

Sheth T. J. Education Society'
Sheth N.K.T.T College of Commerce
and Sheth J.T.T College of Arts, (Autonomous).Thane (W)

T.Y.B.Sc. (Computer Application) 2026-27

	Semester V	Credits		Semester VI	Credits
Major (Mandatory)		2	Major (Mandatory)		
BCA501	Advanced Web Programming	2	BCS601	Security in Networking	2
BCAP502	Advanced Web Programming Practical	2	BCSP602	Security in Networking Practical	2
BCS503	Software Project Management	2	BCM603	Mobile Application Development	2
BCDP504	Data Visualization using Tableau and Software Project Management Practical	2	BCMP604	Mobile Application Development Practical	2
BCI505	Indian Knowledge Systems in Information Technology	2	BCI605	DevOps	2
Major (Elective)			Major (Elective)		
BCJ506	Java Programming	2	BCL606	Machine Learning	2
BCJP507	Java Programming Practical	2	BCLP607	Machine Learning Practical	2
BCE508	Exploratory Data Analysis	2	BCS608	Search Engine Optimization (SEO)	2
BCEP509	Exploratory Data Analysis Practical	2	BCSP609	Search Engine Optimization (SEO) Practical	2
Minor			Minor		
BCE510	Embedded & IoT Systems	2	BCC610	Cloud Computing	2
BCEP511	Embedded & IoT Systems Practical	2	BCCP611	Cloud Computing Practical	2
VSC			SEC	-----	
BCM512	MERN	2			
FP			OJT		
BCFP513	Project Dissertation	2	BGP612	Project Implementation	4
	Total	22			22

Dr. Yogeshwari Patil
Department Coordinator

Prof. Heena Chande
NEP Coordinator

Dr. Dilip Patil
Principal

Sheth T. J. Education Society's
Sheth N.K.T.T College of Commerce and
Sheth J.T.T College of Arts, (Autonomous). Thane (W)

Proramme Name: B.Sc. (Computer Application)		Semester: V
Course Category: Major Mandatory		
Name of the Dept: Science and Technology		
Course Title: Advanced Web Programming		
Course Code: BCA501		Course Level: 5.5
Type: Theory		
Course Credit: 02		
Hours Allotted: 30 Hours		
Marks Allotted: 50 Marks		
Course Objectives: 1. To provide a strong foundation in .NET framework concepts, C# programming, and ASP.NET Web Forms for developing dynamic web applications. 2. To develop practical skills in building data-driven web applications using ADO.NET, server controls, state management, validation, and exception handling techniques.		
Course Outcomes: After completion of the course learners will be able to CO1. develop interactive web applications using C#, ASP.NET Web Forms, server controls, validation techniques, and state management methods. CO2. design and implement database-driven web applications using ADO.NET, SQL queries, data providers, and data-bound controls such as Grid View, Details View, and Form View.		
Description of the course:		This course introduces the basics of the .NET framework and C# programming for building web applications. Students learn about the Common Language Runtime (CLR), .NET class library, and core C# concepts such as variables, data types, objects, namespaces, and assemblies. The course focuses on developing web applications using ASP.NET Web Forms. It covers server controls, event handling, validation controls, state management (View State, cookies, sessions), master pages, user controls, and exception handling. Students also learn how to connect applications to databases using ADO.NET, work with SQL queries, and use data controls like Grid View, Details View, and Form View to create data- driven web applications. Overall, the course helps students develop dynamic and interactive web applications using ASP.NET and C#.

Syllabus: NEP 2020 w.e.f 2026-27

Unit No.	Content	Hours
I	.NET Technology and Framework: C#, VB, and the .NET Languages, The Common Language Runtime, The .NET Class Library The C# Language: C# Language Basics, Variables and Data Types, Objects, and Namespaces and Assemblies. ASP.NET Web Forms, Writing Code, Using the Code-Behind Class, Adding Event Handlers, Understanding the Anatomy of an ASP.NET Application, Introducing Server Controls, Using the Page Class, Using Application Events, Configuring an ASP.NET Application. Form Controls: Stepping Up to Web Controls, Web Control Classes, List Controls, Table Controls, Web Control Events and AutoPostBack, Validation, Understanding Validation, Using the Validation Controls, Rich Controls, The Calendar, The AdRotator, Pages with Multiple Views, User Controls	15
II	Exception Handling, User defined Exceptions, Page Tracing, Client and Server side state management, View State, Transferring Information Between Pages, Cookies- creation, retrieval, reading, Types of Cookies, Using Master Pages and Content Pages ADO.NET Components, Data Providers, SQL Basic command and queries, Understanding the Data Provider Model, Using Direct Data Access, Using Disconnected Data Access. Working with Data Source Controls, Data Controls: GridView, DetailsView and FormView.	15
	Total Hours	30

References:

