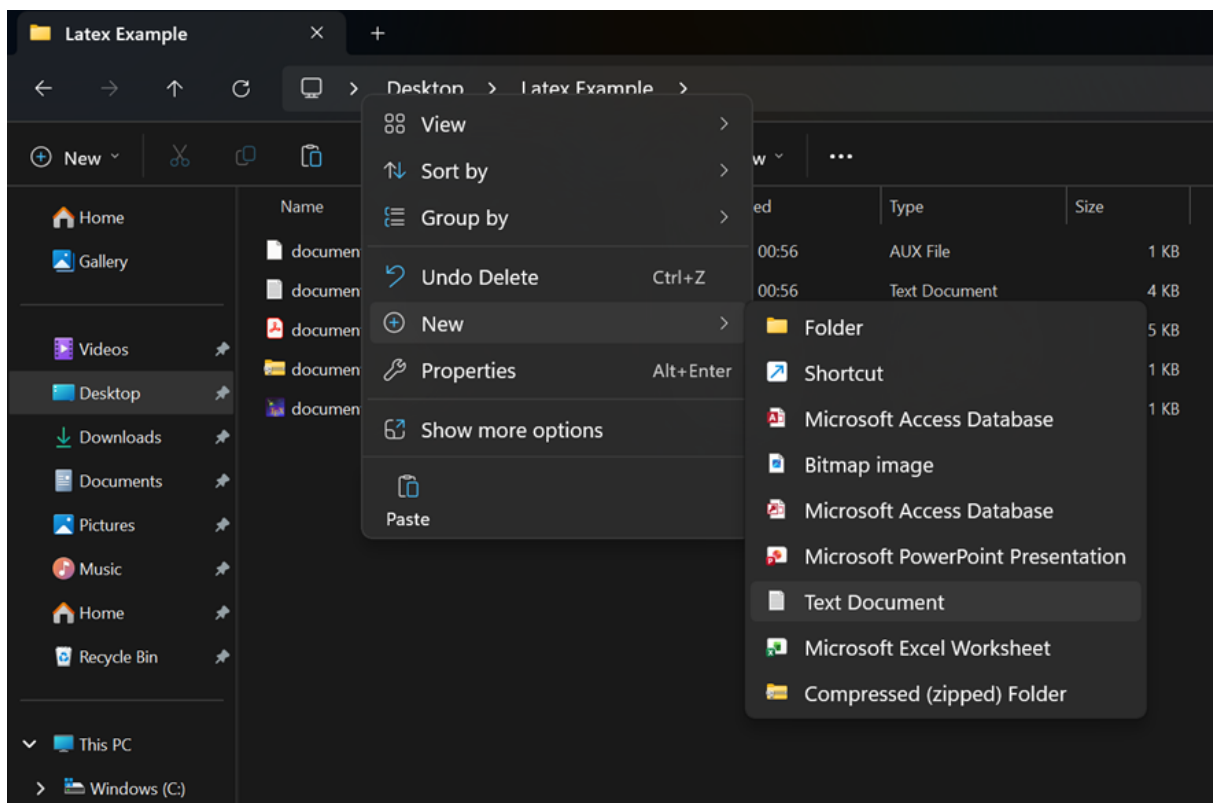


Figure 7.2: Creating a New Text File

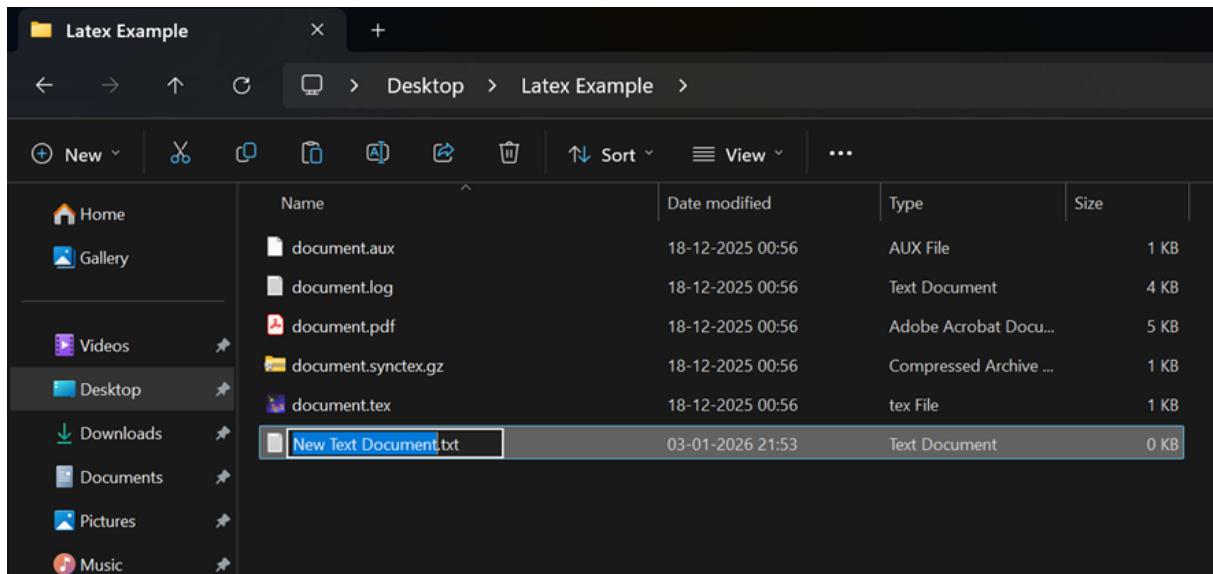


Figure 7.3: Renaming the Text File

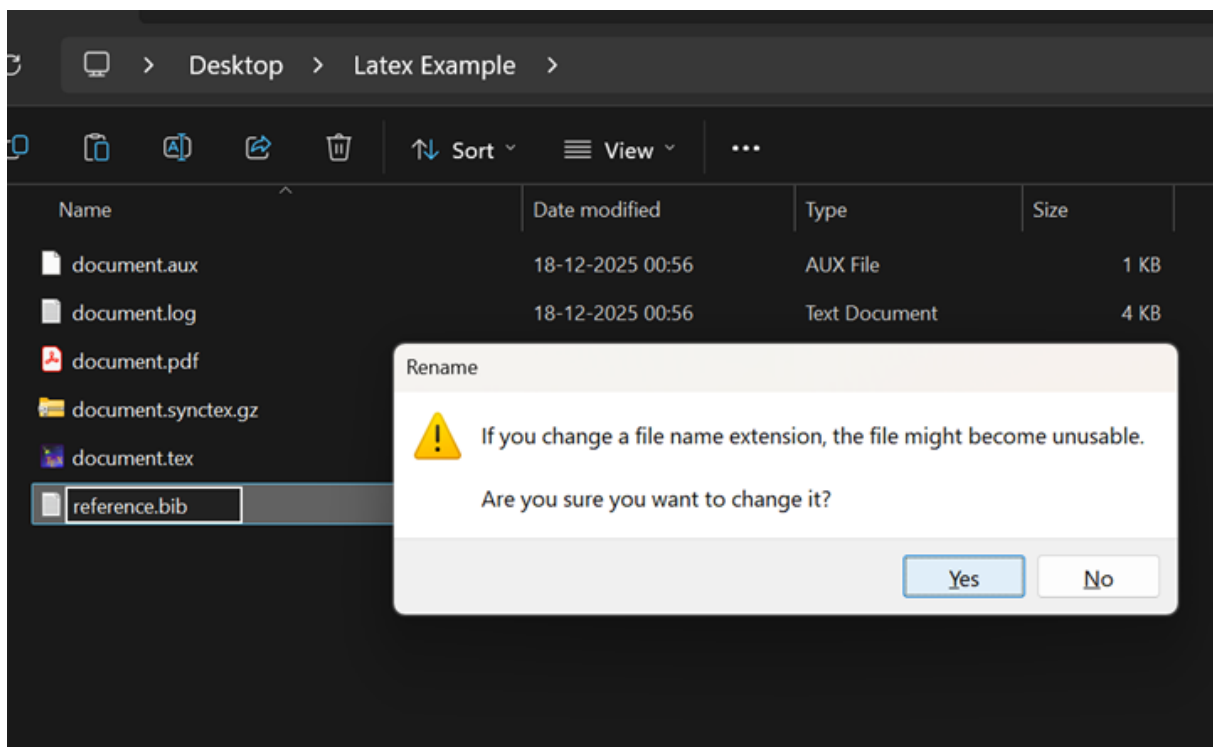


Figure 7.4: Changing the Name of File

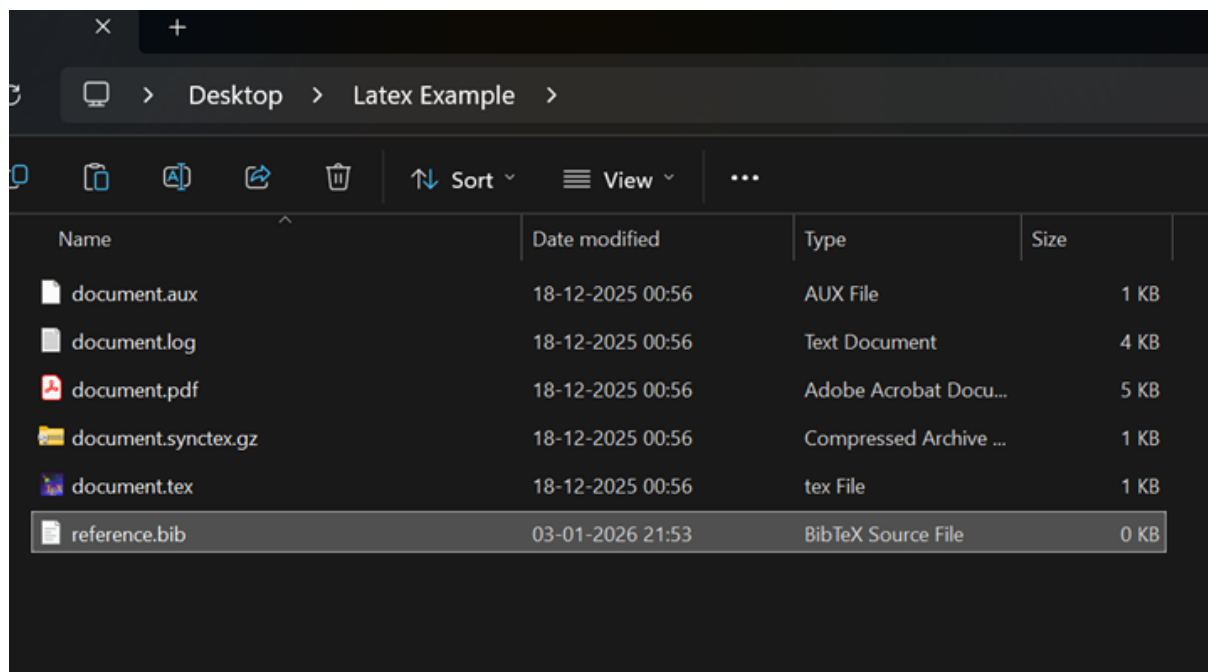


Figure 7.5: Creation of reference.bib

**Step 4: Searching the Article on Google Scholar / चरण 4: Google Scholar पर लेख खोजना**

As shown in Figure 7.6, the citation process begins by searching the required research article on Google Scholar using suitable keywords. The search results display the article title, authors, publication source, and year, helping the author to locate the correct reference.

Figure 7.6 में दर्शाए अनुसार, उद्धरण प्रक्रिया की शुरुआत Google Scholar पर आवश्यक शोध लेख को उपयुक्त कीवर्ड्स का उपयोग करके खोजने से होती है। खोज परिणामों में लेख का शीर्षक, लेखक, प्रकाशन स्रोत और प्रकाशन वर्ष प्रदर्शित होते हैं, जिससे लेखक को सही संदर्भ पहचानने में सहायता मिलती है।

**Step 5: Accessing Citation Options / चरण 5: उद्धरण विकल्पों तक पहुँचना**

In Figure 7.7, the Cite button associated with the selected article is shown. Clicking this button opens a citation dialog that provides different citation formats and export options.

Figure 7.7 में चयनित लेख से संबंधित Cite बटन को दर्शाया गया है। इस बटन पर क्लिक करने से एक उद्धरण संवाद बॉक्स खुलता है, जो विभिन्न उद्धरण फॉर्मेट और निर्यात (export) विकल्प प्रदान करता है।

**Step 6: Viewing Available Citation Formats / चरण 6: उपलब्ध उद्धरण फॉर्मेट देखना**

Figure 7.8 shows the citation dialog window generated by Google Scholar. This dialog lists commonly used citation formats such as MLA, APA, Chicago, Harvard, and Vancouver. It also provides a BibTeX option, which is most suitable for LaTeX documents.

Figure 7.8 में Google Scholar द्वारा उत्पन्न किया गया उद्धरण संवाद विंडो दिखाया गया है। इस संवाद बॉक्स में सामान्य रूप से उपयोग की जाने वाली उद्धरण शैलियाँ जैसे MLA, APA, Chicago, Harvard, और Vancouver सूचीबद्ध होती हैं। इसके साथ ही इसमें BibTeX विकल्प भी उपलब्ध होता है, जो LaTeX दस्तावेज़ों के लिए सबसे उपयुक्त है।

**Step 7: Generating the BibTeX Entry / चरण 7: BibTeX प्रविष्टि तैयार करना**

As illustrated in Figure 7.9, selecting the BibTeX option opens a new page displaying a structured BibTeX entry. This entry includes essential bibliographic fields such as author, title, journal, volume, pages, and year. The BibTeX code can be copied directly from this page.

Figure 7.9 में दर्शाए अनुसार, BibTeX विकल्प का चयन करने पर एक नया पृष्ठ खुलता है, जिसमें एक संरचित BibTeX प्रविष्टि प्रदर्शित होती है। इस प्रविष्टि में लेखक, शीर्षक, जर्नल, वॉल्यूम, पृष्ठ संख्या और प्रकाशन वर्ष जैसे आवश्यक बिलियोग्राफ़िक फ़ील्ड शामिल होते हैं। इस पृष्ठ से BibTeX कोड को सीधे कॉपी किया जा सकता है।

The screenshot shows a Google Scholar search page. The search query is 'incomplete testing of SoC'. The results list three articles. The first article, 'Incomplete Testing of SOC' by KM Singh, J Deka, and S Biswas, is highlighted. The 'Cite' button for this article is circled in red. The left sidebar contains filters for 'Any time', 'Sort by relevance', 'Any type', and 'Create alert'.

Articles About 18,70,000 results (0.12 sec)

**Any time**  
 Since 2026  
 Since 2025  
 Since 2022  
 Custom range...

**Sort by relevance**  
 Sort by date

**Any type**  
 Review articles

include patents  
 include citations  
 Create alert

**Incomplete Testing of SOC**  
 KM Singh, J Deka, S Biswas - Journal of Electronic Testing, 2023 - Springer  
 ... to come up with an efficient **testing** strategy that can help reduce the costs. ... **testing** the **SOC** incompletely instead of **testing** it thoroughly. This method involves compromising on the **test** ...  
 ☆ Save Cite Related articles All 5 versions

**ATPG for Incomplete Testing of SOC Considering Bridging Faults**  
 KM Singh, S Biswas, JK Deka - TENCON 2021-2021 IEEE ..., 2021 - ieeexplore.ieee.org  
 ... It is clear from Table II that our **incomplete testing** approach is efficient for wired-... **incomplete testing** is compromising the quality of the product but significantly reduces the number of **test** ...  
 ☆ Save Cite Related articles

**ATPG for Incomplete Testing of SoC and Power Aware TAM Architecture**  
 K Mrityunjay, S Biswas, JK Deka - 2018 15th IEEE India Council ..., 2018 - ieeexplore.ieee.org  
 ... Hence, huge amount of **test** data leads to sluggish **testing**, wastage of valuable ... **incomplete testing** of **SoC**. To the best of our knowledge there is no existing method proposed to **test SoC** ...  
 ☆ Save Cite Cited by 1 Related articles

Figure 7.6: Searching the Article on Google Scholar

This screenshot is identical to Figure 7.6, but the 'Cite' button for the first article is circled in red to highlight the citation options.

Articles About 18,70,000 results (0.12 sec)

**Any time**  
 Since 2026  
 Since 2025  
 Since 2022  
 Custom range...

**Sort by relevance**  
 Sort by date

**Any type**  
 Review articles

include patents  
 include citations  
 Create alert

**Incomplete Testing of SOC**  
 KM Singh, J Deka, S Biswas - Journal of Electronic Testing, 2023 - Springer  
 ... to come up with an efficient **testing** strategy that can help reduce the costs. ... **testing** the **SOC** incompletely instead of **testing** it thoroughly. This method involves compromising on the **test** ...  
 ☆ Save **Cite** Related articles All 5 versions

**ATPG for Incomplete Testing of SOC Considering Bridging Faults**  
 KM Singh, S Biswas, JK Deka - TENCON 2021-2021 IEEE ..., 2021 - ieeexplore.ieee.org  
 ... It is clear from Table II that our **incomplete testing** approach is efficient for wired-... **incomplete testing** is compromising the quality of the product but significantly reduces the number of **test** ...  
 ☆ Save Cite Related articles

**ATPG for Incomplete Testing of SoC and Power Aware TAM Architecture**  
 K Mrityunjay, S Biswas, JK Deka - 2018 15th IEEE India Council ..., 2018 - ieeexplore.ieee.org

Figure 7.7: Accessing Citation Options

The screenshot shows a Google Scholar search for "incomplete testing of SoC". A "Cite" dialog box is open, displaying the following citation formats:

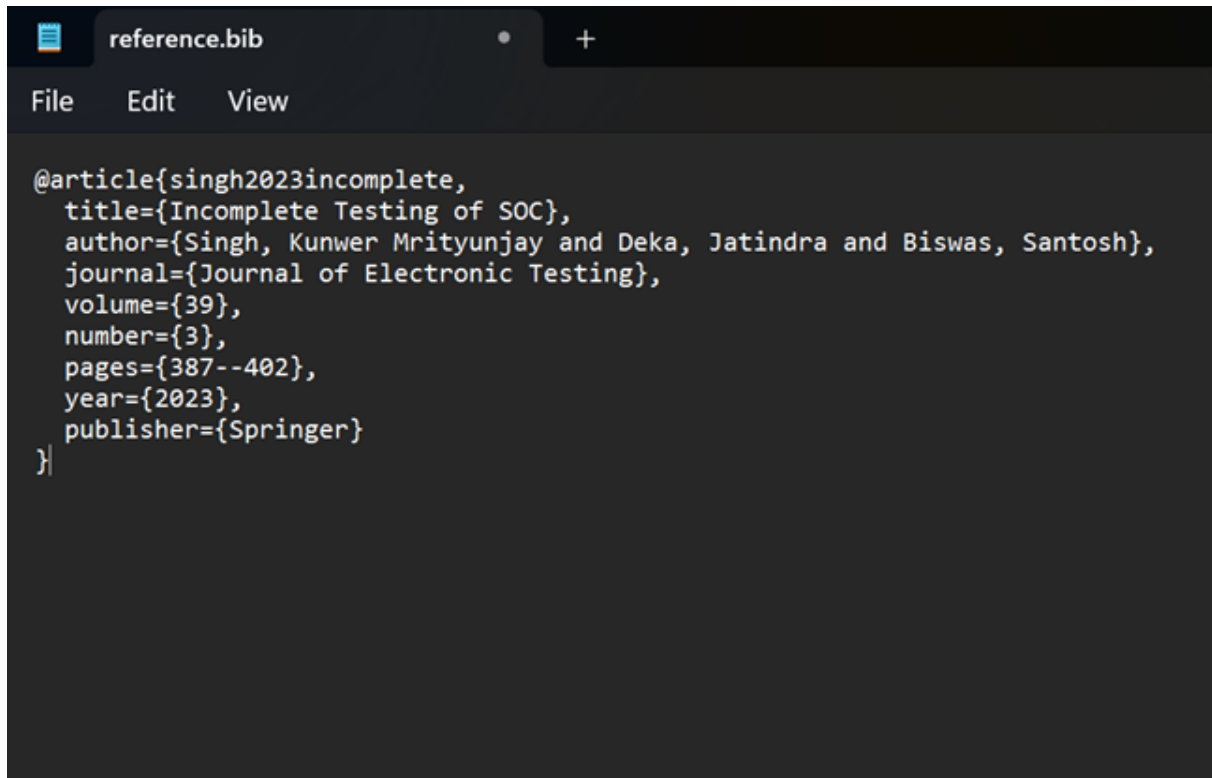
- MLA** Singh, Kunwer Mrityunjay, Jatindra Deka, and Santosh Biswas. "Incomplete Testing of SOC." *Journal of Electronic Testing* 39.3 (2023): 387-402.
- APA** Singh, K. M., Deka, J., & Biswas, S. (2023). Incomplete Testing of SOC. *Journal of Electronic Testing*, 39(3), 387-402.
- Chicago** Singh, Kunwer Mrityunjay, Jatindra Deka, and Santosh Biswas. "Incomplete Testing of SOC." *Journal of Electronic Testing* 39, no. 3 (2023): 387-402.
- Harvard** Singh, K.M., Deka, J. and Biswas, S., 2023. Incomplete Testing of SOC. *Journal of Electronic Testing*, 39(3), pp.387-402.
- Vancouver** Singh KM, Deka J, Biswas S. Incomplete Testing of SOC. *Journal of Electronic Testing*. 2023 Jun;39(3):387-402.

At the bottom of the dialog box, the following options are listed: [BibTeX](#) (circled in red), [EndNote](#), [RefMan](#), and [RefWorks](#).