Sr.No.	Title	Author/s Publisher	Publisher	Edition	Year
1	Beginning Visual C# 2010	K. Watson, C. Nagel, J.H Padderson	Wrox (Wiley)		2010
2	Murach's ASP.NET 4.6 Web Programming in C#2015	Mary Delamater and Anne Bohem	SPD	Sixth	2016
3	ASP.NET 4.0 programming	J. Kanjilal	Tata McGraw Hill		2011

Sheth T. J. Education Society's
Sheth N.K.T.T College of Commerce and Sheth
J.T.T College of Arts, (Autonomous). Thane (W)

Programme Name: B.Sc. (Computer Application)		Semester: V
Course Category: Major Mandatory		
Name of the Dept: Science and Technology		
Course Title: Advanced Web Programming Practical		
Course Code: BCAP502		Course Level: 5.5
Course Type: Practical		
Course Credit: 02		
Total Hours: 60 Hours		
Total Marks: 50 Marks		
Course Objectives: 1. To develop programming skills in C# for building console and web-based applications using object-oriented concepts such as delegates, interfaces, and basic algorithms 2. To enable students to design and develop dynamic ASP.NET web applications using server controls.		
Course Outcomes: After completion of the course learners will be able to CO1: design and implement C# applications demonstrating object-oriented programming concepts, delegates, interfaces, and logical problem-solving techniques. CO2: develop and deploy ASP.NET web applications using server controls, state management, validation techniques, data access operations, and data-bound controls such as GridView, Details View, and Form View.		
Course Description:		This course provides hands-on experience in developing applications using C# and ASP.NET. It covers fundamental programming concepts such as arithmetic operations, Fibonacci series, and prime number testing, along with object-oriented features like delegates and interfaces. The course also focuses on building dynamic web applications using ASP.NET server controls, validation controls, state management techniques, cookies, master pages, user controls, navigation controls, and data-bound controls such as GridView and DetailsView. Students will learn to design interactive web forms, manage application state, and perform database operations including inserting, deleting, sorting, and paging of records. Overall, the course aims to enhance practical skills in developing dynamic, data-driven web applications using Microsoft technologies.

Syllabus: NEP 2020 w.e.f 2026-27

Sr No.	Content (List of Practicals)	Hours
1	Write the program for the following: - Create an application to print on screen the output of adding, subtracting, multiplying and dividing two numbers entered by the user in C#. - Create an application to demonstrate following operations i. Generate Fibonacci series. ii. Test for prime numbers.	
2	Write the program for the following: - Create a simple application to perform addition and subtraction using delegate. - Create a simple application to demonstrate use of the concepts of interfaces.	
3	Write the program for the following: Create a simple web page with various server controls to demonstrate setting and use of their properties. (Example : AutoPostBack)	
4	Write the program for the following: - Create a simple application to demonstrate your vacation using calendar control. - Demonstrate the use of Treeview operations on the web form.	
5	Write the program for the following: Create a Registration form to demonstrate use of various Validation controls	
6	Write the program for the following: - Create Web Form to demonstrate use of Adrotator Control. - Create Web Form to demonstrate use User Controls	
7	Write the program for the following: - Create Web Form to demonstrate use of Website Navigation controls. - Create a web application to demonstrate use of Master Page and content page.	
8	Write the program for the following: - Create a web application to demonstrate various states of ASP.NET Pages. - Create a web application to demonstrate the use of different types of Cookies	
9	Write the program for the following: Create a web application for inserting and deleting records from a database.	
10	Write the program for the following: - Create a web application to demonstrate use of GridView button column and GridView events along with paging and sorting. - Create a web application to demonstrate data binding using DetailsView and FormView Control.	
	Total Hours	60

Sheth T. J. Education Society's
Sheth N.K.T.T College of Commerce and
Sheth J.T.T College of Arts, (Autonomous) Thane (W)

Programme Name: B.Sc. (Computer Application)		Semester: V
Course Category: Major Mandatory		
Name of the Dept: Science & Technology		
Course Title: Software Project Management		
Course Code: BCS503		Course Level: 5.5
Type: Theory		
Course Credit: 02		Total Marks: 50
Course Objectives: 1. To provide fundamental knowledge of software project management concepts including project characteristics, management principles, project evaluation, risk basics, project planning, and software development life cycle models. 2. To develop understanding of software development strategies, estimation techniques, risk and resource management, contract management, team organization, quality management, and project closure practices.		
Course Outcomes: After completion of the course learners will be able to CO1: explain software project management fundamentals, analyze project feasibility using cost-benefit and risk evaluation techniques, and apply appropriate SDLC models and planning approaches for software projects. CO2: estimate software effort, manage risks and resources, understand contract and team management practices, apply quality assurance and testing methods, and prepare project closure documentation.		
Description of the course:		This course provides an introduction to the principles and practices of Software Project Management. It covers project planning, evaluation, risk management, resource allocation, and software development life cycle models. Students learn various development approaches such as Agile, prototyping, and incremental models along with effort estimation techniques. The course also includes contract management, team organization, quality management, software testing, and project closure processes. It equips students with the knowledge required to manage software projects effectively from initiation to completion.

Syllabus: NEP 2020 w.e.f 2026-27

Unit No.	Content	Hours
I	Fundamentals of Software Project Management Introduction to Project, Characteristics of a project, what is Management? Features of Management, Software Project Management, Software Project vs Other Project, Project management life cycles, Categories of Software Projects, Project Charter, Stakeholders, Business Case, Traditional V/S Modern Project Management Practices. Project Evaluation, Risk and Program Management Project Portfolio Management, Cost Benefit, Cost Evaluation Techniques, Software Risk, Risk Evaluation, Risk Management, Tools for Risk Management, Program Management Framework, Stages in Program Management Project Planning and Software Development Approaches Business need, goal, objective, Work Breakdown, Software, Project and its types, Project Methodology, Software Process Model, Software Development Lifecycle, SDLC Models	15
II	Software Development Strategies and Effort Estimation Software Prototyping, Incremental Delivery, DSDM, RAD, Agile, Software Effort Estimation, Software Estimation Techniques, Risk Management and Resource Planning Risk and its categories, Risk Management, Resources and its types, Creating a Framework, Review Process, Software Configuration Management Contract, Team, and Quality Management Contracts and its types, typical terms of contracts, Recruitment Process, Stress Management, Maslow's Hierarchy of Needs, Organization and Team Structures, Software Quality and model, Software Quality Management, Software Testing, Project closure, Project Closure Report	15
	Total Hours	30

References:

Sr. No	Title	Author	Publisher	Edition	Year
1.	Software Project Management	Bob Hughes, Mike Cotterell, Rajib Mall	TMH	6th	2018
2.	Project Management and Tools & Technologies – An overview	Shailesh Mehta	SPD	1st	2017
3.	Software Project Management	Walker Royce	Pearson		2005

**Sheth T. J. Education Society's
Sheth N.K.T.T College of Commerce and
Sheth J.T.T College of Arts, (Autonomous).Thane (W)**

Programme Name: B.Sc. (Computer Application)		Semester: V
Course Category/Vertical: Major Mandatory		
Name of the Dept: Science and Technology		
Course Title: Data Visualization using Tableau a n d Software Project Management Practical		
Course Code: BCDP504		Course Level:5.5
Type: Practical		
Course Credit: 2 credits		
Hours Allotted: 60 Hours		
Marks Allotted: 50 Marks		
Course Objectives: 1. To familiarize students with the Tableau software and its key features. 2. To develop students' understanding of data preparation techniques and best practices in Tableau. 3. To enable students to create a variety of visualizations using different chart types and design principles in Tableau. 4. To introduce students to Agile project management concepts and tools. 5. To provide hands-on experience in creating and managing projects using Jira Cloud. 6. To enhance students' ability to manage tasks, issues, and workflows in a collaborative environment.		
Course Outcomes: After completion of the course learners will be able to CO1. install Tableau Desktop and navigate the Tableau workspace proficiently and demonstrate competence in data preparation, including data connection, joining, data types, and calculated fields in Tableau. CO2. design and create visually compelling and informative visualizations using various chart types and visualization techniques in Tableau and utilize advanced analytics features in Tableau for trend analysis, forecasting, and cluster analysis. CO3. effectively share their visualizations and insights using Tableau Online, Tableau Server, and Tableau Public, and demonstrate an understanding of best practices in data sharing and presentation. CO4. create and manage projects using Agile methodologies in Jira. CO5. handle issues, assign tasks, and track project progress effectively. CO6. generate basic reports and use boards (Scrum/Kanban) for project monitoring.		
Description of the course:	This course introduces students to Tableau, a powerful data visualization and analytics tool. From installing Tableau Desktop to creating interactive dashboards and sharing insights, students will learn data preparation, visualization techniques, advanced analytics, and how to leverage Tableau Online. Students will learn to perform basic database operations, handle data interchange, and manage projects using Scrum and Kanban methodologies. The course emphasizes hands-on learning to build technical and project management skills aligned with industry requirements.	