Figure 7.8: Viewing Available Citation Formats

```
@article{singh2023incomplete,
  title={Incomplete Testing of SOC},
  author={Singh, Kunwer Mrityunjay and Deka, Jatindra and Biswas, Santosh},
  journal={Journal of Electronic Testing},
  volume={39},
  number={3},
  pages={387--402},
  year={2023},
  publisher={Springer}
}
```

Figure 7.9: Generating the BibTeX Entry



```

reference.bib
File Edit View

@article{singh2023incomplete,
  title={Incomplete Testing of SOC},
  author={Singh, Kunwer Mrityunjay and Deka, Jatindra and Biswas, Santosh},
  journal={Journal of Electronic Testing},
  volume={39},
  number={3},
  pages={387--402},
  year={2023},
  publisher={Springer}
}

```

Figure 7.10: Storing the BibTeX Entry in a .bib File

**Step 8: Storing the BibTeX Entry in a .bib File / चरण 8: .bib फ़ाइल में BibTeX प्रविष्टि सहेजना**

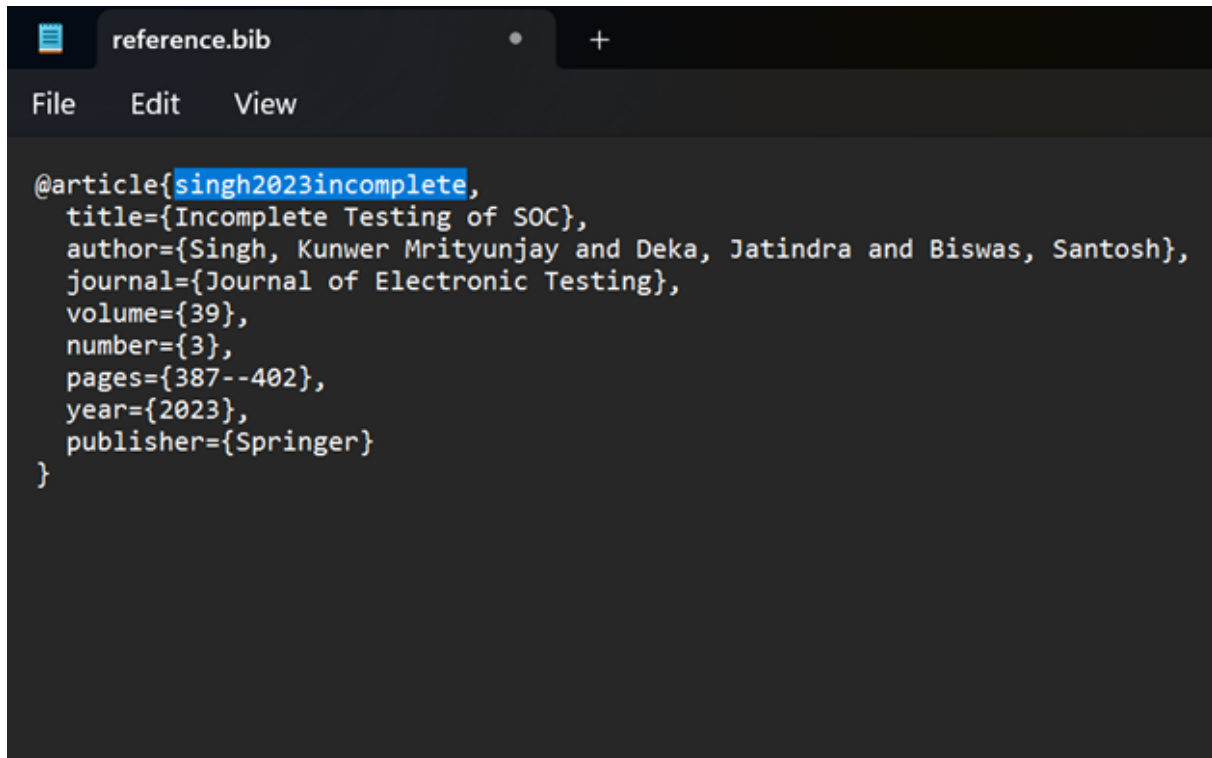
Figure 7.10 shows the copied BibTeX entry being saved into a file named `reference.bib`. This file acts as a centralized database for all references used in the document. Each reference is identified by a unique citation key, which is later used for in-text citations.

Figure 7.10 में कॉपी की गई BibTeX प्रविष्टि को `reference.bib` नामक फ़ाइल में सहेजते हुए दिखाया गया है। यह फ़ाइल दस्तावेज़ में उपयोग किए जाने वाले सभी संदर्भों के लिए एक केंद्रीकृत डेटाबेस के रूप में कार्य करती है। प्रत्येक संदर्भ को एक विशिष्ट उद्धरण कुंजी (citation key) द्वारा पहचाना जाता है, जिसका उपयोग बाद में दस्तावेज़ के भीतर उद्धरण देने के लिए किया जाता है।

**Step 9: Using the Citation in the LaTeX Document / चरण 9: LaTeX दस्तावेज़ में उद्धरण का उपयोग**

Once the `.bib` file is prepared, the reference can be cited in the LaTeX document using the `\cite{}` command. The bibliography is generated by specifying an appropriate bibliography style and the BibTeX file name at the end of the document. During compilation, LaTeX retrieves the bibliographic data from the `.bib` file and formats both in-text citations and the reference list automatically.

`.bib` फ़ाइल तैयार होने के बाद, LaTeX दस्तावेज़ में `\cite{}` कमांड का उपयोग करके संदर्भ को उद्धृत किया जा सकता है। दस्तावेज़ के अंत में उपयुक्त bibliography शैली और BibTeX फ़ाइल का नाम निर्दिष्ट करने पर संदर्भ सूची स्वतः उत्पन्न हो जाती है। कम्पाइलेशन प्रक्रिया के दौरान, LaTeX `.bib` फ़ाइल से बिलियोग्राफ़िक डेटा प्राप्त करता है और इन-टेक्स्ट उद्धरण तथा संदर्भ सूची दोनों को स्वचालित रूप से फ़ॉर्मेट करता है।



```

reference.bib
File Edit View

@article{singh2023incomplete,
  title={Incomplete Testing of SOC},
  author={Singh, Kunwer Mrityunjay and Deka, Jatindra and Biswas, Santosh},
  journal={Journal of Electronic Testing},
  volume={39},
  number={3},
  pages={387--402},
  year={2023},
  publisher={Springer}
}

```

Figure 7.11: Opening Text Editor

**Step 10: Including BibTeX in Your Document / चरण 10: अपने डॉक्यूमेंट में BibTeX जोड़ना**

```

\documentclass{article}
\usepackage{cite}
\usepackage[colorlinks=true,linkcolor=blue,citecolor=blue]{hyperref}
\begin{document}
  \section{Citation Example}
  This paper discusses SoC testing \cite{singh2023incomplete}.
  \bibliographystyle{plain}
  \bibliography{reference}
\end{document}

```

**Output**

# 1 Citation Example

This paper discusses SoC testing [1].

## References

- [1] Kunwer Mrityunjay Singh, Jatindra Deka, and Santosh Biswas. Incomplete testing of soc. *Journal of Electronic Testing*, 39(3):387–402, 2023.

The command `\usepackage{cite}` loads the `cite` package, which is used to improve the formatting of numeric citations in LaTeX documents. This package automatically sorts multiple citation numbers and compresses them

into a compact form (for example, [1, 2, 3, 4] is displayed as [1--4]), resulting in cleaner and more professional-looking references. It is particularly useful when using numeric bibliography styles such as `plain`, `unsrt`, or `ieeetr`.

`\usepackage{cite}` कमांड `cite` पैकेज को लोड करती है, जिसका उपयोग LaTeX दस्तावेज़ों में संख्यात्मक उद्धरणों (numeric citations) के फॉर्मेट को बेहतर बनाने के लिए किया जाता है। यह पैकेज एक से अधिक उद्धरण संख्याओं को स्वतः क्रमबद्ध करता है और उन्हें संक्षिप्त रूप में प्रदर्शित करता है; उदाहरण के लिए [1, 2, 3, 4] को [1--4] के रूप में दिखाया जाता है। इससे संदर्भ अधिक साफ़ और पेशेवर दिखाई देते हैं। यह पैकेज विशेष रूप से `plain`, `unsrt`, या `ieeetr` जैसी संख्यात्मक bibliography शैलियों के साथ उपयोगी होता है।

The command `\usepackage[colorlinks=true, linkcolor=blue, citecolor=blue]{hyperref}` loads the `hyperref` package, which adds hyperlink functionality to the document. The option `colorlinks=true` displays hyperlinks as colored text instead of surrounding them with rectangular boxes. The option `linkcolor=blue` sets the color of internal links, such as section, figure, and table references, to blue. The option `citecolor=blue` specifies that citation links generated by the `\cite` command appear in blue color, making references easily identifiable and clickable in the PDF output.

`\usepackage[colorlinks=true, linkcolor=blue, citecolor=blue]{hyperref}` कमांड `hyperref` पैकेज को लोड करती है, जो दस्तावेज़ में हाइपरलिंक सुविधा जोड़ता है। `colorlinks=true` विकल्प के कारण हाइपरलिंक रंगीन टेक्स्ट के रूप में दिखाई देते हैं, न कि आयताकार बॉक्स के रूप में। `linkcolor=blue` विकल्प आंतरिक लिंक जैसे सेक्शन, फ़िगर और टेबल संदर्भों का रंग नीला निर्धारित करता है। `citecolor=blue` विकल्प यह सुनिश्चित करता है कि `\cite` कमांड द्वारा बनाए गए उद्धरण लिंक नीले रंग में प्रदर्शित हों, जिससे PDF आउटपुट में संदर्भ आसानी से पहचाने जा सकें और उन पर क्लिक किया जा सके।

The sentence uses the command `\cite{singh2023incomplete}` to insert an in-text citation. This citation refers to a bibliographic entry with the same citation key stored in the BibTeX file.

इस वाक्य में `\cite{singh2023incomplete}` कमांड का उपयोग किया गया है, जो BibTeX फ़ाइल में `singh2023incomplete` कुंजी से संबंधित प्रविष्टि के लिए इन-टेक्स्ट उद्धरण प्रदान करती है।

The command `\bibliographystyle{plain}` selects the `plain` bibliography style, which produces numeric citations and lists references in alphabetical order based on the authors' names. This style is simple, clean, and widely used in academic documents.

`\bibliographystyle{plain}` कमांड `plain` bibliography शैली का चयन करती है। यह शैली संख्यात्मक उद्धरण (numeric citations) प्रदान करती है और संदर्भों को लेखकों के नाम के अनुसार वर्णानुक्रम (alphabetical order) में सूचीबद्ध करती है। यह शैली सरल, साफ़ और अकादमिक दस्तावेज़ों में व्यापक रूप से उपयोग की जाती है।

### 7.3.2 Manual Citation Using BibTeX / BibTeX द्वारा मैनुअल उद्धरण

Although Google Scholar is widely used, it is not the only method for obtaining citations. Many academic publishers and digital libraries provide direct BibTeX downloads for journal articles and conference papers. Reference management tools also support BibTeX export. When such options are unavailable, references can be created manually.

Manual citation is fully supported in BibTeX and is particularly useful for books, theses, reports, standards, and online content. By manually defining the appropriate BibTeX entry type and bibliographic fields in the `.bib` file, accurate citations can be produced without relying on Google Scholar or any external tool.

BibTeX provides several predefined entry types, each corresponding to a specific category of source. Core entry types include `@article`, `@book`, `@inbook`, `@incollection`, `@inproceedings`, `@proceedings`, `@techreport`, `@mastersthesis`, `@phdthesis`, `@manual`, and `@misc`. Modern BibTeX systems also support extended entry types such as `@online`, `@report`, `@dataset`, and `@software`. When a specific entry type is not supported by a bibliography style, BibTeX safely defaults to `@misc`.

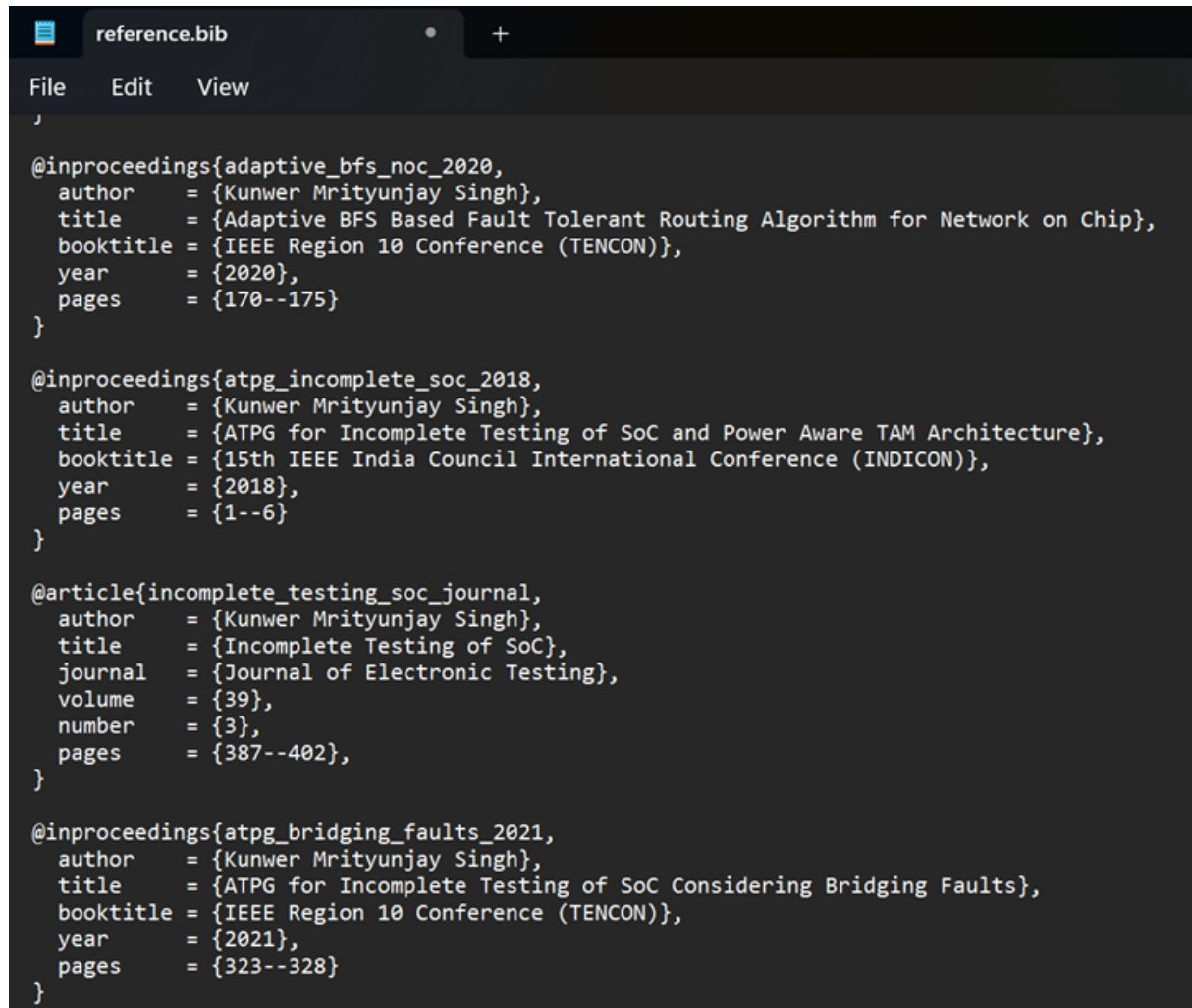
हालाँकि Google Scholar का व्यापक रूप से उपयोग किया जाता है, लेकिन यह उद्धरण प्राप्त करने का एकमात्र तरीका नहीं है। कई अकादमिक प्रकाशक और डिजिटल लाइब्रेरी जर्नल लेखों और सम्मेलन पत्रों के लिए सीधे BibTeX डाउनलोड की सुविधा प्रदान करते हैं। संदर्भ प्रबंधन टूल भी BibTeX निर्यात का समर्थन करते हैं। यदि ये विकल्प उपलब्ध न हों, तो संदर्भों को मैनुअल रूप से भी बनाया जा सकता है।

BibTeX में मैनुअल उद्धरण का पूर्ण समर्थन उपलब्ध है, जो विशेष रूप से पुस्तकों, थीसिस, रिपोर्ट, मानकों (standards) और ऑनलाइन सामग्री के लिए उपयोगी होता है। `.bib` फ़ाइल में उपयुक्त BibTeX एंट्री टाइप और बिलियोग्राफ़िक फ़ील्ड को मैनुअल रूप से परिभाषित करके, Google Scholar या किसी अन्य बाहरी टूल पर निर्भर हुए बिना सटीक उद्धरण तैयार किए जा सकते हैं।

BibTeX कई पूर्वनिर्धारित एंट्री टाइप प्रदान करता है, जो विभिन्न प्रकार के स्रोतों के अनुरूप होते हैं। मुख्य एंट्री टाइप में `@article`, `@book`, `@inbook`, `@incollection`, `@inproceedings`, `@proceedings`, `@techreport`, `@mastersthesis`,

@phdthesis, @manual, और @misc शामिल हैं। आधुनिक BibTeX प्रणालियाँ @online, @report, @dataset, और @software जैसे विस्तारित एंटी टाइप का भी समर्थन करती हैं। यदि कोई विशेष एंटी टाइप bibliography शैली द्वारा समर्थित नहीं होता, तो BibTeX सुरक्षित रूप से @misc का उपयोग करता है।

### Example of Manual Citation / Manual Citation का उदाहरण



```

reference.bib
File Edit View
,
@inproceedings{adaptive_bfs_noc_2020,
  author = {Kunwer Mrityunjay Singh},
  title = {Adaptive BFS Based Fault Tolerant Routing Algorithm for Network on Chip},
  booktitle = {IEEE Region 10 Conference (TENCON)},
  year = {2020},
  pages = {170--175}
}

@inproceedings{atpg_incomplete_soc_2018,
  author = {Kunwer Mrityunjay Singh},
  title = {ATPG for Incomplete Testing of SoC and Power Aware TAM Architecture},
  booktitle = {15th IEEE India Council International Conference (INDICON)},
  year = {2018},
  pages = {1--6}
}

@article{incomplete_testing_soc_journal,
  author = {Kunwer Mrityunjay Singh},
  title = {Incomplete Testing of SoC},
  journal = {Journal of Electronic Testing},
  volume = {39},
  number = {3},
  pages = {387--402},
}

@inproceedings{atpg_bridging_faults_2021,
  author = {Kunwer Mrityunjay Singh},
  title = {ATPG for Incomplete Testing of SoC Considering Bridging Faults},
  booktitle = {IEEE Region 10 Conference (TENCON)},
  year = {2021},
  pages = {323--328}
}

```

Figure 7.12: Manual Citation

### Input (LaTeX Code) / इनपुट (LaTeX कोड)

```

\documentclass{article}

\usepackage{cite}
\usepackage[colorlinks=true,linkcolor=blue,citecolor=blue]{hyperref}

\begin{document}
Fault-tolerant routing in NoC architectures has been widely studied \cite{adaptive_bfs_noc_2020}.
Advanced ATPG techniques improve incomplete testing of SoCs \cite{atpg_incomplete_soc_2018}.
Incomplete testing strategies are crucial for modern SoC designs \cite{incomplete_testing_soc_journal}.
Bridging faults significantly affect test coverage \cite{atpg_bridging_faults_2021}.

\bibliographystyle{plain}
\bibliography{reference}

\end{document}

```

## Output (PDF Result) / आउटपुट (PDF परिणाम)

**Output**

Fault-tolerant routing in NoC architectures has been widely studied [3]. Advanced ATPG techniques improve incomplete testing of SoCs [2]. Incomplete testing strategies are crucial for modern SoC designs [1]. Bridging faults significantly affect test coverage [4].

**References**

- [1] Kunwer Mrityunjay Singh. Incomplete testing of soc. *Journal of Electronic Testing*, 39(3):387–402.
- [2] Kunwer Mrityunjay Singh. Atpg for incomplete testing of soc and power aware tam architecture. In *15th IEEE India Council International Conference (INDICON)*, pages 1–6, 2018.
- [3] Kunwer Mrityunjay Singh. Adaptive bfs based fault tolerant routing algorithm for network on chip. In *IEEE Region 10 Conference (TENCON)*, pages 170–175, 2020.
- [4] Kunwer Mrityunjay Singh. Atpg for incomplete testing of soc considering bridging faults. In *IEEE Region 10 Conference (TENCON)*, pages 323–328, 2021.

This example explains how to cite research papers manually in LaTeX using BibTeX.

यह उदाहरण BibTeX का उपयोग करके LaTeX में शोध पत्रों का मैनुअल उद्धरण करने की प्रक्रिया को समझाता है।

Figure 7.12 illustrates that each reference is written manually in the bibliography file `reference.bib` using an appropriate BibTeX entry type such as `@inproceedings` for conference papers and `@article` for journal papers. Inside each entry, important details like the author name, title of the paper, conference or journal name, year of publication, and page numbers are specified. This step is called manual citation because the references are *typed* by the user instead of being downloaded from Google Scholar.

Figure 7.12 यह दर्शाता है कि प्रत्येक संदर्भ को `reference.bib` bibliography फ़ाइल में मैनुअल रूप से लिखा जाता है। सम्मेलन पत्रों के लिए `@inproceedings` तथा जर्नल पत्रों के लिए `@article` जैसे उपयुक्त BibTeX एंट्री टाइप का उपयोग किया जाता है। प्रत्येक प्रविष्टि में लेखक का नाम, शोध पत्र का शीर्षक, सम्मेलन या जर्नल का नाम, प्रकाशन वर्ष और पृष्ठ संख्या जैसी महत्वपूर्ण जानकारी दी जाती है। इस चरण को मैनुअल उद्धरण कहा जाता है, क्योंकि संदर्भ Google Scholar से डाउनलोड न होकर उपयोगकर्ता द्वारा स्वयं टाइप किए जाते हैं।

Second, in the LaTeX document, the `\cite{}` command is used at the place where a paper is referred to in the text. The text inside the curly brackets is the citation key, which exactly matches the key used in the `reference.bib` file. This key connects the sentence in the document with the correct reference entry. Multiple sentences can cite different papers using different citation keys.

दूसरे चरण में LaTeX दस्तावेज़ के भीतर, जहाँ भी किसी शोध पत्र का उल्लेख किया जाता है, वहाँ `\cite{}` कमांड का उपयोग किया जाता है। कर्ली ब्रैकेट के अंदर लिखा गया टेक्स्ट उद्धरण कुंजी (citation key) होता है, जो `reference.bib` फ़ाइल में प्रयुक्त कुंजी से बिल्कुल मेल खाता है। यह कुंजी दस्तावेज़ के वाक्य को सही संदर्भ प्रविष्टि से जोड़ती है। विभिन्न वाक्यों में अलग-अलग शोध पत्रों को अलग-अलग उद्धरण कुंजियों के माध्यम से उद्धृत किया जा सकता है।

Finally, the commands `\bibliographystyle{plain}` and `\bibliography{reference}` are written at the end of the document. When the document is compiled, BibTeX automatically replaces the citation keys with numbered references such as [1], [2], etc., in the text and generates a properly formatted References section at the end of the PDF.

अंतिम चरण में दस्तावेज़ के अंत में `\bibliographystyle{plain}` और

`\bibliography{reference}` कमांड लिखी जाती हैं। दस्तावेज़ को कम्पाइल करने पर BibTeX उद्धरण कुंजियों को स्वतः

[1], [2] आदि संख्यात्मक उद्धरणों से बदल देता है और PDF के अंत में सुव्यवस्थित References अनुभाग उत्पन्न करता है।

### 7.3.3 Citing Online Websites / ऑनलाइन वेबसाइटों का उद्धरण

Online resources can be cited using BibTeX entry types such as @online or @misc. A website citation typically includes the page title, author or organization, year of publication or last update, URL, and access date. Including the access date is recommended because online content may change over time.