Syllabus : NEP 2020 w.e.f 2026-27

Sr. No.	Content (List of Practicals)	Hours
	Unit I	
1.	Introduction to Tableau - Install, prepare data, navigate workspace, create visualizations, save/share workbooks.	
2.	Adding Data Sources - Set up connectors, select tables, perform joins/unions, edit metadata, add hierarchies/calculated fields, optimize performance.	
3.	Creating Data Visualizations - Explore chart types, design bar/line/highlight/heatmap/bullet charts, understand visualization anatomy.	
4.	Aggregate Functions and Calculated Fields - Use aggregates/calculated fields, handle text/date fields, apply logical functions/parameters, search text fields.	
5.	Table Calculations and Level of Detail Calculations - Perform different calculations, apply quick/customized table calculations, implement level of detail expressions.	
6.	Maps in Tableau - Create symbol/filled/density maps, add layers/pie charts, use viz in tooltip, explore alternative map services, analyze spatial data.	
7.	Advanced Analytics in Tableau - Identify trends/forecasts/clusters, utilize analytics pane, incorporate lines/forecasts, perform cluster analysis.	
8.	Interactive Dashboards - Considerations for dashboard creation, create/place charts, add titles/navigation/buttons/actions, follow best practices.	
9.	Sharing Insights with Tableau - Utilize Tableau Online/Server, publish to Tableau Public, embed visualizations in websites.	
10.	Data Preparation with Tableau Prep - Connect to data, perform wildcard unions, inspect/clean/format data, remove unneeded fields, combine data with unions/joins, run/save flows.	
	Unit II	
1	Create a Jira Cloud account and set up a new project.	
2	Create and configure Scrum or Kanban project in Jira.	
3	Add team members and assign roles in Jira.	
4	Create and configure issue types such as Epic, Story, Task, and Bug.	
5	Create and manage issues under an Epic in Jira.	
6	Assign issues with priority and due dates.	
7	Design and customize workflows with transitions in Jira.	
8	Create and manage Scrum board and perform sprint planning.	
9	Configure Kanban board and track issue progress.	
10	Generate reports and dashboards for project tracking in Jira.	
	Total Hours	60

Sheth T. J. Education Society's
Sheth N.K.T.T College of Commerce and
Sheth J.T.T College of Arts, Thane (W) Autonomous

Syllabus: NEP 2020 w.e.f 2026-27

Programme Name: B.Sc. (Computer Application)		Semester: V
Course Category: Major Mandatory IKS		
Name of the Dept: Science & Technology		
Course Title: Indian Knowledge Systems in Information Technology		
Course Code: BCI505		Course Level:5.5
Type: Theory		
Course Credit: 2		
Hours Allotted: 30 Hours		
Total Marks: 50 Marks		
Course Objectives: 1.To analyze key knowledge-generation methods in the Indian Knowledge System and examine contributions from Nyāya, Vaiśeṣika, Vedic mathematics, and ancient computational traditions in modern data science. 2. To integrate IKS perspectives into data ethics and responsible AI, and apply them to develop culturally rooted, multidisciplinary data science solutions.		
Course Outcomes: After completion of the course learners are able to CO1: apply classical Indian logical, mathematical, and algorithmic traditions to analyze and solve problems in data-intensive environments. CO2: compare and contrast different epistemological frameworks to evaluate methods for generating reliable knowledge in data-intensive environments.		
Description of the course:	This course explores Indian Knowledge Systems through philosophy, mathematics, logic, linguistics, and science, highlighting their relevance to information technology. It examines epistemology, algorithms, number systems, logic, Sanskrit grammar, artificial intelligence, mechanical computation, and modern digital preservation. The syllabus connects ancient Indian thought with contemporary computing, interdisciplinary research, ethical technology, and future innovations across education, governance, culture, and emerging global technologies	

Unit No.	Content	Hours
I	Unit 1: Foundations of Indian Knowledge Systems and Mathematical Logic for Computing Introduction to Indian Knowledge Systems (IKS), Philosophical Foundations: Pramana and Schools of Thought (Nyaya, Vedanta, Mimamsa), IKS and Information Technology: Knowledge Representation and Computational Thinking, Decimal Number System and Place Value Concept, Concept of Zero and Its Computational Significance, Binary Systems and Combinatorics in Pingala's Chanda's Shastra, Aryabhata's Computational Methods, Algorithmic Thinking in Ancient Indian Mathematics	15
II	Unit 2: Indian Logic, Linguistics, and Applications in Modern Computing Fundamentals of Nyaya Logic and Indian Syllogism, Inference Systems and Reasoning Models, Comparison of Indian and Western Logic, Kautilya's Arthashastra and Decision-Making Models, Applications of Indian Logic in AI and Computing Systems, Panini's Ashtadhyayi and Rule-Based Systems, Sanskrit Grammar and Formal Language Theory, Generative Grammar and Programming Concepts, Compiler Design Concepts and Sanskrit Grammar, Natural Language Processing and Computational Linguistics	15
	Total Hours	30

Reference Books

- Mahadevan, B., Bhat, Vinayak Rajat, and Nagendra Pavana R. Introduction to Indian Knowledge System: Concepts and Applications. PHI Learning, 2022.
 - Kapil Kapoor and Avadesh Kumar Singh. Indian Knowledge Systems. D.K. Printworld.
 - Kak, Subhash and Rao, T. R. N. Computing Science in Ancient India. Center for Advanced Computer Studies, University of Southwestern Louisiana, 1998.
 - Bhanu Murthy, T. S. A Modern Introduction to Ancient Indian Mathematics. New Age International, 1992.
 - Kim Plofker. Mathematics in India. Princeton University Press, 2009.
 - C. K. Raju. Cultural Foundations of Mathematics. Pearson Education.
- Briggs, Rick. Knowledge Representation in Sanskrit and Artificial Intelligence. AI Magazine, 1985.