ऑनलाइन संसाधनों का उद्धरण @online या @misc जैसे BibTeX एंट्री टाइप का उपयोग करके किया जा सकता है। एक वेबसाइट उद्धरण में सामान्यतः पृष्ठ का शीर्षक, लेखक या संस्था का नाम, प्रकाशन वर्ष या अंतिम अद्यतन वर्ष, URL तथा एक्सेस तिथि शामिल होती है। एक्सेस तिथि शामिल करना अनुशंसित है, क्योंकि ऑनलाइन सामग्री समय के साथ बदल सकती है।

## 7.4 Modern Citation Using biblatex / biblatex द्वारा आधुनिक उद्धरण

The biblatex package is a modern bibliography and citation management system designed to overcome the limitations of traditional BibTeX-based workflows. While BibTeX requires separate bibliography styles and additional citation packages such as cite or natbib, biblatex integrates citation control and bibliography formatting into a single, flexible framework.

biblatex पैकेज एक आधुनिक bibliography और उद्धरण प्रबंधन प्रणाली है, जिसे पारंपरिक BibTeX आधारित पद्धति की सीमाओं को दूर करने के लिए विकसित किया गया है। जहाँ BibTeX के साथ अलग-अलग bibliography शैलियों और cite या natbib जैसे अतिरिक्त पैकेजों की आवश्यकता होती है, वहीं biblatex उद्धरण नियंत्रण और संदर्भ सूची फॉर्मेटिंग को एक ही लचीले ढाँचे में सम्मिलित करता है।

### 7.4.1 Need for biblatex / biblatex की आवश्यकता

With the increasing complexity of academic documents, multilingual content, and diverse publication requirements, traditional BibTeX styles often provide limited customization. The biblatex package addresses these issues by offering enhanced citation commands, modern entry types, and extensive formatting control.

आधुनिक अकादमिक दस्तावेजों, बहुभाषी सामग्री, और विभिन्न प्रकाशन आवश्यकताओं के कारण पारंपरिक BibTeX शैलियाँ अक्सर सीमित हो जाती हैं। biblatex पैकेज उन्नत उद्धरण कमांड, आधुनिक एंट्री टाइप, और विस्तृत फॉर्मेटिंग नियंत्रण प्रदान करके इन समस्याओं का समाधान करता है।

### 7.4.2 Key Difference from Traditional BibTeX / पारंपरिक BibTeX से मुख्य अंतर

Unlike the traditional approach, biblatex does not use the \bibliographystyle and \bibliography commands. Instead, the bibliography style and citation format are defined directly as package options, and the bibliography is printed using \printbibliography.

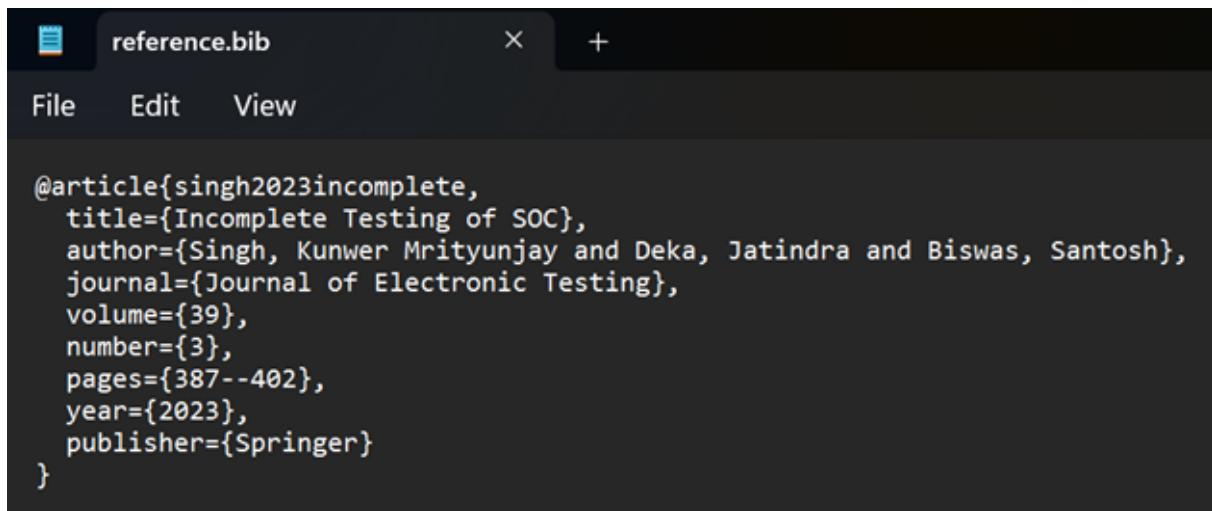
पारंपरिक पद्धति के विपरीत, biblatex में \bibliographystyle और \bibliography कमांड का उपयोग नहीं किया जाता। इसके स्थान पर उद्धरण शैली और संदर्भ सूची का फॉर्मेट सीधे पैकेज विकल्पों द्वारा निर्धारित किया जाता है, और संदर्भ सूची \printbibliography कमांड से प्रदर्शित की जाती है।

### 7.4.3 Backend Used by biblatex / biblatex का बैकएंड

The biblatex package uses Biber as its backend for processing bibliographic data. This provides better Unicode support, improved sorting, and more reliable handling of modern reference types.

biblatex पैकेज बिलियोग्राफिक डेटा प्रोसेसिंग के लिए Biber बैकएंड का उपयोग करता है। यह बेहतर Unicode समर्थन, उन्नत सॉर्टिंग, और आधुनिक संदर्भ प्रकारों को अधिक विश्वसनीय रूप से संभालने की सुविधा प्रदान करता है।

### 7.4.4 Example of Biblatex Citation / Biblatex Citation का उदाहरण



```
reference.bib
File Edit View
@article{singh2023incomplete,
  title={Incomplete Testing of SOC},
  author={Singh, Kunwer Mrityunjay and Deka, Jatindra and Biswas, Santosh},
  journal={Journal of Electronic Testing},
  volume={39},
  number={3},
  pages={387--402},
  year={2023},
  publisher={Springer}
}
```

Figure 7.13: Manual Citation

#### Input (LaTeX Code) / इनपुट (LaTeX कोड)

```
\documentclass{article}

\usepackage[style=authoryear,sorting=nyt]{biblatex}
\addbibresource{reference.bib}

\begin{document}
\section{Citation Example}
This paper discusses incomplete testing strategies for SoC designs \parencite{singh2023
incomplete}.

\printbibliography
\end{document}
```

#### Output (PDF Result) / आउटपुट (PDF परिणाम)

**Output**

## 1 Citation Example

This paper discusses incomplete testing strategies for SoC designs (Singh, Deka, and Biswas 2023).

## References

Singh, Kunwer Mrityunjay, Jatindra Deka, and Santosh Biswas (2023). “Incomplete Testing of SOC”. In: *Journal of Electronic Testing* 39.3, pp. 387–402.

The example explains the complete citation process using the biblatex package in LaTeX. First, a separate bibliography file named reference.bib is created, in which the research paper details are written, as shown in Fig-

Figure 7.13. In the LaTeX document, `\documentclass{article}` defines the document type, while `\usepackage[style=authoryear]{biblatex}` loads the `biblatex` package and selects the author–year citation style, with references sorted by name, year, and title. The command `\addbibresource{reference.bib}` links the bibliography file to the document. Within the document body, the citation is inserted using the command `\parencite{singh2023incomplete}`, where the citation key exactly matches the key defined in the `.bib` file, resulting in a parenthetical author–year citation in the text. Finally, the command `\printbibliography` generates the complete list of references at the end of the document. When the document is compiled using *Biber*, `biblatex` automatically processes the bibliographic data, replaces the citation key with a properly formatted in-text citation, and produces a clean and well-structured References section in the final PDF.

यह उदाहरण `biblatex` पैकेज का उपयोग करके LaTeX में उद्धरण की पूरी प्रक्रिया को समझाता है। सबसे पहले `reference.bib` नाम की एक अलग bibliography फ़ाइल बनाई जाती है, जिसमें शोध पत्र का विवरण मैन्युअल रूप से लिखा जाता है, जैसा कि Figure 7.13 में दर्शाया गया है। LaTeX दस्तावेज़ में `\documentclass{article}` दस्तावेज़ के प्रकार को परिभाषित करता है, जबकि

```
\usepackage[style=authoryear,sorting=nyt]
```

`{biblatex}` कमांड `biblatex` पैकेज को लोड करती है और लेखक--वर्ष उद्धरण शैली का चयन करती है, जिसमें संदर्भों को नाम, वर्ष और शीर्षक के अनुसार क्रमबद्ध किया जाता है।

`\addbibresource{reference.bib}` कमांड bibliography फ़ाइल को दस्तावेज़ से जोड़ती है। दस्तावेज़ की मुख्य सामग्री में `\parencite{singh2023incomplete}` कमांड का उपयोग करके उद्धरण दिया जाता है, जहाँ उद्धरण कुंजी `.bib` फ़ाइल में परिभाषित कुंजी से पूरी तरह मेल खाती है, और परिणामस्वरूप पाठ में कोष्ठकों के भीतर लेखक--वर्ष शैली का उद्धरण प्रदर्शित होता है। अंत में `\printbibliography` कमांड दस्तावेज़ के अंत में पूरी संदर्भ सूची उत्पन्न करती है। जब दस्तावेज़ को *Biber* के माध्यम से कम्पाइल किया जाता है, तो `biblatex` बिलियोग्राफ़िक डेटा को स्वतः प्रोसेस करता है, उद्धरण कुंजियों को सही इन-टेक्स्ट उद्धरणों से बदल देता है, और अंतिम PDF में एक सुव्यवस्थित तथा पेशेवर References अनुभाग तैयार करता है।

### Important Note: Common Issues in `biblatex` / `biblatex` में सामान्य समस्याएँ

If the list of references does not appear while using `biblatex`, the problem is usually due to one of the following reasons:

यदि `biblatex` का उपयोग करते समय संदर्भों की सूची प्रदर्शित नहीं होती है, तो इसका कारण सामान्यतः निम्नलिखित में से कोई एक होता है:

#### 1. Missing `\printbibliography`:

In `biblatex`, the bibliography is displayed only when the

`\printbibliography` command is used. The traditional `\bibliography{}` command does not work with `biblatex`.

`biblatex` में संदर्भ सूची केवल तभी प्रदर्शित होती है, जब `\printbibliography` कमांड का उपयोग किया जाए। पारंपरिक `\bibliography{}` कमांड `biblatex` के साथ कार्य नहीं करती है।

#### 2. Incorrect Compilation Method:

The `biblatex` package requires the *Biber* backend. The document must be compiled using *Biber* only. If you are using *TeXstudio*, you can change the settings as shown in Figure 7.14 by opening the *TeXstudio* settings, going to the Build section, and selecting *Biber* as the bibliography tool instead of *BibTeX*. Using *BibTeX* instead of *Biber* will prevent references from appearing.

`biblatex` पैकेज को *Biber* बैकएंड की आवश्यकता होती है। दस्तावेज़ को केवल *Biber* का उपयोग करके ही कम्पाइल करना चाहिए। यदि आप *TeXstudio* का उपयोग कर रहे हैं, तो Figure 7.14 में दर्शाए अनुसार *TeXstudio* की सेटिंग में जाकर Build विकल्प खोलें और Bibliography Tool को *BibTeX* के स्थान पर *Biber* पर सेट करें। *BibTeX* के स्थान पर *Biber* का उपयोग न करने पर संदर्भ सूची प्रदर्शित नहीं होगी।

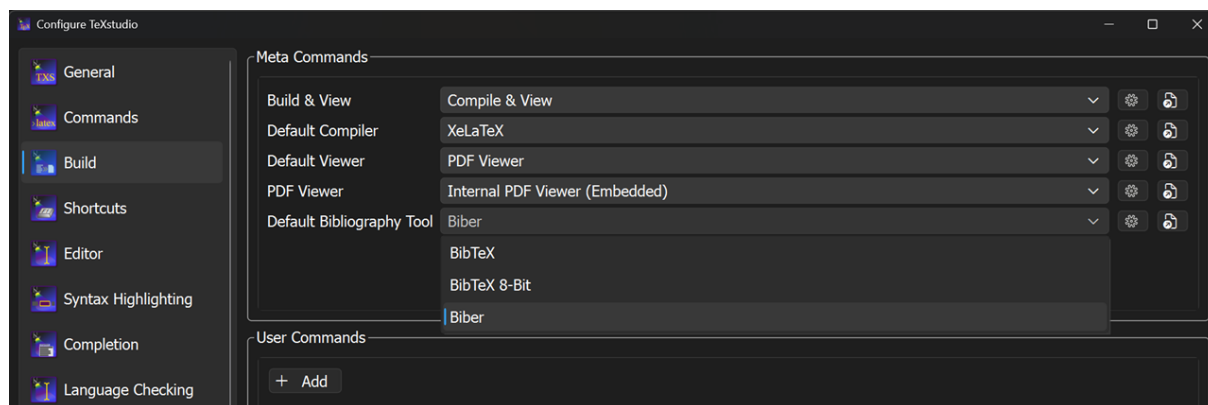


Figure 7.14: Change in Bibliography Tool

### 3. Missing or Incorrect Citations:

Ensure that the bibliography file is correctly linked using `\addbibresource{reference.bib}` and that at least one citation command such as `\parencite{key}` is used with a citation key that exactly matches the entry in the `.bib` file.

यह सुनिश्चित करें कि bibliography फ़ाइल को

`\addbibresource{reference.bib}` कमांड द्वारा सही रूप से जोड़ा गया है। साथ ही, दस्तावेज़ में कम से कम एक उद्धरण कमांड, जैसे `\parencite{key}`, का उपयोग किया गया हो तथा उद्धरण कुंजी `.bib` फ़ाइल में दी गई कुंजी से बिल्कुल मेल खाती हो।

### 7.4.5 Other biblatex Citation Commands / biblatex की अन्य उद्धरण कमांड

In addition to the basic `\cite{}` command, the `biblatex` package provides several other citation commands such as `\parencite`, `\textcite`, and `\footcite`, each designed for a specific citation purpose. These commands allow the author to control how a reference appears within the text, whether in parentheses, as part of the sentence, or as a footnote.

`\cite{}` कमांड के अतिरिक्त, `biblatex` पैकेज में `\parencite`, `\textcite`, और `\footcite` जैसी कई अन्य उद्धरण कमांड उपलब्ध होती हैं, जिनका उपयोग विभिन्न उद्देश्यों के लिए किया जाता है। ये कमांड लेखक को यह नियंत्रित करने की सुविधा देती हैं कि दस्तावेज़ के भीतर संदर्भ किस रूप में प्रदर्शित होगा, जैसे कोष्ठकों में, वाक्य का भाग बनकर, या फुटनोट के रूप में।

Command	Purpose	Citation Output
<code>\textcite{key}</code>	Author as part of the sentence	Singh et al. (2023)
<code>\parencite{key}</code>	Parenthetical citation	(Singh et al., 2023)
<code>\cite{key}</code>	Style-dependent citation	(Singh et al., 2023) or [1]
<code>\footcite{key}</code>	Footnote citation	1. Singh et al., 2023
<code>\autocite{key}</code>	Automatic citation style	(Singh et al., 2023)
<code>\citeauthor{key}</code>	Author name only	Singh et al.
<code>\citeyear{key}</code>	Publication year only	2023

# 8. LaTeX Projects for Students and Educators / छात्रों और शिक्षकों के लिए LaTeX अनुप्रयोग

LaTeX is a widely used typesetting system for creating professional academic documents. Students use LaTeX to write assignments, project reports, theses, and dissertations, while educators and researchers rely on it for preparing textbooks, lecture notes, and journal articles.

LaTeX एक व्यापक रूप से उपयोग किया जाने वाला टाइपसेटिंग सिस्टम है, जिसका प्रयोग व्यावसायिक एवं शैक्षणिक दस्तावेजों के निर्माण में किया जाता है। छात्र इसका उपयोग असाइनमेंट, परियोजना रिपोर्ट, शोध प्रबंध एवं थीसिस लिखने के लिए करते हैं, जबकि शिक्षक और शोधकर्ता पाठ्यपुस्तकें, व्याख्यान नोट्स तथा शोध पत्र तैयार करने के लिए LaTeX पर निर्भर करते हैं।

This chapter presents practical LaTeX projects related to book writing, research papers, and educational material, helping readers apply LaTeX effectively in academic and teaching environments.

यह अध्याय पुस्तक लेखन, शोध पत्रों तथा शैक्षणिक सामग्री से जुड़े व्यावहारिक LaTeX प्रोजेक्ट प्रस्तुत करता है, जिससे पाठक शैक्षणिक और शिक्षण परिस्थितियों में LaTeX का प्रभावी उपयोग सीख सकें।

## 8.1 Write a Book Using Custom Book Class File / कस्टम बुक क्लास फ़ाइल का उपयोग करके पुस्तक लिखना

In professional publishing workflows, the visual design and structural rules of a book are separated from its content. In LaTeX, this separation is achieved through the use of a class file (with extension .cls). In this section, you will learn:

- What a .cls file is ?
- Why create a custom book class ?
- Where the class file, chapters, main files are located ?
- How to write a book class file ?
- How to write a main.tex file?
- How to compile and get the final book pdf ?

व्यावसायिक प्रकाशन प्रक्रिया में किसी पुस्तक की डिज़ाइन और संरचना को उसकी विषय-वस्तु से अलग रखा जाता है। LaTeX में यह कार्य class file (जिसका एक्सटेंशन .cls होता है) के माध्यम से किया जाता है। इस अनुभाग में आप निम्नलिखित बातें सीखेंगे:

- .cls फ़ाइल क्या होती है?
- कस्टम बुक क्लास क्यों बनाई जाती है?
- क्लास फ़ाइल, चैप्टर और मुख्य फ़ाइलें कहाँ रखी जाती हैं?
- बुक क्लास फ़ाइल कैसे लिखी जाती है?
- main.tex फ़ाइल कैसे लिखी जाती है?
- अंतिम पुस्तक PDF कैसे तैयार की जाती है?

### 8.1.1 What Is a Class File .cls? / क्लास फ़ाइल ( .cls ) क्या है?

A  $\LaTeX$  class file defines the *overall structure and appearance* of a document. It controls:

- Page size and margins
- Font selection and font sizes
- Chapter and section formatting
- Headers and footers
- Title page design

The main document file (`main.tex`) should contain only the *content* of the book, not the formatting.

$\LaTeX$  की क्लास फ़ाइल किसी दस्तावेज़ की समग्र संरचना और रूप-रंग को परिभाषित करती है। यह निम्नलिखित बातों को नियंत्रित करती है:

- पृष्ठ का आकार और मार्जिन
- Font का चयन और Font का आकार
- अध्याय और अनुभाग का Formatting
- Headers और Footers
- Title Page की डिज़ाइन

मुख्य दस्तावेज़ फ़ाइल (`main.tex`) में केवल पुस्तक की विषय-वस्तु होनी चाहिए, न कि उसका Formatting।

### 8.1.2 Why Create a Custom Book Class? / कस्टम बुक क्लास क्यों बनाई जाती है?

Creating a custom class file allows authors and publishers to:

- Maintain consistent formatting across multiple books
- Reuse the same layout without repeating code
- Keep the main document clean and readable
- Modify the design without touching the content

This is the same approach used by academic publishers and journals.

कस्टम क्लास फ़ाइल बनाने से लेखक और प्रकाशकों को कई लाभ मिलते हैं:

- कई पुस्तकों में एक समान Formatting बनाए रखना
- एक ही Layout को बार-बार उपयोग करना, बिना कोड दोहराए
- मुख्य दस्तावेज़ को साफ़ और पढ़ने योग्य रखना
- विषय-वस्तु बदले बिना Design में बदलाव करना

यही पद्धति शैक्षणिक प्रकाशकों और Journals द्वारा अपनाई जाती है।

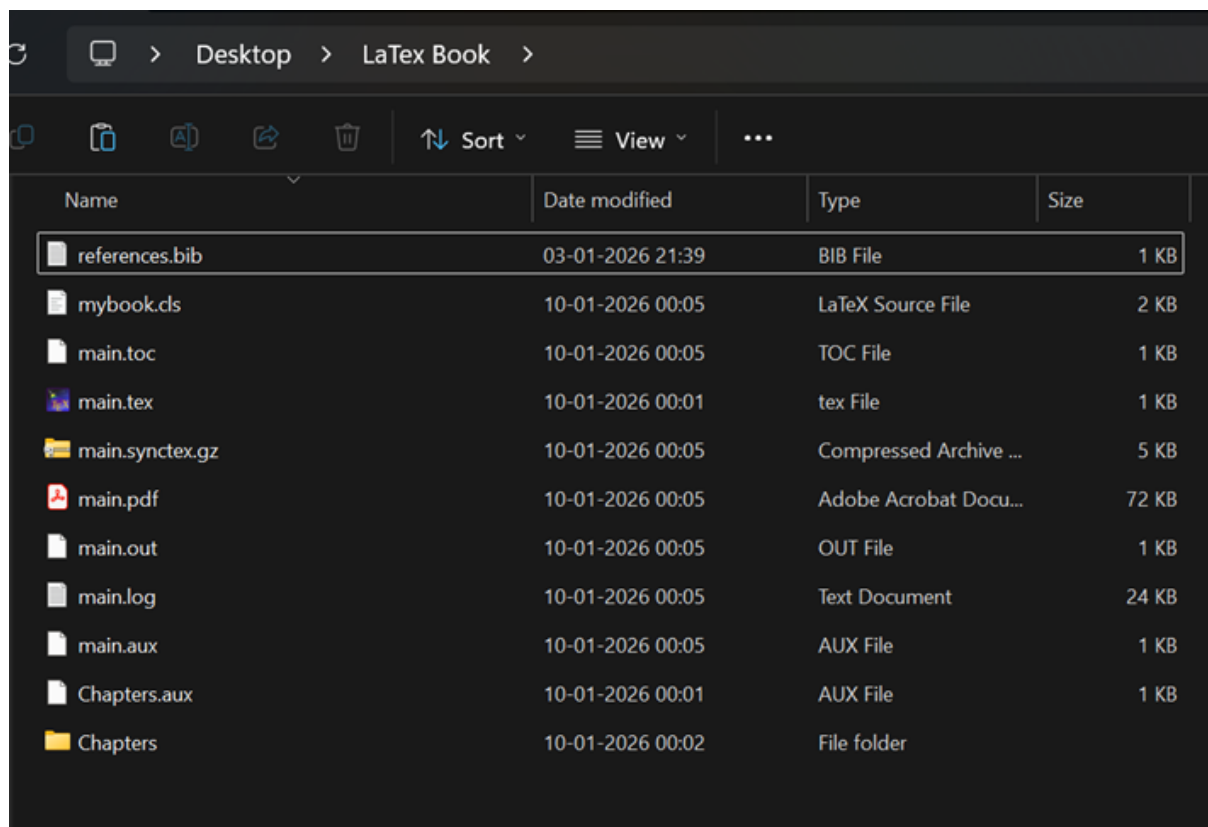


Figure 8.1: Folder Structure

### 8.1.3 Where the class file, chapters, main files are located ?/ क्लास फ़ाइल, चैप्टर और मुख्य फ़ाइलें कहाँ स्थित होती हैं?