**Sheth T. J. Education Society's
Sheth N.K.T.T College of Commerce and
Sheth J.T.T College of Arts, (Autonomous). Thane (W)**

Programme Name: B.Sc. (Computer Application)		Semester: V
Course Category: Major Elective		
Name of the Dept: Science & Technology		
Course Title: Java Programming		
Course Code: BCJ506		Course Level: 5.5
Type: Theory		
Course Credit: 2		
Hours Allotted: 30 Hours		
Total Marks: 50 Marks		
Course Objectives: • To understand the fundamentals of Java programming, including classes, objects, inheritance, interfaces, and program structure for developing object-oriented applications. • To develop Java applications using advanced concepts such as exception handling, multithreading, event handling, and GUI components using AWT.		
Course Outcomes: After the completion of course learners will be able to: CO1: apply object-oriented programming concepts in Java such as classes, inheritance, interfaces, and access control to develop programs. CO2: develop Java applications using exception handling, multithreading, event handling, and AWT-based graphical user interfaces.		
Description of the course:		This course introduces the fundamentals of Java programming and object-oriented concepts such as classes, inheritance, and interfaces. It also covers advanced topics including exception handling, multithreading, event handling, and GUI development using AWT.

Unit No.	Content	Hours
I	Introduction: History, Features of Java, Java Development Kit, Java Application Programming Interface, Java Virtual Machine, Java Program Structure. Classes: The Class Object and Its Attributes, Class Methods, Accessing A Method, Method Overloading, Instantiating Objects from A Class, Constructors, this keyword, super keyword, Types of Classes, Scope Rules, Access Modifier, constants, static members of a class, garbage collection. Inheritance: Derived Class Objects, Inheritance and Access Control, Default Base Class Constructors, this and super keywords. Abstract Classes and Interfaces, Abstract Classes, Abstract Methods, Interfaces: What Is an Interface? How Is an Interface Different from An Abstract Class? Multiple Inheritance, Defining an Interface, Implementing Interfaces.	15
II	Exceptions: Catching Java Exceptions, Catching Run-Time Exceptions, Handling Multiple Exceptions, The finally Clause, The throws Clause, Built in Exceptions in java Multithreading: Thread Creations, Thread Life Cycle, Life Cycle Methods, Synchronization, wait() notify() notify all() methods Event Handling: Delegation Event Model, Events, Event classes, Eventlistener interfaces, Using delegation event model, adapter classes and inner classes. Abstract Window Toolkit: Window Fundamentals, Component, Container, Panel, Window, Frame, Canvas. Components – Labels, Buttons, Check Boxes, Radio Buttons, Choice Menus, Text Fields, Text, Scrolling List, Scrollbars, Panels, Frames Layouts: Flow Layout, Grid Layout, Border Layout, Card Layout.	15
	Total Hours	30

References:

Sr. No	Title	Author	Publisher	Edition	Year
1.	Core Java 8 for Beginners	Vaishali Shah, Sharnam Shah	SPD	Ist	2015
2.	Java: The Complete Reference	Herbert Schildt	McGraw Hill	9 th	2014
3.	Murach's beginning Java with Net Beans	Joel Murach , Michael Urban	SPD	Ist	2016
4.	Core Java, Volume I: Fundamentals	Hortsman	Pearson	9 th	2013
5.	Programming with Java 3e A pirmer	E. Balguruswami	McGraw Hill	3 rd	2007

**Sheth T. J. Education Society's
Sheth N.K.T.T College of Commerce and
Sheth J.T.T College of Arts, (Autonomous). Thane (W)**

Programme Name: B.Sc. (Computer Application)		Semester: V
Course Category: Major Elective		
Name of the Dept: Science & Technology		
Course Title: Java Programming Practical		
Course Code: BCJP507		Course Level: 5.5
Type: Practical		
Course Credit: 02		
Hours Allotted: 60 Hours		
Total Marks: 50 Marks		
Course Objectives: - To implement object-oriented programming concepts in Java such as constructors, method overloading, inheritance, abstract classes, and interfaces. - To develop Java applications using advanced features including exception handling, multithreading, Swing GUI components, layouts, and event handling.		
Course Outcomes: After completion of the course learners will be able to CO1: apply object-oriented programming concepts in Java to design and implement programs. CO2: develop Java applications using exception handling, multithreading, Swing components, layouts, and event handling techniques.		
Description of the course:		This practical course focuses on implementing object-oriented programming concepts in Java and developing applications using exception handling, multithreading, Swing GUI components, layouts, and event handling.