Before creating the files, organize your project directory properly.

फ़ाइलें बनाने से पहले प्रोजेक्ट डायरेक्टरी को सही तरीके से व्यवस्थित करना आवश्यक है।

MyBook/

```
mybook.cls
main.tex
references.bib
Chapters/
  chapter01.tex
  chapter02.tex
```

The class file `mybook.cls` must be placed in the same directory as `main.tex`. A real time screen shot of a LaTeX Book folder is shown in Figure 8.1.

This project folder contains both source files and automatically generated compilation files. The file `main.tex` is the primary source document that defines the structure and content of the book, while `mybook.cls` is the custom class file responsible for controlling the overall layout, fonts, headers, and formatting. The file `references.bib` stores bibliographic entries used for citations. The `Chapters` directory contains individual chapter files that are included into the main document. During compilation,  $\text{\LaTeX}$  generates auxiliary files such as `main.aux` and `Chapters.aux` for cross-referencing, `main.toc` for the table of contents, `main.out` and `main.log` for compilation messages, and `main.synctex.gz` for source–PDF synchronization. The final output of the compilation process is the file `main.pdf`, which represents the completed book.

क्लास फ़ाइल `mybook.cls` को `main.tex` के साथ उसी डायरेक्टरी में रखा जाना आवश्यक है। चित्र 8.1 में LaTeX बुक फ़ोल्डर की वास्तविक संरचना दिखाई गई है।

यह प्रोजेक्ट फ़ोल्डर दो प्रकार की फ़ाइलें रखता है: स्रोत फ़ाइलें और कंपाइलेशन के दौरान स्वतः बनने वाली फ़ाइलें। `main.tex` मुख्य स्रोत फ़ाइल होती है, जो पुस्तक की संरचना और विषय-वस्तु को परिभाषित करती है, जबकि `mybook.cls`

कस्टम क्लास फ़ाइल होती है, जो लेआउट, फ़ॉन्ट, हेडर और फ़ॉर्मेटिंग को नियंत्रित करती है। `references.bib` फ़ाइल का उपयोग संदर्भ (citations) संग्रहित करने के लिए किया जाता है। Chapters डायरेक्टरी में प्रत्येक अध्याय की अलग-अलग फ़ाइलें होती हैं, जिन्हें मुख्य दस्तावेज़ में शामिल किया जाता है।

कंपाइलेशन के दौरान LaTeX सहायक फ़ाइलें उत्पन्न करता है, जैसे `main.aux` और `Chapters.aux` (क्रॉस-रेफरेंस के लिए), `main.toc` (टेबल ऑफ़ कंटेंट के लिए), `main.out` और `main.log` (कंपाइलेशन संदेशों के लिए), तथा `main.synctex.gz` (स्रोत और PDF सिंक्रोनाइज़ेशन के लिए)। कंपाइलेशन प्रक्रिया का अंतिम परिणाम `main.pdf` फ़ाइल होती है, जो पूरी तरह से तैयार पुस्तक को दर्शाती है।

### 8.1.4 How to write a book class file ?/ बुक क्लास फ़ाइल कैसे लिखी जाती है?

```

\NeedsTeXFormat{LaTeX2e}
\ProvidesClass{mybook}[2026/01/09 Simple Custom Book Class]

%-----
% Load base class
%-----
\LoadClass[12pt,openright]{book}

%-----
% Page layout
%-----
\RequirePackage[
  paperwidth=6in,
  paperheight=9in,
  margin=1in
]{geometry}

%-----
% Fonts (XeLaTeX / LuaLaTeX)
%-----
\RequirePackage{fontspec}
\setmainfont{Times New Roman}

%-----
% Common packages
%-----
\RequirePackage{graphicx}
\RequirePackage{amsmath,amssymb}
\RequirePackage{xcolor}
\RequirePackage{hyperref}
\hypersetup{
  colorlinks=false,
  pdfborder={0 0 0}
}

%-----
% Header and footer
%-----
\RequirePackage{fancyhdr}
\pagestyle{fancy}
\fancyhf{}
\fancyfoot[C]{\thepage}
\fancyhead[LO,RE]{\nouppercase{\leftmark}}
\renewcommand{\headrulewidth}{0.4pt}

%-----
% Title metadata
%-----
\newcommand{\subtitle}[1]{\def\@subtitle{#1}}

%-----
% Custom title page
%-----
\renewcommand{\maketitle}{
  \begin{titlepage}
    \thispagestyle{empty}
    \centering

```

```

{\Huge\bfseries \@title\par}
\vspace{0.5cm}
{\Large\itshape \@subtitle\par}
\vfill
{\large \@author\par}
\vspace{0.5cm}
{\large \@date\par}
\end{titlepage}
}

%-----
\endinput

```

Each line of the custom class file `mybook.cls`, highlighting its purpose and role in defining the structure and appearance of a book.

कस्टम क्लास फ़ाइल `mybook.cls` की प्रत्येक पंक्ति, किसी पुस्तक की संरचना और रूप-रंग को परिभाषित करने में उसके उद्देश्य और भूमिका को दर्शाती है।

### Class Declaration

`\NeedsTeXFormat{LaTeX2e}` specifies that the class file requires the  $\LaTeX 2e$  format and prevents it from being used with older versions of LaTeX.

`\ProvidesClass{mybook}[2026/01/09 Simple Custom Book Class]` registers the class with LaTeX. The optional argument contains the release date and a brief description, which appears in the compilation log.

`\NeedsTeXFormat{LaTeX2e}` यह निर्दिष्ट करता है कि क्लास फ़ाइल को  $\LaTeX 2e$  फ़ॉर्मेट की आवश्यकता है और इसे LaTeX के पुराने संस्करणों के साथ उपयोग होने से रोकता है।

`\ProvidesClass{mybook}[2026/01/09 Simple Custom Book Class]` LaTeX में क्लास को पंजीकृत करता है। इसका वैकल्पिक आर्युमेंट रिलीज़ तिथि और एक संक्षिप्त विवरण को दर्शाता है, जो कंपाइलेशन लॉग में दिखाई देता है।

### Loading the Base Class / आधार क्लास को लोड करना

`\LoadClass[12pt,openright]{book}` loads the standard book class as the base. The option `12pt` sets the default font size, while `openright` ensures that each chapter begins on a right-hand (odd-numbered) page.

`\LoadClass[12pt,openright]{book}` मानक `book` क्लास को आधार (base) के रूप में लोड करता है। `12pt` विकल्प डिफ़ॉल्ट फ़ॉन्ट आकार निर्धारित करता है, जबकि `openright` यह सुनिश्चित करता है कि प्रत्येक अध्याय दाएँ पृष्ठ (विषम संख्या वाले पृष्ठ) से प्रारंभ हो।

### Page Layout / पृष्ठ विन्यास

The `geometry` package is loaded using `\RequirePackage` to control the page dimensions. The settings define a page size of  $6 \times 9$  inches with uniform one-inch margins, a common size for textbooks.

`geometry` पैकेज को `\RequirePackage` का उपयोग करके पृष्ठ के आयामों (page dimensions) को नियंत्रित करने के लिए लोड किया जाता है। ये सेटिंग्स  $6 \times 9$  इंच के पृष्ठ आकार को परिभाषित करती हैं, जिसमें सभी ओर समान एक इंच के मार्जिन होते हैं, जो पाठ्यपुस्तकों के लिए एक सामान्य आकार है।

### Font Configuration / फ़ॉन्ट विन्यास

The `fontspec` package enables the use of system fonts and requires Xe $\LaTeX$  or Lua $\LaTeX$  for compilation. The command `\setmainfont{Times New Roman}` sets Times New Roman as the main text font for the entire document.

`fontspec` पैकेज सिस्टम फ़ॉन्ट्स के उपयोग को सक्षम करता है और कंपाइलेशन के लिए Xe $\LaTeX$  या Lua $\LaTeX$  की आवश्यकता होती है। `\setmainfont{Times New Roman}` कमांड पूरे दस्तावेज़ के लिए Times New Roman को मुख्य टेक्स्ट फ़ॉन्ट के रूप में निर्धारित करता है।

### Common Packages / सामान्य पैकेज

The package `graphicx` allows images to be included in the document. The packages `amsmath` and `amssymb` provide advanced mathematical environments and symbols. The `xcolor` package adds support for color handling. The `hyperref` package creates hyperlinks for references, the table of contents, and cross-references.

The `\hypersetup` command disables colored link text and removes visible link borders, resulting in a clean and print-friendly appearance.

`graphicx` पैकेज दस्तावेज़ में चित्रों को सम्मिलित करने की सुविधा देता है। `amsmath` और `amssymb` पैकेज उन्नत गणितीय पर्यावरण (environments) और प्रतीकों को उपलब्ध कराते हैं। `xcolor` पैकेज रंगों के प्रबंधन (color handling) के लिए समर्थन जोड़ता है। `hyperref` पैकेज संदर्भों, विषय-सूची (table of contents) और क्रॉस-रेफरेंस के लिए हाइपरलिंक बनाता है।

`\hypersetup` कमांड रंगीन लिंक टेक्स्ट को निष्क्रिय करता है और दिखाई देने वाली लिंक सीमाओं (link borders) को हटाता है, जिससे दस्तावेज़ का रूप साफ़ और प्रिंट-फ्रेंडली बनता है।

### Header and Footer Customization / हेडर और फुटर का अनुकूलन

The `fancyhdr` package is used to define custom headers and footers. All default header and footer content is cleared using `\fancyhf{}`. The page number is placed at the center of the footer using `\fancyfoot[C]{\thepage}`.

The command `\fancyhead[LO,RE]{\nouppercase{\leftmark}}` displays the current chapter title in the header on odd and even pages, without automatic capitalization. A horizontal rule of thickness 0.4 pt is added below the header.

`fancyhdr` पैकेज का उपयोग कस्टम हेडर और फुटर को परिभाषित करने के लिए किया जाता है। डिफ़ॉल्ट हेडर और फुटर की सभी सामग्री को `\fancyhf{}` का उपयोग करके साफ़ किया जाता है। पृष्ठ संख्या को फुटर के मध्य में `\fancyfoot[C]{\thepage}` के माध्यम से रखा जाता है।

`\fancyhead[LO,RE]{\nouppercase{\leftmark}}` कमांड विषम और सम पृष्ठों पर हेडर में वर्तमान अध्याय का शीर्षक प्रदर्शित करता है, बिना स्वचालित बड़े अक्षरों (capitalization) के। हेडर के नीचे 0.4 pt मोटाई की एक क्षैतिज रेखा जोड़ी जाती है।

### Title Metadata / शीर्षक मेटाडेटा

The command

```
\newcommand{\subtitle}[1]{\def\@subtitle{#1}}
```

defines a new macro for storing the subtitle text, which can later be used on the title page. This command is used to create a simple command for writing the subtitle of a book or document. This command allows the author to write a subtitle in the main document using `\subtitle{...}`. The subtitle text is saved so that it can be printed later on the title page.

The command `\newcommand` creates a new command in LaTeX. The name of the new command is `\subtitle`, which is meant to be used by the author. The part `[1]` means that the command takes one input, which is the subtitle text.

The symbol `#1` represents the subtitle text provided by the user. When the author writes `\subtitle{Designing Professional Books}` the text `Designing Professional Books` is stored by the class file.

The command `\newcommand{\subtitle}[1]{\def\@subtitle{#1}}` saves this text in a hidden variable named `\@subtitle`. The symbol `@` is used to mark internal commands in class files, meaning that this variable is not meant to be used directly in the main document.

Later, when the title page is created, the class file uses `\@subtitle` to display the subtitle. In this way, the author provides only the content, while the class file controls how the subtitle looks on the page.

In simple terms, the `\subtitle` command stores the subtitle text so that it can be used later when the title page is generated. This approach keeps the content separate from the design and makes the document easy to write and maintain.

यह कमांड `\newcommand{\subtitle}[1]{\def\@subtitle{#1}}` सबटाइटल के टेक्स्ट को संग्रहित करने के लिए एक नया मैक्रो परिभाषित करता है, जिसका उपयोग बाद में टाइटल पेज पर किया जा सकता है। इस कमांड का उपयोग पुस्तक या दस्तावेज़ का सबटाइटल लिखने के लिए एक सरल कमांड बनाने हेतु किया जाता है। यह लेखक को मुख्य दस्तावेज़ में `\subtitle{...}` का उपयोग करके सबटाइटल लिखने की सुविधा देता है। सबटाइटल का टेक्स्ट सहेज लिया जाता है ताकि बाद में उसे टाइटल पेज पर प्रदर्शित किया जा सके।

`\newcommand` कमांड LaTeX में एक नया कमांड बनाता है। नए कमांड का नाम `\subtitle` होता है, जिसका उपयोग लेखक द्वारा किया जाता है। `[1]` का अर्थ है कि यह कमांड एक इनपुट लेता है, जो कि सबटाइटल का टेक्स्ट होता है।

`#1` प्रतीक उपयोगकर्ता द्वारा प्रदान किए गए सबटाइटल टेक्स्ट को दर्शाता है। जब लेखक

`\subtitle{Designing Professional Books}` लिखता है, तो `Designing Professional Books` टेक्स्ट को क्लास फ़ाइल द्वारा संग्रहित कर लिया जाता है।

यह कमांड इस टेक्स्ट को `\@subtitle` नामक एक छिपे हुए वेरिएबल में सहेजता है। `@` चिन्ह का उपयोग क्लास फ़ाइलों में आंतरिक कमांड को दर्शाने के लिए किया जाता है, जिसका अर्थ है कि इस वेरिएबल का उपयोग सीधे मुख्य दस्तावेज़ में नहीं किया जाना चाहिए।

बाद में, जब टाइटल पेज बनाया जाता है, तब क्लास फ़ाइल `\@subtitle` का उपयोग करके सबटाइटल को प्रदर्शित करती है। इस प्रकार लेखक केवल विषय-वस्तु प्रदान करता है, जबकि क्लास फ़ाइल यह नियंत्रित करती है कि सबटाइटल पेज पर कैसे दिखाई देगा।

सरल शब्दों में, `\subtitle` कमांड सबटाइटल के टेक्स्ट को संग्रहित करता है ताकि टाइटल पेज बनने पर उसका उपयोग किया जा सके। यह विधि विषय-वस्तु और डिज़ाइन को अलग रखती है और दस्तावेज़ को लिखना व संभालना आसान बनाती है।

### Custom Title Page / कस्टम शीर्षक पृष्ठ

```
\renewcommand{\maketitle}{
```

This line redefines the existing `\maketitle` command provided by the standard book class, allowing us to design a custom title page.

```
\begin{titlepage}
```

This command starts the `titlepage` environment, which creates a separate page used only for the title of the book.

```
\thispagestyle{empty}
```

This command removes headers and footers from the title page so that no page number is printed.

```
\centering
```

This command centers all the content that follows on the title page.

```
{\Huge\bfseries \@title\par}
```

This line prints the main title of the book. The command `\Huge` sets a very large font size, `\bfseries` makes the text bold, and `\@title` contains the title text provided using `\title{}`.

```
\vspace{0.5cm}
```

This command adds vertical space between the title and the subtitle.

```
{\Large\itshape \@subtitle\par}
```

This line prints the subtitle of the book. The command `\Large` sets the font size, `\itshape` makes the subtitle italic, and `\@subtitle` stores the subtitle text defined using `\subtitle{}`.

```
\vfill
```

This command pushes the remaining content toward the bottom of the page, helping to balance the layout vertically.

```
{\large \@author\par}
```

This line displays the author name(s) provided using `\author{}`.

```
\vspace{0.5cm}
```

This command adds vertical space between the author name and the date.

```
{\large \@date\par}
```

This line displays the date of the document provided using `\date{}`.

```
\end{titlepage}
```

This command ends the `titlepage` environment.

```
}
```

This closing brace completes the redefinition of the `\maketitle`

command. `\renewcommand{\maketitle}{` यह पंक्ति मानक `book` क्लास द्वारा दिए गए `\maketitle` कमांड को पुनः परिभाषित करती है, जिससे हम अपना कस्टम शीर्षक पृष्ठ डिज़ाइन कर सकते हैं।

`\begin{titlepage}` यह कमांड `titlepage` एनवायरनमेंट को प्रारंभ करता है, जो पुस्तक के शीर्षक के लिए एक अलग पृष्ठ बनाता है।

`\thispagestyle{empty}` यह कमांड शीर्षक पृष्ठ से हेडर और फुटर को हटा देता है, जिससे पृष्ठ संख्या प्रदर्शित नहीं होती।

`\centering` यह कमांड शीर्षक पृष्ठ पर आने वाली सभी सामग्री को केंद्र में संरेखित करता है।

`{\Huge\bfseries \@title\par}` यह पंक्ति पुस्तक के मुख्य शीर्षक को प्रदर्शित करती है। यहाँ `\Huge` बड़ा फ़ॉन्ट आकार निर्धारित करता है, `\bfseries` टेक्स्ट को बोल्ड बनाता है, और `\@title` वह शीर्षक टेक्स्ट रखता है जो `\title{}` द्वारा प्रदान किया गया होता है।

`\vspace{0.5cm}` यह कमांड शीर्षक और उप-शीर्षक (subtitle) के बीच ऊर्ध्वाधर रिक्त स्थान जोड़ता है।

`{\Large\itshape \@subtitle\par}` यह पंक्ति पुस्तक का उप-शीर्षक प्रदर्शित करती है। यहाँ `\Large` फ़ॉन्ट आकार निर्धारित करता है, `\itshape` टेक्स्ट को इटैलिक बनाता है, और `\@subtitle` उप-शीर्षक टेक्स्ट को संग्रहीत करता है।

`\vfill` यह कमांड शेष सामग्री को पृष्ठ के निचले भाग की ओर ले जाता है, जिससे पृष्ठ का लेआउट संतुलित दिखाई देता है।

`{\large \@author\par}` यह पंक्ति लेखक का नाम प्रदर्शित करती है, जो `\author{}` द्वारा प्रदान किया गया होता है।

`\vspace{0.5cm}` यह कमांड लेखक के नाम और तिथि के बीच रिक्त स्थान जोड़ता है।

`{\large \@date\par}` यह पंक्ति दस्तावेज़ की तिथि प्रदर्शित करती है, जो `\date{}` द्वारा दी जाती है।

`\end{titlepage}` यह कमांड `titlepage` एनवायरनमेंट को समाप्त करता है।

`\endinput` यह कमांड क्लास फ़ाइल के अंत को दर्शाता है। इसके बाद लिखा गया कोई भी पाठ `LaTeX` द्वारा अनदेखा कर दिया जाता है।

### End of the Class File / क्लास फ़ाइल का अंत

The command `\endinput` marks the end of the class file. Any text written after this command is ignored by `LaTeX`, making it a recommended best practice in class and package files.

`\endinput` कमांड क्लास फ़ाइल के अंत को दर्शाता है। इस कमांड के बाद लिखा गया कोई भी पाठ `LaTeX` द्वारा अनदेखा कर दिया जाता है, जिसके कारण यह क्लास और पैकेज फ़ाइलों में एक अनुशंसित सर्वोत्तम अभ्यास (best practice) माना जाता है।

### 8.1.5 How to write a main.tex file?

```
\documentclass{mybook}

%-----
% Book information
%-----
\title{Learning LaTeX Class Files}
\subtitle{Designing Professional Books}
\author{Dr. Kunwer Mrityunjay Singh}
\date{2026}

\begin{document}

\frontmatter
\maketitle
\tableofcontents

\mainmatter
```

```
\include{Chapters/chapter01}
\include{Chapters/chapter02}
```

```
\backmatter
\include{Chapters/chapter03}
```

```
\end{document}
```

mybook.cls.

```
\documentclass{mybook}
```

This line tells  $\LaTeX$  to use the custom class file named mybook.cls. All formatting rules, page layout, fonts, and title page design are taken from this class file.

`\documentclass{mybook}` पंक्ति mybook.cls नामक कस्टम क्लास फ़ाइल का उपयोग करने के लिए LaTeX को निर्देश देती है। पृष्ठ विन्यास, फ़ॉन्ट, शीर्षक पृष्ठ की डिज़ाइन और सभी फ़ॉर्मेटिंग नियम इसी क्लास फ़ाइल से लिए जाते हैं।

```
%-----
```

This line is a comment used only for readability. Comments begin with the % symbol and are ignored by LaTeX.

यह पंक्ति केवल पठनीयता (readability) के लिए लिखी गई एक टिप्पणी है। % चिन्ह से शुरू होने वाली पंक्तियाँ LaTeX द्वारा अनदेखा कर दी जाती हैं।

```
% Book information
```

This comment indicates that the following commands define the book metadata.

यह टिप्पणी दर्शाती है कि इसके बाद आने वाले कमांड पुस्तक की मेटाडेटा जानकारी को परिभाषित करते हैं।

```
\title{Learning LaTeX Class Files}
```

This command sets the main title of the book. The value provided here is stored internally and later printed on the title page using the class file.

`\title{Learning LaTeX Class Files}` कमांड पुस्तक का मुख्य शीर्षक निर्धारित करता है। यह शीर्षक आंतरिक रूप से संग्रहीत किया जाता है और बाद में शीर्षक पृष्ठ पर प्रदर्शित किया जाता है।

```
\subtitle{Designing Professional Books}
```

This command sets the subtitle of the book. It uses the custom `\subtitle` command defined in the class file.

`\subtitle{Designing Professional Books}` कमांड पुस्तक का उप-शीर्षक निर्धारित करता है। यह क्लास फ़ाइल में परिभाषित कस्टम `\subtitle` कमांड का उपयोग करता है।

```
\author{Dr. Kunwer Mrityunjay Singh}
```

This command specifies the author name. The text is stored internally and displayed on the title page.

`\author{Dr. Kunwer Mrityunjay Singh}` कमांड लेखक का नाम निर्धारित करता है। यह जानकारी आंतरिक रूप से संग्रहीत होती है और शीर्षक पृष्ठ पर प्रदर्शित की जाती है।

```
\date{2026}
```

This command defines the date of publication. If omitted,  $\LaTeX$  would use the current date by default.

`\date{2026}` कमांड प्रकाशन की तिथि निर्धारित करता है। यदि यह कमांड न दिया जाए, तो LaTeX स्वतः वर्तमान तिथि का उपयोग करता है।

```
\begin{document}
```

This command marks the beginning of the document content. Everything before this line is considered preamble information.

`\begin{document}` कमांड दस्तावेज़ की वास्तविक सामग्री की शुरुआत को दर्शाता है। इस पंक्ति से पहले की सभी जानकारी को प्रीएम्बल माना जाता है।

```
\frontmatter
```

This command switches the document to front-matter mode. In this mode, page numbers are usually in Roman numerals and chapters are not numbered.

`\frontmatter` कमांड दस्तावेज़ को फ्रंट-मैटर मोड में स्थानांतरित करता है। इस मोड में पृष्ठ संख्याएँ प्रायः रोमन अंकों में होती हैं और अध्याय क्रमांकित नहीं होते।

`\maketitle`

This command generates the title page using the custom definition provided in the class file.

`\maketitle` कमांड क्लास फ़ाइल में परिभाषित कस्टम डिज़ाइन के अनुसार शीर्षक पृष्ठ तैयार करता है।

`\tableofcontents`

This command generates the table of contents automatically based on the chapter and section headings in the document.

`\tableofcontents` कमांड दस्तावेज़ के अध्याय और अनुभाग शीर्षकों के आधार पर स्वतः विषय-सूची तैयार करता है।

`\mainmatter`

This command switches the document to main-matter mode. Chapters are numbered normally and page numbers use Arabic numerals.

`\mainmatter` कमांड दस्तावेज़ को मुख्य भाग (main matter) में ले जाता है। इस मोड में अध्याय सामान्य रूप से क्रमांकित होते हैं और पृष्ठ संख्याएँ अरबी अंकों में होती हैं।

`\include{Chapters/chapter01}`

This command includes the file `chapter01.tex` from the `Chapters` directory. Each chapter is written in a separate file for better organization.

`\include{Chapters/chapter01}` कमांड `Chapters` डायरेक्टरी से `chapter01.tex` फ़ाइल को दस्तावेज़ में सम्मिलित करता है। बेहतर संगठन के लिए प्रत्येक अध्याय अलग फ़ाइल में लिखा जाता है।

`\include{Chapters/chapter02}`

This line includes the second chapter file in the document.

यह पंक्ति दस्तावेज़ में दूसरे अध्याय की फ़ाइल को सम्मिलित करती है।

`\backmatter`

This command switches the document to back-matter mode. It is commonly used for appendices, references, or concluding chapters.

`\backmatter` कमांड दस्तावेज़ को बैक-मैटर मोड में ले जाता है। इसका उपयोग सामान्यतः परिशिष्ट, संदर्भ या निष्कर्ष अध्यायों के लिए किया जाता है।

`\include{Chapters/chapter03}`

This line includes a chapter placed in the back matter, such as a conclusion or appendix.

यह पंक्ति बैक-मैटर में रखे गए अध्याय, जैसे निष्कर्ष या परिशिष्ट, को दस्तावेज़ में सम्मिलित करती है।

`\end{document}`

This command marks the end of the document. Anything written after this line is ignored by LaTeX.

`\end{document}` कमांड दस्तावेज़ के अंत को दर्शाता है। इस पंक्ति के बाद लिखा गया कोई भी पाठ LaTeX द्वारा अनदेखा कर दिया जाता है।

## 8.1.6 How to compile and get the final book pdf ? / अंतिम पुस्तक PDF कैसे तैयार करें?

To get the final book PDF, first make sure that the files `mybook.cls`, `main.tex`, and all chapter files are in the correct folders. Since the document uses modern fonts, select *XeLaTeX* as the compiler. Open `main.tex` in a LaTeX editor such as TeXstudio or TeXworks and click the Compile button, or run the command `xelatex main.tex` from the command line. Compile the document at least two times so that the table of contents and page numbers are updated correctly. After successful compilation, a file named `main.pdf` is created in the project folder, which is the final printable version of the book.

अंतिम पुस्तक PDF प्राप्त करने के लिए सबसे पहले यह सुनिश्चित करें कि `mybook.cls`, `main.tex` और सभी चैप्टर फ़ाइलें सही फ़ोल्डर में मौजूद हों। क्योंकि इस दस्तावेज़ में आधुनिक फ़ॉन्ट्स का उपयोग किया गया है, इसलिए कंपाइलर के रूप में XeLaTeX का चयन करें। `main.tex` फ़ाइल को TeXstudio या TeXworks जैसे LaTeX एडिटर में खोलें और Compile बटन पर क्लिक करें, या कमांड लाइन से `xelatex main.tex` कमांड चलाएँ। टेबल ऑफ़ कंटेंट और पेज नंबर सही ढंग से अपडेट होने के लिए दस्तावेज़ को कम से कम दो बार कंपाइल करें। सफल कंपाइलेशन के बाद प्रोजेक्ट फ़ोल्डर में `main.pdf` नाम की फ़ाइल बनती है, जो पुस्तक का अंतिम प्रिंट करने योग्य संस्करण होती है।

## 8.2 Writing a Journal or Conference Paper Using a Given Template / दिए गए टेम्पलेट का उपयोग करके जर्नल या कॉन्फ्रेंस पेपर लिखना

### Step 1: Download the Official Template / आधिकारिक टेम्पलेट डाउनलोड करें

Every journal or conference provides an official L<sup>A</sup>T<sub>E</sub>X template. Download the template files (usually `.tex`, `.cls`, or `.sty`) from the journal or conference website.

प्रत्येक जर्नल या कॉन्फ्रेंस अपनी आधिकारिक LaTeX टेम्पलेट प्रदान करता है। इन टेम्पलेट फ़ाइलों को (जैसे `.tex`, `.cls`, `.sty`) जर्नल या कॉन्फ्रेंस की वेबसाइट से डाउनलोड करें।

### Step 2: Read the Author Guidelines / लेखक निर्देश पढ़ें

Before writing, carefully read the author guidelines. These instructions specify page limits, font size, column format, reference style, and submission rules.

लेखन शुरू करने से पहले लेखक निर्देशों को ध्यान से पढ़ें। इनमें पृष्ठ सीमा, फ़ॉन्ट आकार, कॉलम फ़ॉर्मेट, संदर्भ शैली और सबमिशन नियम दिए होते हैं।

### Step 3: Open the Main Template File / मुख्य टेम्पलेट फ़ाइल खोलें

Open the main `.tex` file provided in the template. Do not change the document class or formatting commands. टेम्पलेट में दी गई मुख्य `.tex` फ़ाइल खोलें। डॉक्यूमेंट क्लास या फ़ॉर्मेटिंग कमांड में बदलाव न करें।

### Step 4: Replace Sample Content with Your Work / नमूना सामग्री को अपने शोध से बदलें

Replace the sample title, author names, abstract, and sections with your own research content while keeping the structure intact.

नमूना शीर्षक, लेखक नाम, सार (abstract) और अनुभागों को अपने शोध कार्य से बदलें, लेकिन संरचना वही रखें।

### Step 5: Write the Paper Section by Section / अनुभाग-दर-अनुभाग पेपर लिखें

Write each section carefully: Introduction, Related Work, Methodology, Results, and Conclusion. Follow the order and headings given in the template.

प्रत्येक अनुभाग को सावधानीपूर्वक लिखें: Introduction, Related Work, Methodology, Results और Conclusion। टेम्पलेट में दिए गए क्रम और शीर्षकों का पालन करें।

### Step 6: Insert Figures, Tables, and References / चित्र, तालिकाएँ और संदर्भ जोड़ें

Add figures and tables using the commands already shown in the template. Manage references using the recommended bibliography style.

चित्र और तालिकाएँ टेम्पलेट में दिए गए कमांड के अनुसार जोड़ें। संदर्भों के लिए सुझाई गई बिब्लियोग्राफी शैली का उपयोग करें।

### Step 7: Compile Using the Recommended Compiler / अनुशंसित कंपाइलर से कंपाइल करें

Compile the document using the compiler specified by the template (e.g., pdfLaTeX or XeLaTeX). Compile multiple times to fix references.

टेम्पलेट में बताए गए कंपाइलर (जैसे pdfLaTeX या XeLaTeX) का उपयोग करके दस्तावेज़ को कंपाइल करें। संदर्भ सही करने के लिए कई बार कंपाइल करें।

### Step 8: Final Check and Submission / अंतिम जाँच और सबमिशन

Check formatting, page limits, and PDF quality. Submit the final PDF through the journal or conference submission system.

फ़ॉर्मेटिंग, पृष्ठ सीमा और PDF गुणवत्ता की अंतिम जाँच करें। इसके बाद जर्नल या कॉन्फ्रेंस की सबमिशन प्रणाली में अंतिम PDF अपलोड करें।

# 9. Beamer: Presentation Design / Beamer: प्रस्तुति बनाना

## 9.1 Introduction to Beamer / Beamer का परिचय

### 9.1.1 What is Beamer? / Beamer क्या है?

Beamer is a powerful  $\LaTeX$  class used for creating professional-quality presentations in PDF format. It is widely used in academic, scientific, and technical fields because it allows the creation of well-structured slides containing mathematical equations, figures, tables, and references. Since Beamer presentations are written as plain text source files, they are easy to edit, manage using version control systems, and maintain over a long period of time.

Beamer एक शक्तिशाली  $\LaTeX$  क्लास है जिसका उपयोग PDF प्रारूप में उच्च गुणवत्ता वाली प्रस्तुतियाँ बनाने के लिए किया जाता है। यह विशेष रूप से शैक्षणिक, वैज्ञानिक तथा तकनीकी क्षेत्रों में लोकप्रिय है क्योंकि इसके माध्यम से गणितीय समीकरण, चित्र, सारणियाँ और संदर्भों के साथ सुव्यवस्थित स्लाइड्स तैयार की जा सकती हैं। Beamer में प्रस्तुति साधारण टेक्स्ट फाइल के रूप में लिखी जाती है, जिससे इसे संशोधित करना, version control में रखना तथा लंबे समय तक बनाए रखना आसान होता है।

### 9.1.2 Why Use Beamer for Technical Presentations / तकनीकी प्रस्तुतियों के लिए Beamer का उपयोग क्यों करें

- Beamer provides excellent support for mathematical equations, symbols, and scientific notation, which are essential in technical presentations.
- It allows precise control over slide structure and layout, helping in the clear and systematic presentation of complex technical concepts.
- Beamer produces presentations with high typographical quality, ensuring professional and visually consistent slides.
- Presentations are created using  $\LaTeX$  source files, making them easy to edit, reuse, and maintain over time.
- Beamer integrates well with version control systems, which is particularly useful for collaborative academic and research work.
- Due to these features, Beamer is widely used by engineers, researchers, and educators for technical and academic presentations.
- Beamer गणितीय समीकरणों, प्रतीकों तथा वैज्ञानिक संकेतन के लिए उत्कृष्ट समर्थन प्रदान करता है, जो तकनीकी प्रस्तुतियों में आवश्यक होते हैं।
- यह स्लाइड की संरचना और लेआउट पर सटीक नियंत्रण देता है, जिससे जटिल तकनीकी अवधारणाओं को स्पष्ट और क्रमबद्ध रूप में प्रस्तुत किया जा सकता है।
- Beamer उच्च गुणवत्ता वाली टाइपोग्राफी के साथ प्रस्तुतियाँ तैयार करता है, जिससे स्लाइड्स पेशेवर और दृश्य रूप से सुसंगत बनती हैं।
- प्रस्तुतियाँ  $\LaTeX$  स्रोत फाइलों के रूप में बनाई जाती हैं, जिससे उन्हें संशोधित करना, पुनः उपयोग करना और लंबे समय तक बनाए रखना आसान होता है।
- Beamer का version control प्रणालियों के साथ अच्छा एकीकरण होता है, जो सहयोगात्मक शैक्षणिक और शोध कार्य के लिए विशेष रूप से उपयोगी है।
- इन सभी विशेषताओं के कारण Beamer का उपयोग अभियंताओं, शोधकर्ताओं और शिक्षकों द्वारा तकनीकी तथा शैक्षणिक प्रस्तुतियों के लिए व्यापक रूप से किया जाता है।

### 9.1.3 Advantages of Beamer over PowerPoint and LibreOffice Impress / PowerPoint एवं LibreOffice Impress पर Beamer के लाभ

Feature (Beamer)	Limitation in PowerPoint / LibreOffice Impress
Native $\text{\LaTeX}$ mathematical typesetting with perfect equation alignment	Limited and inconsistent equation formatting
Complete presentation defined in a single text source file	Slides stored as binary or complex GUI files
Seamless integration with version control systems (Git, SVN)	Version control support is difficult and unreliable
Automatic and global formatting through themes and styles	Manual formatting required for each slide
Programmatic slide generation and reuse of content	No direct support for programmatic slide creation
Reproducible and platform-independent PDF output	Output may vary across systems and versions

Table 9.1: Advantages of Beamer over PowerPoint and LibreOffice Impress

Beamer की विशेषता	PowerPoint / LibreOffice Impress की सीमा
पूर्णतः मूल $\text{\LaTeX}$ आधारित गणितीय टाइपसेटिंग एवं सटीक समीकरण संरेखण	सीमित एवं असंगत समीकरण फॉर्मेटिंग
पूरी प्रस्तुति एक ही टेक्स्ट स्रोत फाइल में परिभाषित	स्लाइड्स binary या जटिल GUI फाइलों में संग्रहीत
Git, SVN जैसी version control प्रणालियों के साथ सहज एकीकरण	version control का उपयोग कठिन और अविश्वसनीय
थीम और स्टाइल द्वारा स्वचालित एवं वैश्विक फॉर्मेटिंग	प्रत्येक स्लाइड के लिए मैनुअल फॉर्मेटिंग आवश्यक
प्रोग्रामेटिक रूप से स्लाइड निर्माण एवं सामग्री का पुनः उपयोग	प्रोग्रामेटिक स्लाइड निर्माण का अभाव
पुनरुत्पादनीय एवं प्लेटफॉर्म-स्वतंत्र PDF आउटपुट	भिन्न प्रणालियों पर आउटपुट में अंतर संभव

Table 9.2: PowerPoint एवं LibreOffice Impress पर Beamer के लाभ

## 9.2 First Beamer Presentation / प्रथम Beamer प्रस्तुति

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{beamer}
\usetheme{default}

\begin{document}

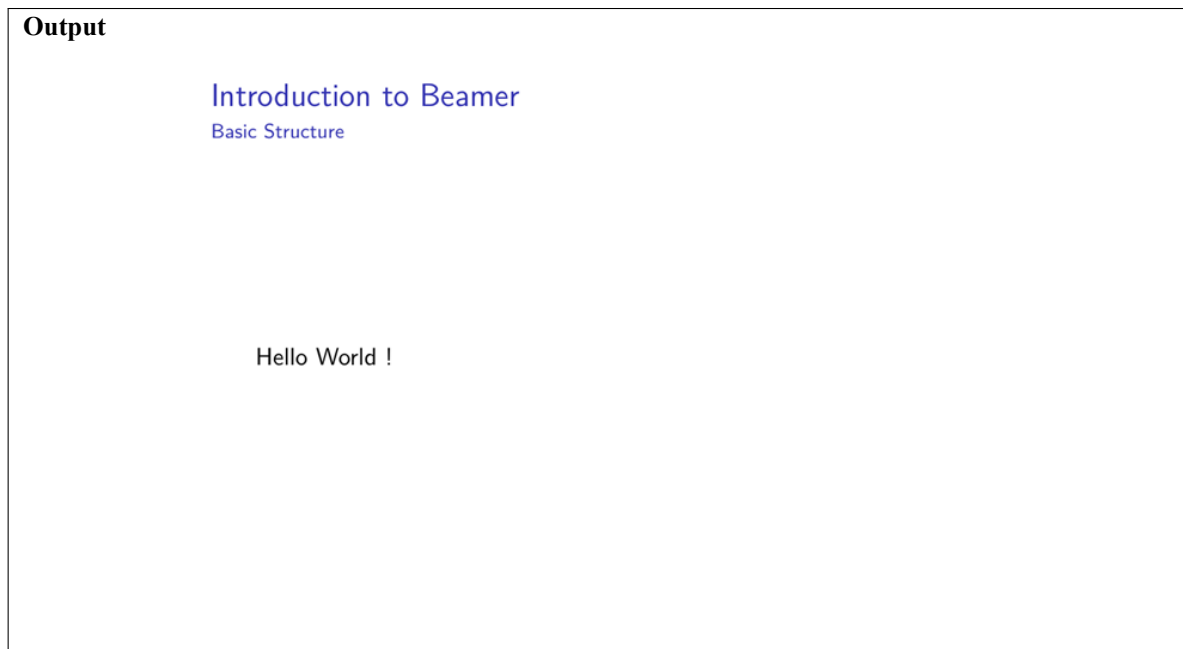
\begin{frame}{Title of the Slide}
\framesubtitle{Basic Structure}

Hello World !

\end{frame}

\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)**



This program creates a simple Beamer presentation with a single slide. The command `\documentclassbeamer` selects the Beamer class, and

`\usethemedefault` applies the default theme. Inside the `document` environment, the `frame` environment defines one slide with the title `Title of the Slide`. The command `\framesubtitleBasic Structure` adds a subtitle below the title, and the text `Hello World !` appears as the main content of the slide when the document is compiled.

यह प्रोग्राम एक सरल Beamer प्रस्तुति तैयार करता है जिसमें एक ही स्लाइड होती है।

`\documentclassbeamer` कमांड Beamer क्लास को चुनता है और `\usethemedefault` डिफ़ॉल्ट theme लागू करता है। `document` environment के भीतर `frame` environment एक स्लाइड को परिभाषित करता है, जिसका शीर्षक `Title of the Slide` होता है।

`\framesubtitleBasic Structure` कमांड शीर्षक के नीचे उपशीर्षक जोड़ता है और `Hello World !` पाठ compile करने पर स्लाइड की मुख्य सामग्री के रूप में दिखाई देता है।

### 9.2.1 Frames and Slide Structure / Frames एवं स्लाइड संरचना

The `frame` environment is the core component of a Beamer presentation. It is used to define individual slides in the output PDF. Each frame corresponds to exactly one slide, and all visible content of a slide must be placed inside a frame. Without using the `frame` environment, Beamer cannot generate any slide content.

A frame may contain a title, text, lists, figures, tables, and mathematical equations. In practical usage, presentations are divided into multiple frames so that each slide presents a limited amount of information in a clear and readable form.

frame environment Beamer प्रस्तुति का मुख्य घटक होता है। इसी के माध्यम से अंतिम PDF में प्रत्येक स्लाइड परिभाषित की जाती है। हर frame ठीक एक स्लाइड का प्रतिनिधित्व करता है और स्लाइड की समस्त दृश्य सामग्री frame के भीतर ही लिखी जाती है। frame के बिना Beamer में स्लाइड बनाना संभव नहीं होता।

एक frame में शीर्षक, पाठ, सूचियाँ, चित्र, सारणियाँ तथा गणितीय समीकरण शामिल किए जा सकते हैं। व्यवहार में प्रस्तुति को कई frames में विभाजित किया जाता है ताकि प्रत्येक स्लाइड पर सीमित और स्पष्ट जानकारी प्रस्तुत की जा सके।

```
\begin{frame}{Title of the Slide}
\framesubtitle{Basic Structure}
Hello World !
\end{frame}
```

In this program, the command `\begin{frame}{Title of the Slide}` starts a new slide and sets the slide title. The text `Hello World !` is written inside the frame, therefore it appears as the main content of the slide. When the document is compiled, this frame produces one slide with the given title and content. Adding more frame environments will generate additional slides in the presentation.

इस प्रोग्राम में `\begin{frame}{Title of the Slide}` एक नई स्लाइड शुरू करता है और उसका शीर्षक निर्धारित करता है। `Hello World !` पाठ frame के भीतर लिखा गया है, इसलिए यह स्लाइड की मुख्य सामग्री के रूप में दिखाई देता है। दस्तावेज़ को compile करने पर यह frame एक स्लाइड उत्पन्न करता है। अधिक frame जोड़ने से प्रस्तुति में नई स्लाइड्स प्राप्त होती हैं।

## 9.2.2 Frame Titles and Subtitles / Frame शीर्षक एवं उपशीर्षक

A frame title identifies the main topic of a slide and is provided as an argument to the frame environment. A subtitle gives additional information and is defined using the `\framesubtitle` command inside the frame.

Frame का शीर्षक स्लाइड के मुख्य विषय को दर्शाता है और frame environment के साथ दिया जाता है। उपशीर्षक अतिरिक्त जानकारी प्रदान करता है और इसे

`\framesubtitle` command द्वारा frame के भीतर परिभाषित किया जाता है।

When this program in section 9.2 is compiled, it produces a slide with the title `Introduction to Beamer` and the subtitle `Basic Structure`.

अनुभाग 9.2 में दिए गए प्रोग्राम को compile करने पर एक स्लाइड प्राप्त होती है, जिसमें शीर्षक `Introduction to Beamer` तथा उपशीर्षक `Basic Structure` प्रदर्शित होता है।

## 9.2.3 Multiple Frames and Frame Numbering / Multiple Frames एवं Frame Numbering

A Beamer presentation can contain multiple frames, where each frame creates a separate slide in the output PDF. Frames are written one after another inside the document environment. By default, Beamer automatically numbers the frames, and the slide number appears in the footer when a suitable theme is used.

Frame numbering helps the audience track progress during a presentation and allows the presenter to refer to specific slides. Beamer also provides commands to control frame numbering, such as excluding a frame from numbering or restarting the frame count when required.

Beamer प्रस्तुति में कई frames शामिल किए जा सकते हैं, जहाँ प्रत्येक frame आउटपुट PDF में एक अलग स्लाइड बनाता है। frames को document environment के भीतर क्रम से लिखा जाता है। डिफ़ॉल्ट रूप से Beamer frames को स्वतः संख्या प्रदान करता है, और उपयुक्त theme के उपयोग पर स्लाइड संख्या footer में दिखाई देती है।

Frame numbering श्रोताओं को प्रस्तुति की प्रगति समझने में सहायता करती है तथा प्रस्तोता को किसी विशेष स्लाइड का संदर्भ देने में उपयोगी होती है। आवश्यकता होने पर Beamer में frame numbering को नियंत्रित करने के लिए विशेष commands भी उपलब्ध हैं, जैसे किसी frame को numbering से बाहर रखना या frame count को पुनः प्रारंभ करना।

### Input (LaTeX Code) / इनपुट (LaTeX कोड)

```
\documentclass{beamer}
\usetheme{default}

\begin{document}

\begin{frame}{First Slide}
```

```

This is the first frame.
\end{frame}

\begin{frame}{Second Slide}
This is the second frame.
\end{frame}

\begin{frame}{Third Slide}
This is the third frame.
\end{frame}

\end{document}

```

## 9.3 Writing Content in Beamer / Beamer में सामग्री लेखन

Writing content in Beamer is very similar to writing content in a standard  $\LaTeX$  document, as described in the previous chapters. Text, lists, figures, tables, and mathematical expressions are written using the same  $\LaTeX$  commands and environments. The main difference is that all visible content must be placed inside the `frame` environment so that it appears on a slide. Due to this similarity, users who are already familiar with  $\LaTeX$  documentation can easily write and manage presentation content in Beamer.