Syllabus: NEP 2020 w.e.f 2026-27

Sr. No.	Content (List of Practical)	Hours
1.	OOPs concepts in Java-1	
a.	Write a program to create a class and implement a default, overloaded and copy Constructor.	
b.	Write a program to create a class and implement the concepts of Method Overloading.	
2	OOPs concepts in Java-2	
a.	Write a program to implement the concepts of Inheritance and Method Overriding	
b.	Write a program to implement the concepts of Abstract classes and methods	
c.	Write a program to implement the concept of interfaces	
3.	Exceptions:	
a.	Write a program to raise built-in exceptions and raise them as per the Requirements	
b.	Write a program to define user defined exceptions and raise them as per the Requirements	
4.	Multithreading:	
a.	Write a Java program to create and start multiple threads that increment a shared counter variable concurrently	
b.	Write a java application to guess the Jackpot (a random no) game using Multithreading.	
5.	Swing:	
a.	Create a swing application that randomly changes color on button click.	
b.	Create a Swing application to demonstrate use of Text Area using scrollpane to show content of text file in textarea selected using file chooser.	
c.	Create a Swing application to demonstrate use of scrollpane to change its color selected using colour chooser	
6.	Layouts: Write programs for the following layouts:	
a.	Flow Layout	
b.	Grid Layout	
c.	Border Layout	
7.	Events: Write programs to demonstrate the following events:	
a.	ActionEvent	

b.	MouseEvent	
c.	KeyEvent	
d.	SelectionEvent	
8.	Demonstrate the use of Adapter Class in Event Handling	
9.	Demonstrate the use of Anonymous Inner Class in Event Handling	
	Total Hours	60

**Sheth T. J. Education Society's
Sheth N.K.T.T College of Commerce and
Sheth J.T.T College of Arts (Autonomous), Thane (W)**

Programme Name: B.Sc. (Computer Application)		Semester: V
Course Category: Major Elective		
Name of the Dept: Science and Technology		
Course Title: Exploratory Data Analysis		
Course Code: BCE508		Course Level: 5.5
Type: Theory		
Course Credit: 2 credits		
Hours Allotted: 30 Hours		
Marks Allotted: 50 Marks		
Course Objectives: - To introduce the concepts and techniques of Exploratory Data Analysis (EDA) for understanding data characteristics, assumptions, and patterns. - To develop the ability to perform data preparation, univariate, bivariate, and multivariate analysis for discovering relationships, handling missing data, and identifying patterns in datasets.		
Course Outcomes: After completion of the course learners will be able to CO1: apply EDA techniques to summarize, visualize, and interpret datasets using statistical measures and graphical representations. CO2: analyze relationships among variables and perform data preparation and dimensionality reduction techniques to support data-driven decision making.		
Description of the course:		This course introduces the fundamental concepts and techniques of Exploratory Data Analysis (EDA) for understanding and summarizing datasets. It covers data preparation, handling missing values, and identifying outliers, along with univariate, bivariate, and multivariate analysis using statistical measures and visualization techniques. The course also introduces correlation analysis and dimensionality reduction concepts such as PCA to discover patterns and relationships in data.

Unit No.	Content	Hours
I	Introduction to Exploratory Data Analysis (EDA) : Definition of EDA, difference between EDA with classical and Bayesian Analysis, comparison of EDA with Classical data summary measures, goals of EDA, Underlying assumptions in EDA, importance of EDA in data exploration techniques, introduction to different techniques to test the assumptions involved in EDA, role of graphics in data exploration, introduction to unidimensional, bidimensional and multidimensional graphical representation of data. DATA PREPARATION: Introduction to data exploration process for data preparation, data discovery, issues related with data access, characterization of data, consistency and pollution of data, duplicate or redundant variables, outliers and leverage data, noisy data, missing values, imputation of missing and empty places, with different techniques, missing pattern and its importance, handling non numerical data in missing places.	15
II	Univariate Analysis : Measures of Central Tendency (Mean, Median, Mode), Measures of Dispersion (Variance, SD, IQR), Shape of Data: Skewness (Pearson's coefficients) and Kurtosis, Distribution Analysis : Normal distribution, Binomial, and Poisson; Probability Density Functions (PDF) and Cumulative Distribution Functions (CDF). Multivariate Analysis & Visualization : Discovering relationships between variables and dimensionality reduction. Bivariate Analysis: Continuous vs. Continuous: Pearson and Spearman Correlation. Categorical vs. Categorical: Chi-square test and Contingency tables. Categorical vs. Continuous: Z-test, T-test, and ANOVA basics. Multivariate Exploration: Heatmaps for correlation matrices, Pair plots, and Parallel Coordinates. Dimensionality Introduction: The "Curse of Dimensionality," Intuition behind Principal Component Analysis (PCA) as an EDA tool.	15
	Total Hours	30

References:

1. Exploratory Data Analysis, Tukey, J. W. (1977). Exploratory Data Analysis. Reading, MA: Addison-Wesley.
2. Python for Data Analysis, McKinney, W. (2018). Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython (2nd ed.). Sebastopol, CA: O'Reilly Media.
3. Practical Statistics for Data Scientists, Bruce, P., Bruce, A., & Gedeck, P. (2020). Practical Statistics for Data Scientists: 50+ Essential Concepts Using R and Python (2nd ed.). Sebastopol, CA: O'Reilly Media.
4. An Introduction to Statistical Learning, James, G., Witten, D., Hastie, T., & Tibshirani, R. (2021). An Introduction to Statistical Learning with Applications in R (2nd ed.). New York, NY: Springer.