Beamer में सामग्री लेखन की प्रक्रिया पहले के अध्यायों में वर्णित  $\LaTeX$  दस्तावेज़ लेखन के समान होती है। पाठ, सूचियाँ, चित्र, सारणियाँ तथा गणितीय अभिव्यक्तियाँ वही  $\LaTeX$  commands और environments का उपयोग करके लिखी जाती हैं। मुख्य अंतर यह है कि सभी दृश्य सामग्री को `frame` environment के भीतर रखना आवश्यक होता है ताकि वह स्लाइड पर प्रदर्शित हो सके। इस समानता के कारण  $\LaTeX$  से परिचित उपयोगकर्ता Beamer में प्रस्तुति सामग्री आसानी से तैयार कर सकते हैं।

### 9.3.1 Types of Blocks / Blocks के प्रकार

Beamer provides different types of block environments to present content in a structured and visually distinct manner. Each block type is used for a specific purpose, such as highlighting important information, showing examples, or drawing attention to warnings. These blocks help improve clarity and readability on slides.

Beamer में विभिन्न प्रकार के block environments उपलब्ध होते हैं, जिनका उपयोग सामग्री को संरचित और दृश्य रूप से अलग दिखाने के लिए किया जाता है। प्रत्येक block का एक विशेष उद्देश्य होता है, जैसे महत्वपूर्ण जानकारी को उजागर करना, उदाहरण प्रस्तुत करना या चेतावनी दर्शाना। इन blocks के प्रयोग से स्लाइड्स अधिक स्पष्ट और पठनीय बनती हैं।

#### Common Block Types

##### 1. Block :

The `block` environment is used for normal information that needs to be highlighted.

`block` environment का उपयोग सामान्य लेकिन महत्वपूर्ण जानकारी को प्रदर्शित करने के लिए किया जाता है।

#### Input (LaTeX Code) / इनपुट (LaTeX कोड)

```

\documentclass{beamer}
\usetheme{default}
\begin{document}

\begin{frame}{Block}

\begin{block}{Definition}
This is a normal block.
\end{block}

\end{frame}

\end{document}

```

##### 2. Alert Block

The `alertblock` environment is used to emphasize warnings or critical points.

`alertblock` environment का उपयोग चेतावनी या अत्यंत महत्वपूर्ण बिंदुओं को दर्शाने के लिए किया जाता है।

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```

\documentclass{beamer}
\usetheme{default}
\begin{document}

\begin{frame}{alertblock}

\begin{alertblock}{Important}
This is an alert block.
\end{alertblock}

\end{frame}

\end{document}

```

**3. Example Block**

The `exampleblock` environment is used to present examples or illustrative content.

`exampleblock` environment का उपयोग उदाहरण या व्याख्यात्मक सामग्री प्रस्तुत करने के लिए किया जाता है।

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```

\documentclass{beamer}
\usetheme{default}
\begin{document}

\begin{frame}{exampleblock}

\begin{exampleblock}{Example}
This is an example block.
\end{exampleblock}

\end{frame}

\end{document}

```

**9.4 Themes in Beamer / Beamer में थीम्स**

Themes in Beamer control the overall appearance and layout of a presentation. They define how slides look, including colors, fonts, headlines, footlines, and the style of content elements. By using themes, a consistent and professional design can be applied to all slides without manually formatting each one.

Beamer में themes का उपयोग प्रस्तुति के समग्र रूप और लेआउट को नियंत्रित करने के लिए किया जाता है। themes स्लाइड्स के रंग, फॉन्ट, हेडलाइन, फुटलाइन तथा सामग्री की शैली को निर्धारित करते हैं। themes के माध्यम से बिना प्रत्येक स्लाइड को अलग-अलग फॉर्मेट किए पूरी प्रस्तुति में एकसमान और पेशेवर डिज़ाइन लागू किया जा सकता है।

**9.4.1 Overview of Beamer Theme Architecture / Beamer Theme संरचना का अवलोकन**

Beamer follows a modular theme architecture. Instead of using a single monolithic theme, the appearance of a presentation is controlled by different types of themes such as main themes, inner themes, outer themes, color themes, and font themes. Each theme type is responsible for a specific visual aspect of the slides.

Beamer में theme की संरचना modular होती है। किसी एक ही theme पर निर्भर रहने के बजाय प्रस्तुति का रूप विभिन्न प्रकार के themes द्वारा नियंत्रित किया जाता है, जैसे main themes, inner themes, outer themes, color themes तथा font themes। प्रत्येक theme प्रकार स्लाइड के किसी विशिष्ट दृश्य पहलू को नियंत्रित करता है।

**9.4.2 Main Themes / Main Themes**

A main theme provides a predefined overall layout for the presentation. It usually combines suitable inner, outer, color, and font settings. Using a main theme is the quickest way to give a presentation a complete and coherent look.

Main theme प्रस्तुति के लिए एक पूर्वनिर्धारित समग्र लेआउट प्रदान करता है। इसमें सामान्यतः inner, outer, color और font settings का संयोजन शामिल होता है। main theme का उपयोग करके प्रस्तुति को शीघ्र ही एक पूर्ण और सुसंगत रूप दिया जा सकता है।

### Input (LaTeX Code) / इनपुट (LaTeX कोड)

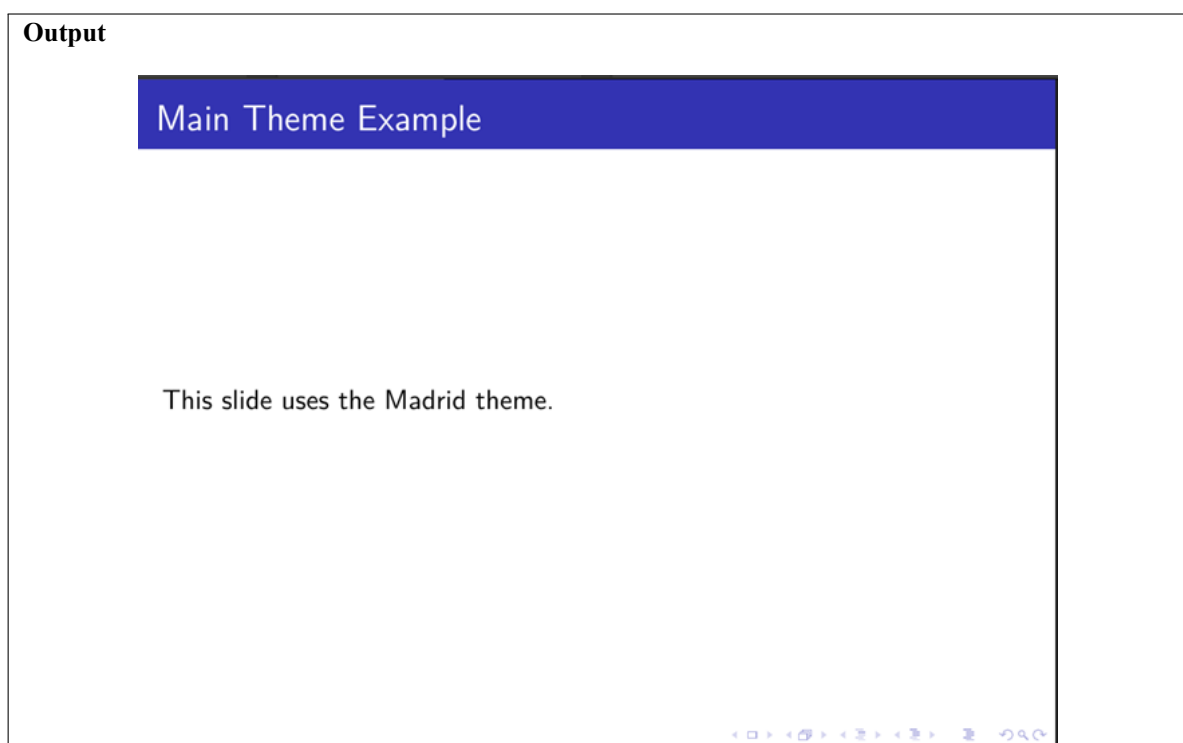
```
\documentclass{beamer}
\usetheme{Madrid}

\begin{document}

\begin{frame}{Main Theme Example}
This slide uses the Madrid theme.
\end{frame}

\end{document}
```

### Output (PDF Result) / आउटपुट (PDF परिणाम)



### List of Main Themes / Main Themes की सूची

The following are the standard main themes provided by Beamer:

- AnnArbor
- Antibes
- Bergen
- Berkeley
- Berlin
- Boadilla
- CambridgeUS
- Copenhagen

- Darmstadt
- Dresden
- Frankfurt
- Goettingen
- Hannover
- Ilmenau
- JuanLesPins
- Madrid
- Malmoe
- Marburg
- Montpellier
- PaloAlto
- Pittsburgh
- Rochester
- Singapore
- Szeged
- Warsaw

Beamer में उपलब्ध standard main themes निम्नलिखित हैं। इन themes का उपयोग प्रस्तुति के समग्र लेआउट और संरचना को निर्धारित करने के लिए किया जाता है। उपयुक्त main theme का चयन करके पूरी प्रस्तुति को एक सुसंगत और पेशेवर रूप दिया जा सकता है।

### 9.4.3 Inner Themes / Inner Themes

Inner themes control the appearance of content inside the slide, such as blocks, lists, and item markers. They mainly affect how information is presented within the main content area.

Inner themes स्लाइड के भीतर की सामग्री के स्वरूप को नियंत्रित करते हैं, जैसे blocks, सूचियाँ और item markers। ये मुख्य रूप से स्लाइड के कंटेंट क्षेत्र में जानकारी के प्रदर्शन को प्रभावित करते हैं।

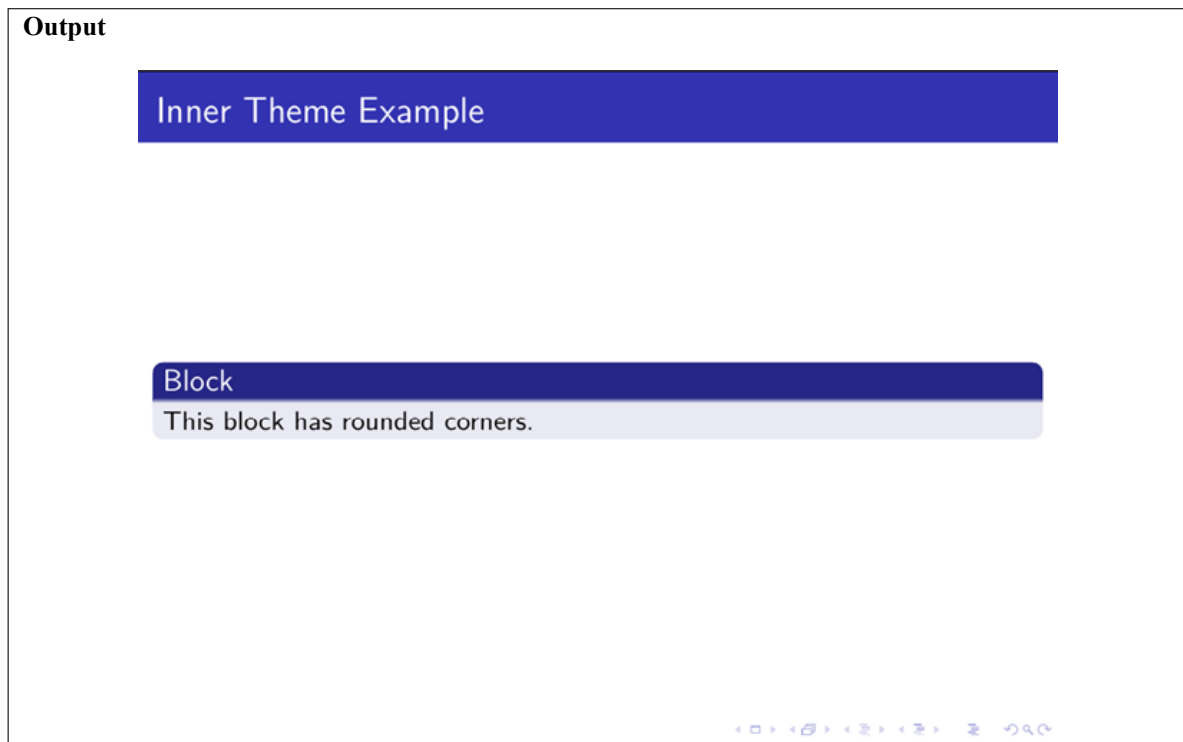
#### Input (LaTeX Code) / इनपुट (LaTeX कोड)

```
\documentclass{beamer}
\usetheme{Madrid}
\useinnertheme{rounded}
\setbeamertemplate{blocks}[rounded]

\begin{document}

\begin{frame}{Inner Theme Example}
\begin{block}{Block}
This block has rounded corners.
\end{block}
\end{frame}

\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)****List of Inner Themes / Inner Themes की सूची**

Beamer provides the following standard inner themes, which control the appearance of content elements such as blocks, lists, and item markers:

- default
- rounded
- circles

Beamer में उपलब्ध standard inner themes उपर्युक्त हैं। ये themes मुख्य रूप से स्लाइड के भीतर की सामग्री, जैसे blocks, सूचियाँ और item markers के स्वरूप को नियंत्रित करते हैं। उपयुक्त inner theme का चयन करने से स्लाइड की पठनीयता और दृश्य स्पष्टता बेहतर होती है।

**9.4.4 Outer Themes / Outer Themes**

Outer themes define the outer structure of slides, including headlines, footlines, navigation bars, and sidebars. They influence how navigation and structural information is displayed during the presentation.

Outer themes स्लाइड की बाहरी संरचना को निर्धारित करते हैं, जिनमें headlines, footlines, navigation bars और sidebars शामिल होते हैं। ये themes प्रस्तुति के दौरान नेविगेशन और संरचनात्मक जानकारी के प्रदर्शन को प्रभावित करते हैं।

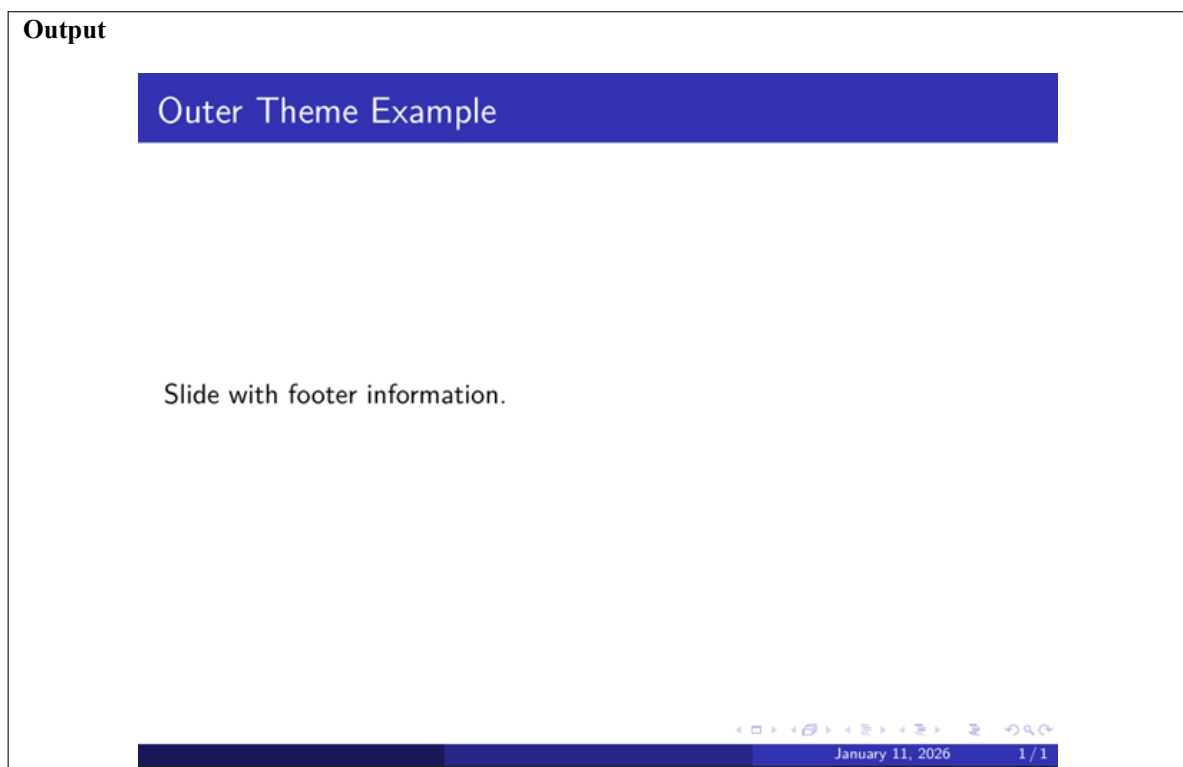
**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{beamer}
\usetheme{madrid}
\useoutertheme{infolines}

\begin{document}

\begin{frame}{Outer Theme Example}
Slide with footer information.
\end{frame}

\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)****List of Outer Themes / Outer Themes की सूची**

Outer themes in Beamer control the outer structure of slides, such as headlines, footlines, navigation bars, sidebars, and frame navigation. The following are the standard outer themes provided by Beamer:

- default
- infolines
- split
- shadow
- sidebar
- miniframes
- smoothbars
- smoothtree
- tree

Beamer में outer themes का उपयोग स्लाइड की बाहरी संरचना को नियंत्रित करने के लिए किया जाता है, जैसे headline, footline, navigation bar, sidebar तथा frame navigation। Beamer द्वारा प्रदान किए गए standard outer themes उपर्युक्त हैं। उपयुक्त outer theme का चयन करने से प्रस्तुति की संरचना और नेविगेशन अधिक प्रभावी बनती है।

**9.4.5 Color Themes / Color Themes**

Color themes specify the color scheme used in a presentation. They control the colors of titles, text, blocks, alerts, and navigation elements. Using an appropriate color theme improves readability and visual consistency.

Color themes प्रस्तुति में प्रयुक्त रंग योजना को निर्धारित करते हैं। ये शीर्षकों, पाठ, blocks, alerts तथा navigation elements के रंग नियंत्रित करते हैं। उपयुक्त color theme का चयन पठनीयता और दृश्य एकरूपता को बेहतर बनाता है।

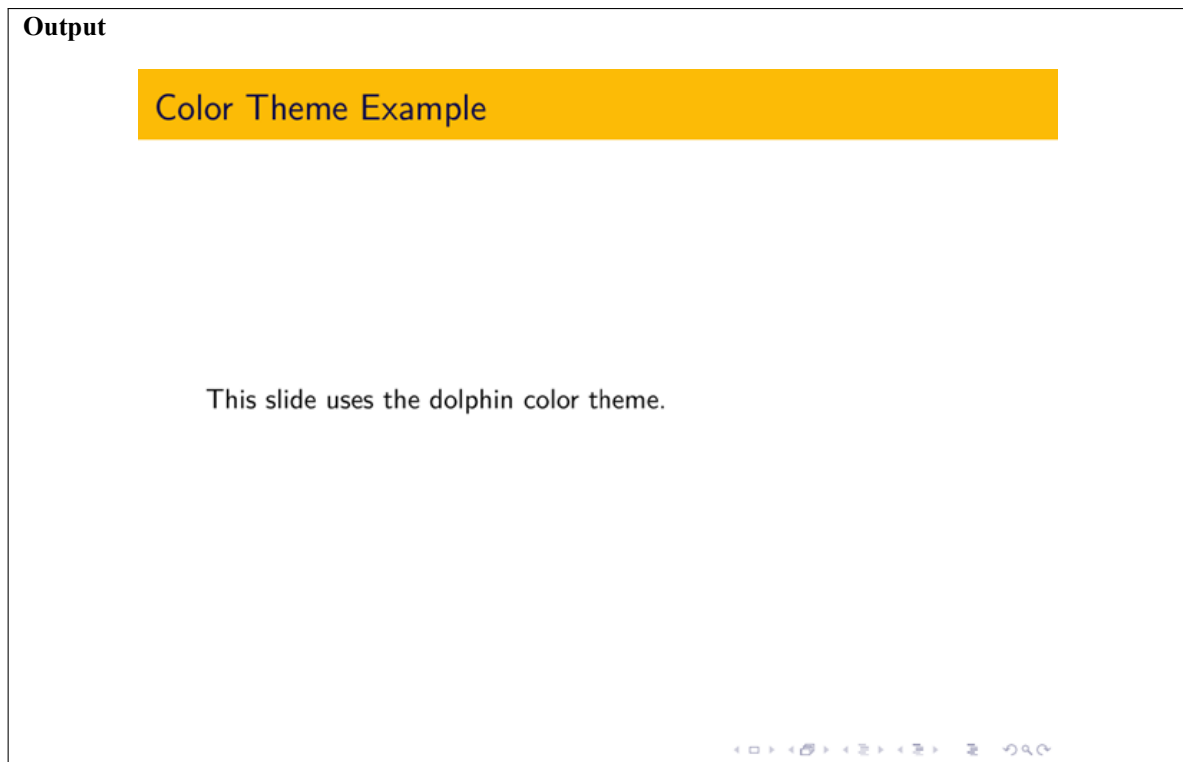
**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{beamer}
\usetheme{default}
\usecolortheme{crane}

\begin{document}

\begin{frame}{Color Theme Example}
This slide uses the dolphin color theme.
\end{frame}

\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)****List of Color Themes / Color Themes की सूची**

Color themes in Beamer control the color scheme of presentation elements such as titles, text, blocks, alerts, and navigation components. The following are the commonly used standard color themes provided by Beamer:

- default
- albatross
- beaver
- beetle
- crane
- dolphin
- dove
- fly
- lily

- orchid
- rose
- seagull
- seahorse
- whale
- wolverine

Beamer में color themes का उपयोग प्रस्तुति के विभिन्न तत्वों जैसे शीर्षक, पाठ, blocks, alerts तथा navigation components के रंग निर्धारित करने के लिए किया जाता है। Beamer द्वारा प्रदान किए गए सामान्यतः उपयोग होने वाले standard color themes उपर्युक्त हैं। उपयुक्त color theme का चयन प्रस्तुति की पठनीयता और दृश्य आकर्षण को बेहतर बनाता है।

### 9.4.6 Font Themes / Font Themes

Font themes control the fonts used for different elements such as titles, body text, and structural elements. They help maintain typographical consistency throughout the presentation.

Font themes प्रस्तुति में विभिन्न तत्वों जैसे शीर्षक, मुख्य पाठ और संरचनात्मक भागों के लिए प्रयुक्त फ़ॉन्ट को नियंत्रित करते हैं। ये पूरी प्रस्तुति में टाइपोग्राफ़िक एकरूपता बनाए रखने में सहायता करते हैं।

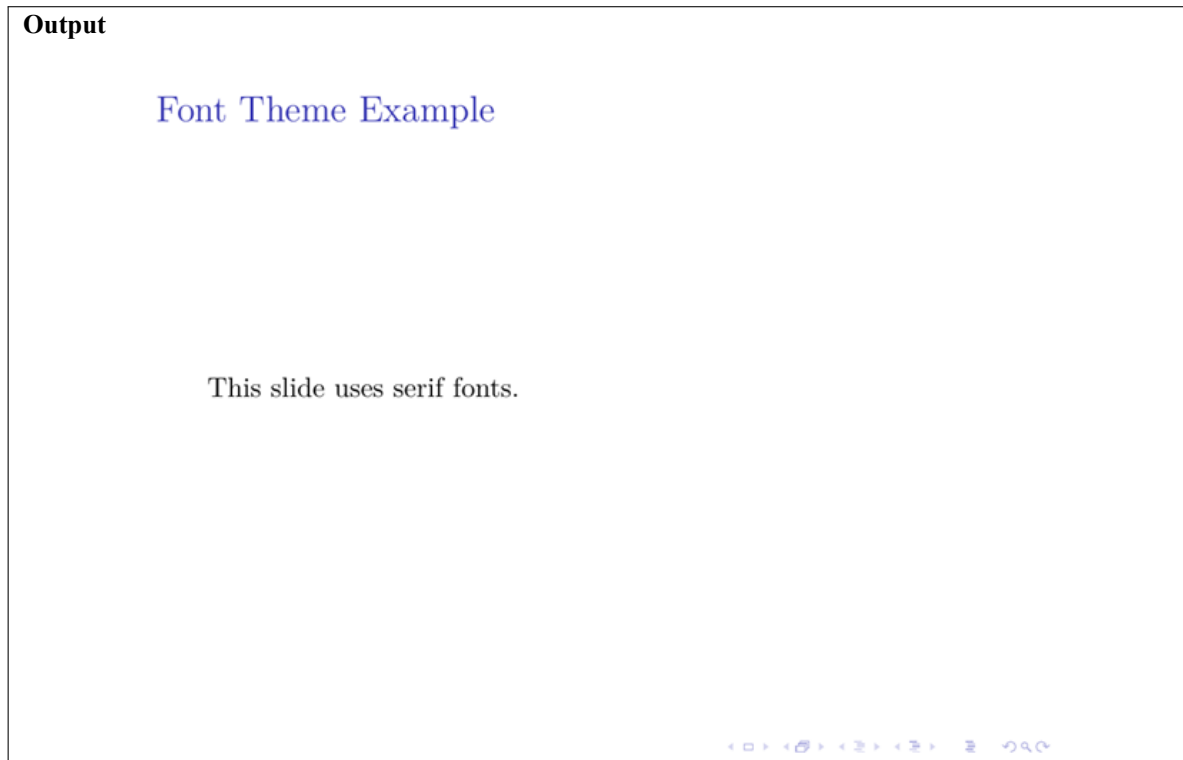
#### Input (LaTeX Code) / इनपुट (LaTeX कोड)

```
\documentclass{beamer}
\usetheme{default}
\usefonttheme{serif}

\begin{document}

\begin{frame}{Font Theme Example}
This slide uses serif fonts.
\end{frame}

\end{document}
```

**Output (PDF Result) / आउटपुट (PDF परिणाम)****List of Font Themes / Font Themes की सूची**

Font themes in Beamer control the font style used for titles, body text, and structural elements in a presentation. The following are the standard font themes provided by Beamer:

- default
- professionalfonts
- serif
- structure

Beamer में font themes का उपयोग प्रस्तुति में शीर्षक, मुख्य पाठ और संरचनात्मक तत्वों के लिए प्रयुक्त फ़ॉन्ट शैली को नियंत्रित करने के लिए किया जाता है। Beamer द्वारा प्रदान किए गए standard font themes उपर्युक्त हैं। उपयुक्त font theme का चयन प्रस्तुति की टाइपोग्राफ़िक एकरूपता और पठनीयता को बेहतर बनाता है।

**9.4.7 Combining and Customizing Themes / Themes का संयोजन एवं अनुकूलन**

Beamer allows different theme types to be combined to create a custom appearance. Users can select a main theme and then modify or override specific aspects using inner, outer, color, or font themes. This flexibility makes it possible to design presentations that match specific academic or institutional requirements.

Beamer में विभिन्न प्रकार के themes को संयोजित करके इच्छानुसार डिज़ाइन तैयार किया जा सकता है। उपयोगकर्ता किसी main theme का चयन करके inner, outer, color या font themes के माध्यम से विशेष पहलुओं को संशोधित कर सकते हैं। यह लचीलापन शैक्षणिक या संस्थागत आवश्यकताओं के अनुसार प्रस्तुति डिज़ाइन करने में सहायक होता है।

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```
\documentclass{beamer}

\usetheme{CambridgeUS}
\useinnertheme{rounded}
\useoutertheme{infolines}
```

```

\usecolortheme{dolphin}
\usefonttheme{serif}

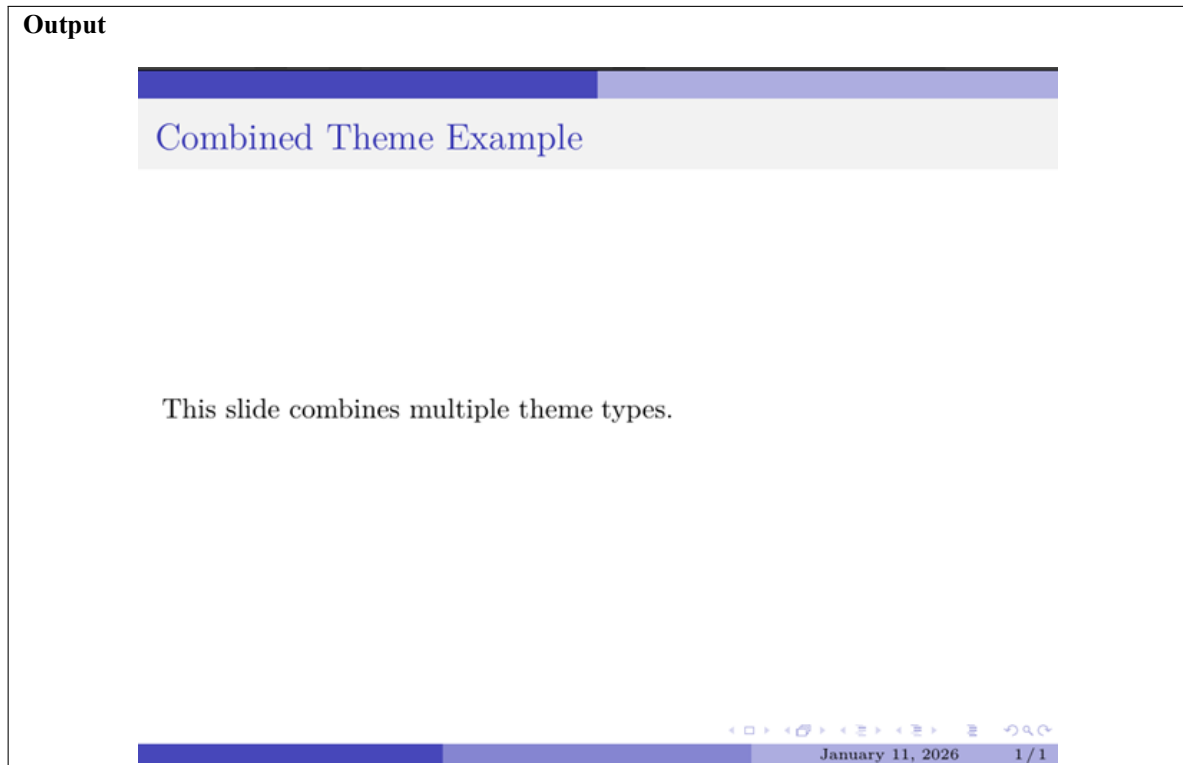
\begin{document}

\begin{frame}{Combined Theme Example}
This slide combines multiple theme types.
\end{frame}

\end{document}

```

### Output (PDF Result) / आउटपुट (PDF परिणाम)



## 9.5 Navigation and Layout

Navigation and layout features in Beamer help organize slides and guide the audience through the presentation. These features control how structural information such as slide titles, section progress, and navigation elements are displayed. Proper use of navigation and layout improves clarity and makes presentations easier to follow.

Beamer में navigation और layout सुविधाएँ स्लाइड्स को व्यवस्थित करने और श्रोताओं को प्रस्तुति के प्रवाह को समझाने में सहायता करती हैं। ये सुविधाएँ शीर्षक, अनुभाग प्रगति तथा नेविगेशन तत्वों के प्रदर्शन को नियंत्रित करती हैं। navigation और layout का उचित उपयोग प्रस्तुति को अधिक स्पष्ट और सुव्यवस्थित बनाता है।

### 9.5.1 Headlines and Footlines / Headlines एवं Footlines

Headlines and footlines are used to display information at the top and bottom of a slide. Headlines often show section or subsection titles, while footlines commonly display slide numbers, author names, or short titles. They help the audience understand the current position within the presentation.

Headlines और footlines स्लाइड के ऊपर और नीचे जानकारी प्रदर्शित करने के लिए उपयोग किए जाते हैं। headlines में सामान्यतः section या subsection के शीर्षक दिखाई देते हैं, जबकि footlines में स्लाइड संख्या, लेखक का नाम या संक्षिप्त शीर्षक दर्शाया जाता है। ये प्रस्तुति में वर्तमान स्थिति को समझने में सहायता करते हैं।

### Input (LaTeX Code) / इनपुट (LaTeX कोड)

```

\documentclass{beamer}
\usetheme{warsaw}
\useoutertheme{infolines}

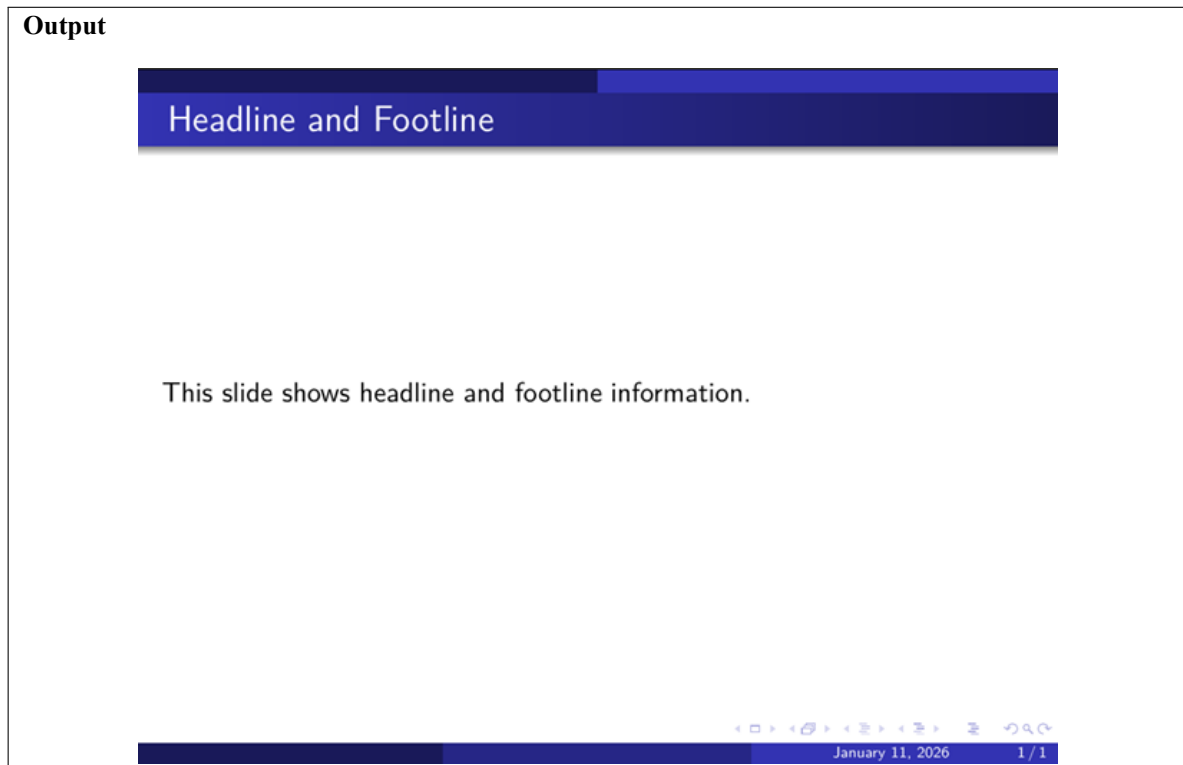
\begin{document}

\begin{frame}{Headline and Footline}
This slide shows headline and footline information.
\end{frame}

\end{document}

```

### Output (PDF Result) / आउटपुट (PDF परिणाम)



### 9.5.2 Navigation Symbols / Navigation Symbols

Navigation symbols provide clickable icons for moving between slides during a presentation. These symbols allow quick navigation to the previous or next slide and can be enabled or disabled as required. In practice, they are often turned off for clean and distraction-free slides.

Navigation symbols प्रस्तुति के दौरान स्लाइड्स के बीच जाने के लिए clickable icons प्रदान करते हैं। इनके माध्यम से अगली या पिछली स्लाइड पर शीघ्र पहुँचा जा सकता है। व्यवहार में, स्वच्छ और बिना विचलन वाली स्लाइड्स के लिए इन्हें प्रायः बंद कर दिया जाता है।

#### Input (LaTeX Code) / इनपुट (LaTeX कोड)

```

\documentclass{beamer}
\usetheme{warsaw}

\setbeamertemplate{navigation symbols}{}

\begin{document}

\begin{frame}{Navigation Symbols Disabled}
Navigation symbols are removed.
\end{frame}

\end{document}

```

### 9.5.3 Sections, Subsections, and TOC Slides / Sections, Subsections एवं TOC स्लाइड्स

Sections and subsections are used to logically divide a presentation into topics and subtopics. Beamer can automatically generate table of contents (TOC) slides based on these divisions. TOC slides help the audience understand the structure of the presentation and track progress.

Sections और subsections का उपयोग प्रस्तुति को विषयों और उपविषयों में तार्किक रूप से विभाजित करने के लिए किया जाता है। Beamer इनके आधार पर स्वतः table of contents (TOC) स्लाइड्स बना सकता है। TOC स्लाइड्स श्रोताओं को प्रस्तुति की संरचना समझने और प्रगति को ट्रैक करने में सहायता करती हैं।

#### Input (LaTeX Code) / इनपुट (LaTeX कोड)

```
\documentclass{beamer}
\usetheme{default}

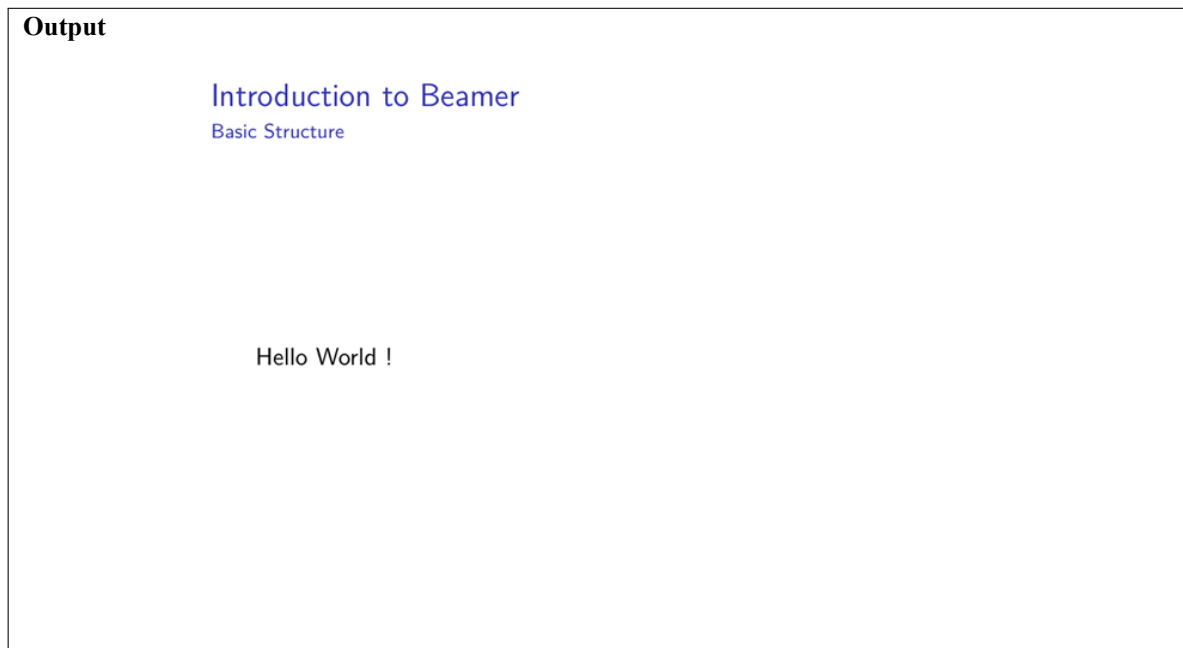
\begin{document}

\section{Introduction}
\begin{frame}{Outline}
\tableofcontents
\end{frame}

\section{Main Topic}
\begin{frame}{First Slide}
Content of the main topic.
\end{frame}

\end{document}
```

#### Output (PDF Result) / आउटपुट (PDF परिणाम)



### 9.5.4 Sidebar and Miniframe Navigation / Sidebar एवं Miniframe Navigation

Sidebar and miniframe navigation provide visual indicators of progress through sections and frames. A sidebar displays section titles along the side of the slide, while miniframes show small markers representing individual frames. These navigation styles are useful for long or highly structured presentations.

Sidebar और miniframe navigation प्रस्तुति में sections और frames की प्रगति को दृश्य रूप से प्रदर्शित करते हैं। sidebar में स्लाइड के किनारे section शीर्षक दिखाई देते हैं, जबकि miniframes छोटे markers के रूप में frames को दर्शाते हैं। ये नेविगेशन शैलियाँ लंबी या अधिक संरचित प्रस्तुतियों के लिए उपयोगी होती हैं।

**Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```

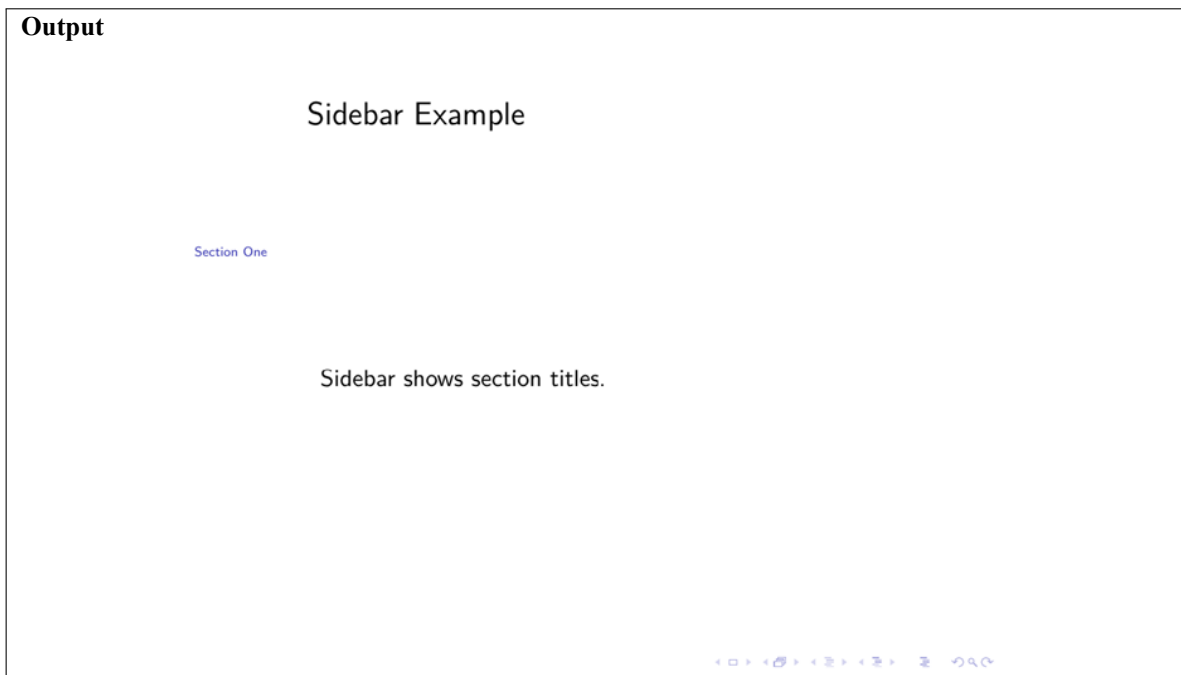
\documentclass{beamer}
\usetheme{default}
\useoutertheme{sidebar}

\begin{document}

\section{Section One}
\begin{frame}{Sidebar Example}
Sidebar shows section titles.
\end{frame}

\end{document}

```

**Output (PDF Result) / आउटपुट (PDF परिणाम)****Input (LaTeX Code) / इनपुट (LaTeX कोड)**

```

\documentclass{beamer}
\usetheme{default}
\useoutertheme{miniframes}

\begin{document}

\section{Section One}
\begin{frame}{Miniframe Example}
Miniframes show slide progress.
\end{frame}

\begin{frame}{Next Slide}
Another frame in the same section.
\end{frame}

\end{document}

```



### 9.6.1 Introduction to Overlays / Overlays का परिचय

An overlay refers to a technique where a single frame is shown in multiple steps. Each step reveals additional content on the same slide. Overlays are widely used in teaching and technical presentations.

Overlay एक ऐसी तकनीक है जिसमें एक ही frame को कई चरणों में प्रदर्शित किया जाता है। प्रत्येक चरण में स्लाइड पर नई सामग्री दिखाई देती है। overlays का उपयोग शिक्षण और तकनीकी प्रस्तुतियों में व्यापक रूप से किया जाता है।

### 9.6.2 Overlay Specifications / Overlay Specifications

Overlay specifications are written using angle brackets and numbers to control when an element appears. They are commonly used with lists, blocks, and text to define the order of appearance.

Overlay specifications को angle brackets और संख्याओं की सहायता से लिखा जाता है, जिससे यह नियंत्रित किया जा सके कि कोई तत्व कब दिखाई देगा। इनका उपयोग सामान्यतः सूचियों, blocks और पाठ के साथ किया जाता है।

#### Input (LaTeX Code) / इनपुट (LaTeX कोड)

```
\begin{frame}{Overlay Specification}
\begin{itemize}
  \item<1-> First point
  \item<2-> Second point
  \item<3-> Third point
\end{itemize}
\end{frame}
```

### 9.6.3 The \pause Command / \pause Command

The \pause command is the simplest way to create overlays. Each use of \pause stops the slide and reveals the next content in a new step.

\pause command overlays बनाने का सबसे सरल तरीका है। \pause का प्रत्येक प्रयोग स्लाइड को एक चरण पर रोकता है और अगली सामग्री अगले चरण में प्रदर्शित होती है।

#### Input (LaTeX Code) / इनपुट (LaTeX कोड)

```
\begin{frame}{Pause Example}
First point
\pause
Second point
\pause
Third point
\end{frame}
```

### 9.6.4 Incremental Lists and Blocks / Incremental Lists एवं Blocks

Incremental lists and blocks reveal list items or blocks one by one. This technique is useful when explaining stepwise procedures or algorithms.

Incremental lists और blocks में सूची के बिंदु या blocks एक-एक करके दिखाई देते हैं। यह तकनीक चरणबद्ध प्रक्रियाओं या algorithms को समझाने में उपयोगी होती है।

#### Example