**Sheth T. J. Education Society's
Sheth N.K.T.T College of Commerce and
Sheth J.T.T College of Arts (Autonomous), Thane (W)**

Programme Name: B.Sc. (Computer Application)		Semester: V
Course Category / Vertical: Major Elective		
Name of the Dept: Science and Technology		
Course Title: Exploratory Data Analysis Practical		
Course Code: BCEP509		Course Level: 5.5
Type: Practical		
Course Credit: 02		
Hours Allotted: 60 Hours		
Marks Allotted: 50 Marks		
Course Objectives: • To introduce the concepts and techniques of Exploratory Data Analysis (EDA) for understanding data characteristics, assumptions, and patterns. • To develop the ability to perform data preparation, univariate, bivariate, and multivariate analysis for discovering relationships, handling missing data, and identifying patterns in datasets.		
Course Outcomes: After completion of the course learners will be able to CO1: apply EDA techniques to summarize, visualize, and interpret datasets. CO2: analyze relationships among variables and perform data preparation and dimensionality reduction techniques to support data-driven decision making.		
Description of the course:		This course introduces the fundamental concepts and techniques of Exploratory Data Analysis (EDA) for understanding and summarizing datasets. It covers data preparation, handling missing values, and identifying outliers, along with univariate, bivariate, and multivariate analysis using statistical measures and visualization techniques. The course also introduces correlation analysis and dimensionality reduction concepts such as PCA to discover patterns and relationships in data.

Syllabus: NEP 2020 w.e.f 2026-27

Unit No.	Content(List of Practicals)	Hours
	1.Managing Data Frames with the dplyr package 2 Use dplyr Grammar for inbuilt data set car. 3 Use group by(), %>%,mutate(), rename(),arrange(), filter(), select() 4 Use the data set air quality from inbuilt data sets library. a) Use summary statistics and find the important key values from the output b) Use boxplot and find the interquartile range. Interpret the boxplot and inner and outerfencing of outliers c) Check the missing value in the data set and fine the suitable solution for the missing values. d) Using histogram, find the distribution of data and give proper comment over the dataset. 5 Use bar plot and identify the difference between bar plot and histograms. Conclude the appropriate use of bar plot and histogram. 6 Explore the two dimensional data 7 Scatter plot between two variables 8 Five number summary in exploratory data analysis 9 Multiple histogram and multiple boxplots, Multiple scatter plots and colouring the graph 10 Lattice system in R environment and Graphical window in R and its uses	
	Total Hours	60

References:

1. Exploratory Data Analysis, Tukey, J. W. (1977). Exploratory Data Analysis. Reading, MA: Addison-Wesley.
2. Python for Data Analysis, McKinney, W. (2018). Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython (2nd ed.). Sebastopol, CA: O'Reilly Media.
3. Practical Statistics for Data Scientists, Bruce, P., Bruce, A., & Gedeck, P. (2020). Practical Statistics for Data Scientists: 50+ Essential Concepts Using R and Python (2nd ed.). Sebastopol, CA: O'Reilly Media.

Sheth T. J. Education Society's
Sheth N.K.T.T College of Commerce and
Sheth J.T.T College of Arts (Autonomous), Thane (W)

Proramme Name: B.Sc. (Computer Application)		Semester: V
Course Category: Minor		
Name of the Dept: Science and Technology		
Course Title: Embedded and IOT Systems		
Course Code: BCE510		Course Level: 5.5
Type: Theory		
Course Credit: 02		
Hours Allotted: 30 Hours		
Marks Allotted: 50 Marks		
Course Objectives: 1. To develop fundamental understanding of PIC microcontroller architecture, memory organization, instruction set, programming in Assembly and C, and interfacing techniques including I/O ports, timers, communication protocols, and peripheral devices. 2. To introduce students to Internet of Things (IoT) concepts and Arduino-based system design, enabling them to interface sensors, implement communication protocols (I2C, SPI, Bluetooth, Wi-Fi, RFID, GSM, etc.), and build real-time embedded applications.		
Course Outcomes: After completion of the course learners will be able to CO1. analyze and program PIC microcontrollers using Assembly and C language, implement timer operations, memory allocation, data conversion, and interface communication protocols such as I2C, SPI, Bluetooth, Wi-Fi, and RFID. CO2. design and develop IoT-based embedded applications by interfacing various sensors and implementing wireless communication technologies for real-world problem solving.		
Description of the course:	This course introduces the fundamentals of embedded systems using PIC microcontrollers and Arduino platforms. It covers microcontroller architecture, memory organization, instruction sets, programming in Assembly and C, and interfacing techniques including I/O ports, timers, and data conversion. The course also explores various communication protocols such as I2C, SPI, Bluetooth, Wi-Fi, RFID, GSM, and GPS. Students are introduced to Internet of Things (IoT) concepts and learn to interface sensors and develop real-time embedded applications for practical and industrial use.	