```
\begin{frame}{Incremental Block}
\begin{block}{Steps}
\begin{itemize}[<+>]
  \item Step one
  \item Step two
  \item Step three
\end{itemize}
\end{block}
\end{frame}
```

### 9.6.5 Best Practices for Animations / Animations के लिए सर्वोत्तम अभ्यास

Animations and overlays should be used sparingly and only when they improve understanding. Overusing animations can distract the audience and reduce clarity. In practice, overlays are best suited for lists, definitions, and stepwise explanations.

Animations और overlays का उपयोग सीमित रूप से और केवल तब करना चाहिए जब वे समझ को बेहतर बनाते हों। अत्यधिक animations श्रोताओं का ध्यान भटका सकते हैं और स्पष्टता को कम कर सकते हैं। व्यवहार में, overlays का उपयोग सूचियों, परिभाषाओं और चरणबद्ध व्याख्याओं के लिए सबसे उपयुक्त होता है।

# 10. Online Resources / ऑनलाइन संसाधन

Writing in LaTeX becomes much easier when you know where to find the right tools. This chapter introduces practical and reliable online resources that students, researchers, and teachers can use to write, edit, manage references, create tables, and design equations efficiently. Instead of just listing websites, this section explains how you can actually use them in your daily academic work.

LaTeX में लेखन करना तब कहीं अधिक आसान हो जाता है जब आपको सही उपकरण कहीं मिलेंगे, यह पता हो। यह अध्याय उन व्यावहारिक और विश्वसनीय ऑनलाइन संसाधनों का परिचय कराता है जिनका उपयोग छात्र, शोधकर्ता और शिक्षक लेखन, संपादन, संदर्भ प्रबंधन, तालिकाएँ बनाने और समीकरणों को प्रभावी ढंग से तैयार करने के लिए कर सकते हैं। केवल वेबसाइटों की सूची देने के बजाय, यह अनुभाग यह भी समझाता है कि आप इन संसाधनों का उपयोग अपने दैनिक शैक्षणिक कार्य में वास्तव में कैसे कर सकते हैं।

## 10.1 Online LaTeX Editors / ऑनलाइन LaTeX संपादक

### 10.1.1 Overleaf

Overleaf is one of the most popular online LaTeX editors. It runs directly in your web browser, so you do not need to install anything on your computer.

Overleaf सबसे लोकप्रिय ऑनलाइन LaTeX संपादकों में से एक है। यह सीधे आपके वेब ब्राउज़र में चलता है, इसलिए आपको अपने कंप्यूटर पर कोई सॉफ़्टवेयर इंस्टॉल करने की आवश्यकता नहीं होती।

**How to Use It in Practice:** / व्यवहार में इसका उपयोग कैसे करें:

First, create a free account. After logging in, click on “New Project.” You can either start with a blank project or choose from a wide range of templates such as journal articles, conference papers, theses, resumes, and reports.

The screen is divided into two parts: On the left, you write your LaTeX code. On the right, you see the compiled PDF output.

As you type, Overleaf automatically compiles your document and shows you the updated PDF. If there is an error, it highlights the issue and helps you locate the problem.

You can upload images, bibliography files (.bib), or additional chapter files. For collaborative research, you can share your project with co-authors, and everyone can edit the document in real time.

सबसे पहले, एक निःशुल्क खाता बनाएँ। लॉग-इन करने के बाद “New Project” पर क्लिक करें। आप या तो एक खाली प्रोजेक्ट से शुरुआत कर सकते हैं या विभिन्न टेम्पलेट्स में से चुन सकते हैं, जैसे जर्नल लेख, कॉन्फ्रेंस पेपर, थीसिस, रेज़्यूमे और रिपोर्ट आदि।

स्क्रीन दो भागों में विभाजित होती है: बाएँ भाग में आप अपना LaTeX कोड लिखते हैं। दाएँ भाग में संकलित (compiled) PDF आउटपुट दिखाई देता है।

जैसे-जैसे आप टाइप करते हैं, Overleaf स्वतः ही आपके दस्तावेज़ को संकलित करता है और अद्यतन PDF दिखाता है। यदि कोई त्रुटि होती है, तो वह उसे हाइलाइट करता है और समस्या का स्थान खोजने में आपकी सहायता करता है।

आप चित्र (images), संदर्भ सूची की फ़ाइलें (.bib), या अतिरिक्त अध्याय फ़ाइलें अपलोड कर सकते हैं। सहयोगात्मक शोध के लिए आप अपने प्रोजेक्ट को सह-लेखकों के साथ साझा कर सकते हैं, और सभी लोग वास्तविक समय (real time) में दस्तावेज़ को संपादित कर सकते हैं।

**When Should You Use Overleaf?** / Overleaf का उपयोग कब करना चाहिए?

Overleaf is especially useful for:

- Writing research papers
- Preparing a Ph.D. thesis using university templates
- Working collaboratively with supervisors or co-authors
- Teaching students who are new to LaTeX

Overleaf विशेष रूप से निम्नलिखित कार्यों के लिए उपयोगी है:

- शोध-पत्र लिखने के लिए
- विश्वविद्यालय के टेम्पलेट का उपयोग करते हुए पी.एच.डी. थीसिस तैयार करने के लिए
- मार्गदर्शक या सह-लेखकों के साथ सहयोगात्मक रूप से कार्य करने के लिए
- उन विद्यार्थियों को पढ़ाने के लिए जो LaTeX में नए हैं

### 10.1.2 CoCalc

CoCalc is another online platform that supports LaTeX, along with mathematical computation tools such as Jupyter Notebook and SageMath.

CoCalc एक अन्य ऑनलाइन प्लेटफॉर्म है जो LaTeX का समर्थन करता है, साथ ही Jupyter Notebook और SageMath जैसे गणनात्मक गणितीय उपकरण भी प्रदान करता है।

**How to Use It:** / इसे कैसे उपयोग करें:

After creating an account, start a new project and create a LaTeX file inside it. You can edit and compile the document within the platform. CoCalc is particularly helpful when your document includes heavy mathematical computation or when you want to combine coding and documentation in one place.

It is commonly used in mathematics, engineering, and scientific research environments.

खाता बनाने के बाद एक नया प्रोजेक्ट प्रारंभ करें और उसके भीतर एक LaTeX फ़ाइल बनाएँ। आप उसी प्लेटफॉर्म के अंदर दस्तावेज़ को संपादित तथा संकलित कर सकते हैं। CoCalc विशेष रूप से तब उपयोगी होता है जब आपके दस्तावेज़ में जटिल गणनात्मक कार्य शामिल हों या जब आप कोडिंग और दस्तावेज़ीकरण को एक ही स्थान पर संयोजित करना चाहते हों। यह सामान्यतः गणित, अभियांत्रिकी तथा वैज्ञानिक अनुसंधान के वातावरण में उपयोग किया जाता है।

## 10.2 Official Documentation and Learning Resources / आधिकारिक प्रलेखन एवं अध्ययन संसाधन

### 10.2.1 CTAN (Comprehensive TeX Archive Network) / सीटैन (समग्र टेक्स अभिलेख नेटवर्क)

CTAN is the official repository of LaTeX packages. Whenever you need a new package—for example, for drawing diagrams, formatting tables, or creating advanced mathematical symbols—you can search for it on CTAN.

CTAN, LaTeX पैकेजों का आधिकारिक भंडार (repository) है। जब भी आपको किसी नए पैकेज की आवश्यकता हो—जैसे आरेख (diagrams) बनाने, तालिकाओं को स्वरूपित करने, या उन्नत गणितीय प्रतीकों को तैयार करने के लिए—आप उसे CTAN पर खोज सकते हैं।

**How to Use It:** / इसे कैसे उपयोग करें:

Search for the package name. Each package page provides:

- A description of what it does
- Installation instructions
- Documentation in PDF format
- Example usage

If you are using an offline LaTeX distribution (such as TeX Live or MiKTeX), CTAN helps you understand what packages are available and how they work.

पैकेज के नाम से खोज करें। प्रत्येक पैकेज के पृष्ठ पर निम्नलिखित जानकारी उपलब्ध होती है:

- उसके कार्य का विवरण
- स्थापना (Installation) के निर्देश
- PDF प्रारूप में प्रलेखन (Documentation)

- उदाहरण सहित उपयोग विधि

यदि आप ऑफ़लाइन LaTeX वितरण (जैसे TeX Live या MiKTeX) का उपयोग कर रहे हैं, तो CTAN आपको उपलब्ध पैकेजों तथा उनके कार्य करने के तरीके को समझने में सहायता करता है।

## 10.3 Online Table Generators / ऑनलाइन तालिका जनरेटर

### 10.3.1 www.tablesgenerator.com

Creating complex tables manually in LaTeX can be challenging. Online table generators simplify this process. LaTeX में जटिल तालिकाएँ (tables) को मैन्युअली तैयार करना चुनौतीपूर्ण हो सकता है। ऑनलाइन टेबल जनरेटर इस प्रक्रिया को सरल बना देते हैं।

**How to Use It:** इसे कैसे उपयोग करें:

You visually design your table by selecting rows, columns, borders, alignment, and merged cells. Once the design is complete, the tool automatically generates the corresponding LaTeX code. You can then copy and paste this code into your document.

आप पंक्तियों (rows), स्तंभों (columns), सीमाओं (borders), संरेखण (alignment) तथा संयोजित कक्षों (merged cells) का चयन करके अपनी तालिका को दृश्य रूप में तैयार करते हैं। जब डिज़ाइन पूर्ण हो जाता है, तो यह उपकरण स्वचालित रूप से उसके अनुरूप LaTeX कोड उत्पन्न कर देता है। इसके बाद आप उस कोड को कॉपी करके अपने दस्तावेज़ में पेस्ट कर सकते हैं।

This is extremely useful for:

- Research data tables
- Comparison tables
- Multi-row or multi-column formatted tables

यह निम्नलिखित कार्यों के लिए अत्यंत उपयोगी है:

- शोध संबंधी डेटा तालिकाएँ
- तुलनात्मक तालिकाएँ
- बहु-पंक्ति या बहु-स्तंभ स्वरूपित तालिकाएँ

## 10.4 Graphics and Diagram Tools / ग्राफ़िक्स एवं आरेख उपकरण

### 10.4.1 Inkscape

Inkscape is a vector graphics editor useful for creating high-quality diagrams. You can export diagrams as PDF files and include them in your LaTeX document using the `graphicx` package.

Inkscape एक वेक्टर ग्राफ़िक्स संपादक है जो उच्च-गुणवत्ता वाले आरेख (diagrams) बनाने के लिए उपयोगी है। आप अपने आरेखों को PDF फ़ाइल के रूप में निर्यात (export) कर सकते हैं और उन्हें `graphicx` पैकेज का उपयोग करके अपने LaTeX दस्तावेज़ में सम्मिलित कर सकते हैं।

### 10.4.2 TikZ

TikZ is a powerful LaTeX package for drawing diagrams directly within your document. It is especially useful for:

- Flowcharts
- Block diagrams
- Circuit diagrams
- Mathematical illustrations

Although it has a learning curve, it produces highly professional and scalable graphics.

TikZ एक शक्तिशाली LaTeX पैकेज है, जिसकी सहायता से आप अपने दस्तावेज़ के भीतर सीधे आरेख (diagrams) बना सकते हैं। यह विशेष रूप से निम्नलिखित के लिए उपयोगी है:

- फ्लोचार्ट
- ब्लॉक आरेख
- परिपथ आरेख
- गणितीय चित्रण

यद्यपि इसे सीखने में कुछ समय लगता है, फिर भी यह अत्यंत पेशेवर और विस्तार योग्य (scalable) ग्राफ़िक्स प्रदान करता है।

## 10.5 Writing Journal and Conference Papers in LaTeX / LaTeX में जर्नल एवं कॉन्फ़ेंस शोध-पत्र लेखन

One of the most important uses of LaTeX in academic life is writing journal articles and conference papers. Most reputed journals and conferences provide official LaTeX templates that ensure your manuscript follows their formatting guidelines exactly. Using the correct template saves time and reduces the chances of rejection due to formatting issues.

शैक्षणिक जीवन में LaTeX का एक अत्यंत महत्वपूर्ण उपयोग जर्नल लेखों तथा कॉन्फ़ेंस शोध-पत्रों के लेखन में होता है। अधिकांश प्रतिष्ठित जर्नल और कॉन्फ़ेंस आधिकारिक LaTeX टेम्पलेट प्रदान करते हैं, जो यह सुनिश्चित करते हैं कि आपका पांडुलिपि (manuscript) उनके निर्धारित स्वरूपण दिशानिर्देशों का पूर्णतः पालन करे।

सही टेम्पलेट का उपयोग करने से समय की बचत होती है तथा स्वरूपण संबंधी त्रुटियों के कारण अस्वीकृति (rejection) की संभावना कम हो जाती है।

### 10.5.1 Selecting the Target Journal or Conference / लक्षित जर्नल या कॉन्फ़ेंस का चयन

Before you begin writing, decide where you plan to submit your paper. Different publishers and conferences have different formatting requirements. Always download and use the official template provided on the journal or conference website.

लेखन प्रारंभ करने से पहले यह निर्धारित करें कि आप अपना शोध-पत्र कहाँ प्रस्तुत करने की योजना बना रहे हैं। विभिन्न प्रकाशकों और कॉन्फ़ेंसों के स्वरूपण संबंधी नियम अलग-अलग होते हैं।

हमेशा संबंधित जर्नल या कॉन्फ़ेंस की आधिकारिक वेबसाइट से उपलब्ध आधिकारिक टेम्पलेट डाउनलोड करें और उसी का उपयोग करें।

### 10.5.2 Downloading the Official LaTeX Template / आधिकारिक LaTeX टेम्पलेट डाउनलोड करना

Below are some commonly used publishers and how to obtain their templates:

नीचे कुछ सामान्यतः उपयोग किए जाने वाले प्रकाशक तथा उनके टेम्पलेट प्राप्त करने की विधि दी गई है:

- **IEEE (Institute of Electrical and Electronics Engineers)**

For engineering, electronics, computer science, and related fields, IEEE templates are widely used.

You can download official templates from:

<https://template-selector.ieee.org/secure/templateSelector/publicationType>

These include:

- Journal templates
- Conference paper templates
- Two-column formatting styles

IEEE templates are also directly available on Overleaf by searching for “IEEE Conference Template” or “IEEE Journal Template”.

- IEEE (Institute of Electrical and Electronics Engineers)  
अभियांत्रिकी, इलेक्ट्रॉनिक्स, कंप्यूटर विज्ञान तथा संबंधित क्षेत्रों में IEEE के टेम्पलेट व्यापक रूप से उपयोग किए जाते हैं।  
आप आधिकारिक टेम्पलेट निम्न लिंक से डाउनलोड कर सकते हैं:  
<https://template-selector.ieee.org/secure/templateSelector/publicationType>  
इनमें सामान्यतः निम्न प्रकार शामिल होते हैं:
  - जर्नल टेम्पलेट
  - कॉन्फ्रेंस शोध-पत्र टेम्पलेट
  - दो-स्तंभीय (Two-column) स्वरूपण शैली
 IEEE के टेम्पलेट Overleaf पर भी सीधे उपलब्ध हैं। इसके लिए "IEEE Conference Template" या "IEEE Journal Template" खोजें।
- **Springer**  
Springer journals and conferences provide official LaTeX templates.  
You can download official templates from:  
<https://www.springernature.com/gp/authors/campaigns/latex-author-support>
- Springer  
Springer के जर्नल और कॉन्फ्रेंस आधिकारिक LaTeX टेम्पलेट प्रदान करते हैं।  
आप आधिकारिक टेम्पलेट निम्न लिंक से डाउनलोड कर सकते हैं:  
<https://www.springernature.com/gp/authors/campaigns/latex-author-support>
- **Elsevier**  
Elsevier provides official LaTeX templates.  
You can download official templates from:  
<https://www.elsevier.com/authors/policies-and-guidelines/latex-instructions>  
You can download the package and use the `elsarticle.cls` file in your project.
- Elsevier  
Elsevier आधिकारिक LaTeX टेम्पलेट प्रदान करता है।  
आप आधिकारिक टेम्पलेट निम्न लिंक से डाउनलोड कर सकते हैं:  
<https://www.elsevier.com/authors/policies-and-guidelines/latex-instructions>  
आप पैकेज डाउनलोड करके अपने प्रोजेक्ट में `elsarticle.cls` फ़ाइल का उपयोग कर सकते हैं।
- **ACM (Association for Computing Machinery)**  
For computer science conferences and journals, ACM provides the "acmart" template.  
You can download official templates from:  
<https://www.acm.org/publications/authors/submissions>  
On Overleaf, search for "ACM Conference Template" or "acmart".
- ACM (Association for Computing Machinery)  
कंप्यूटर विज्ञान की कॉन्फ्रेंसों और जर्नलों के लिए ACM "acmart" टेम्पलेट प्रदान करता है।  
आप आधिकारिक टेम्पलेट निम्न लिंक से डाउनलोड कर सकते हैं:  
<https://www.acm.org/publications/authors/submissions>  
Overleaf पर "ACM Conference Template" या "acmart" खोजकर भी आप इसे सीधे उपयोग कर सकते हैं।

### 10.5.3 How to Use the Template / टेम्पलेट का उपयोग कैसे करें

After downloading the template:

1. Unzip the template folder.
2. Open the main .tex file.
3. Read the sample content carefully.
4. Replace:
  - Title
  - Author names and affiliations
  - Abstract
  - Keywords
  - Sections (Introduction, Methodology, Results, Conclusion)
5. Add your bibliography file (.bib).
6. Compile and check formatting compliance.

टेम्पलेट डाउनलोड करने के बाद निम्नलिखित चरणों का पालन करें:

1. टेम्पलेट फ़ोल्डर को अनज़िप करें।
2. मुख्य .tex फ़ाइल खोलें।
3. नमूना सामग्री (sample content) को ध्यानपूर्वक पढ़ें।
4. निम्नलिखित भागों को अपनी सामग्री से प्रतिस्थापित करें:
  - शीर्षक (Title)
  - लेखकों के नाम एवं संबद्धता (Affiliations)
  - सारांश (Abstract)
  - मुख्य शब्द (Keywords)
  - विभाग जैसे — परिचय (Introduction), कार्यविधि (Methodology), परिणाम (Results), निष्कर्ष (Conclusion)
5. अपनी संदर्भ सूची फ़ाइल (.bib) जोड़ें।
6. दस्तावेज़ को संकलित करें और स्वरूपण की शुद्धता की जाँच करें।

If using Overleaf:

- Click “New Project”
- Select “Upload Project”
- Upload the zipped template folder
- Start editing online

यदि आप Overleaf का उपयोग कर रहे हैं:

- “New Project” पर क्लिक करें।
- “Upload Project” का चयन करें।
- ज़िप की गई टेम्पलेट फ़ोल्डर अपलोड करें।
- ऑनलाइन संपादन प्रारंभ करें।

### 10.5.4 Basic Structure of a Journal or Conference Paper / जर्नल या कॉन्फ्रेंस शोध-पत्र की मूल संरचना

Most technical papers follow this general structure:

- Title
- Authors and Affiliations
- Abstract
- Keywords
- Introduction
- Literature Review (optional in short conference papers)
- Methodology / Proposed Work
- Results and Discussion
- Conclusion and Future Work
- Acknowledgment (if applicable)
- References

अधिकांश तकनीकी शोध-पत्र सामान्यतः निम्नलिखित संरचना का पालन करते हैं:

- शीर्षक (Title)
- लेखक एवं संबद्धताएँ (Authors and Affiliations)
- सारांश (Abstract)
- मुख्य शब्द (Keywords)
- परिचय (Introduction)
- साहित्य समीक्षा (Literature Review) — लघु कॉन्फ्रेंस शोध-पत्रों में वैकल्पिक
- कार्यविधि / प्रस्तावित कार्य (Methodology / Proposed Work)
- परिणाम एवं चर्चा (Results and Discussion)
- निष्कर्ष एवं भविष्य का कार्य (Conclusion and Future Work)
- आभार (Acknowledgment) — यदि लागू हो
- संदर्भ (References)

### 10.5.5 Important Practical Advice / महत्वपूर्ण व्यावहारिक सुझाव

- Always read the “Author Guidelines” of the target journal.
- Do not modify margins, font size, or spacing unless permitted.
- Use BibTeX for citation management.
- Check plagiarism before submission.
- Validate figures for resolution (usually 300 dpi or higher).

Using official templates not only ensures professional formatting but also reflects academic discipline and attention to detail. For researchers and faculty members, mastering journal and conference templates in LaTeX is an essential academic skill.

- लक्षित जर्नल के "Author Guidelines" को सदैव ध्यानपूर्वक पढ़ें।
- जब तक अनुमति न हो, मार्जिन, फ्रॉन्ट आकार या स्पेसिंग में परिवर्तन न करें।
- संदर्भ प्रबंधन के लिए BibTeX का उपयोग करें।
- प्रस्तुत करने से पहले प्लेगरिज़्म (Plagiarism) की जाँच अवश्य करें।
- चित्रों (Figures) के रिज़ॉल्यूशन की पुष्टि करें (सामान्यतः 300 dpi या अधिक)।

आधिकारिक टेम्पलेट का उपयोग न केवल पेशेवर स्वरूपण सुनिश्चित करता है, बल्कि यह शैक्षणिक अनुशासन और सूक्ष्म विवरणों के प्रति सजगता को भी दर्शाता है। शोधकर्ताओं और संकाय सदस्यों के लिए LaTeX में जर्नल एवं कॉन्फ्रेंस टेम्पलेट्स में दक्षता प्राप्त करना एक आवश्यक शैक्षणिक कौशल है।

The book **ACADEMIC WRITING WITH LaTeX** will help researchers, mathematicians, students—especially those from a technical background—and academicians present their technical ideas in a clear and professional format. LaTeX is an open-source software tool that is widely used for writing journals, theses, books, reports, and articles. It is free to use and supported by a large community, which makes it highly reliable and powerful.

This book teaches you how to prepare academic documents using practical examples of code, images, and tables. It follows a hands-on approach so that readers can learn by doing rather than just reading theory. The primary language of the book is English, but Hindi support is also provided so that Indian students whose mother tongue is Hindi can understand the concepts more easily. For a better understanding of the concepts, tutorial videos are also available at [www.lecturestand.com](http://www.lecturestand.com)

Dr. Kunwer Mrityunjay Singh is an academician and researcher with a Ph.D. and M.Tech in Computer Science and Engineering from IIT Guwahati, and a B.Tech in Electronics and Communication Engineering from the University of Allahabad. With over eight years of teaching experience in the Department of Technical Education, Government of Uttar Pradesh, he currently serves as a Lecturer in Electronics Engineering (Gazetted). He has led the development of curricula aligned with AICTE guidelines and NEP 2020, now implemented across the diploma sector in Uttar Pradesh. He secured second rank state-wide in the Uttar Pradesh Public Service Commission (UPPSC) Lecturer examination and qualified GATE for three consecutive years (2011–2013).