Syllabus: NEP 2020 w.e.f 2026-27

Unit No.	Content	Hours
I	PIC Microcontroller: Architecture – memory organization – addressing modes – instruction set – PIC programming in Assembly & C –I/O port Data Conversion, RAM & ROM Allocation, Timer programming Communication Protocol & Implementation: Introduction to Communication Protocol, I2C, I2C devices – RTC, Memory, ADC- DAC Port Expander, SPI (Serial Peripheral Interface), Bluetooth, Wi- Fi and RFID. Understanding Serial, Communication, Bluetooth Communication, SPI Interface ZigBee, Wi-Fi, I2C, Infrared, RFID, GSM, GPS PDH/SDH/Ethernet The Internet of Things: An Overview : The Flavour of the Internet of Things, The “Internet” of “Things”, The Technology of the Internet of Things, Enchanted Objects.	15
II	Getting Started with Arduino: Introduction, Arduino Variants, Install the Drivers, Arduino IDE Basic Functions: Structure, Digital I/O Functions, Analog I/O Functions, Advanced I/O Functions, Timer Functions, Communication Functions, Interrupt Functions, Math Functions. Using Sensors with the Arduino: Light Sensitive Sensors, Temperature Sensors, Temperature and Humidity Sensor, Line Tracking Sensor, Ultrasonic Sensors, Digital Infrared Motion Sensor, Gas Sensor, Color Sensor.	15
	Total Hours	30

References:

Sr.No.	Title	Author/s Publisher	Publisher	Edition	Year
1	Introduction to embedded systems	Shibu K V	Tata McGraw-Hill	First	2012
2	Embedded Systems	Rajkamal	Tata McGraw-Hill		
3	Programming Embedded Systems in C and C++	Michael Barr	O'Reilly	First	1999
4	Designing the Internet of Things	Adrian McEwen, Hakim Cassi Mally	WILEY	First	2014
5	Internet of Things – Architecture and Design	Raj Kamal	McGraw Hill	First	2017

Sheth T. J. Education Society's
Sheth N.K.T.T College of Commerce and Sheth J.T.T College of Arts,
(Autonomous). Thane (W)

Programme Name: B.Sc (Computer Application)		Semester: V
Course Category: Minor		
Name of the Dept: Science and Technology		
Course Title: Embedded and IOT System Practical		
Course Code: BCEP511		Course Level: 5.5
Course Credit: 02		
Total Hours: 60 Hours		
Total Marks: 50 Marks		
Course Objectives: 1. To provide hands-on experience in programming and interfacing Arduino and Raspberry Pi with LEDs, displays, sensors, and peripheral devices. 2. To enable students to design and implement IoT-based real-time applications using Raspberry Pi by integrating communication modules, camera, GPS, RFID, fingerprint sensor, and web-based control systems.		
Course Outcomes: After completion of the course learners will be able to CO1: implement basic embedded applications such as LED control, display interfacing (7-segment, LCD), and sensor integration using Arduino and Raspberry Pi. CO2: develop and deploy IoT-based projects including home automation, web-controlled systems, GPS tracking, camera interfacing, RFID authentication, and communication-based control applications.		
Description of the course:		This laboratory course provides practical exposure to Arduino and Raspberry Pi platforms for embedded and IoT applications. Students perform experiments involving LED control, display interfacing, sensor integration, GPS, RFID, fingerprint modules, camera interfacing, and web-based automation systems. The course emphasizes real-time implementation, hardware interfacing, and IoT-based application development.

Syllabus: NEP 2020 w.e.f 2026-27

Sr No.	Content (List of the Practicals)	Hours
1	Introduction to Arduino: Blinking of LEDs.	
2	Starting Raspbian OS, Familiarizing with Raspberry Pi Components and interface, Connecting to ethernet, Monitor, USB. a. Displaying different LED patterns with Raspberry Pi.	
3	Displaying Time over 4-Digit 7-Segment Display using Raspberry Pi	
4	Interfacing 16X2 LCD with Raspberry Pi to display different messages.	
5	Raspberry Pi Based Oscilloscope	
6	Controlling Raspberry Pi with WhatsApp.	
7	Fingerprint Sensor interfacing with Raspberry Pi	
8	Raspberry Pi GPS Module Interfacing	
9	IoT based Web Controlled Home Automation using Raspberry Pi	
10	Interfacing Pi Camera with Raspberry Pi	
11	Interfacing Raspberry Pi with RFID.	
	Total Hours	60

Sheth T. J. Education Society's
Sheth N.K.T.T College of Commerce and
Sheth J.T.T College of Arts (Autonomous), Thane (W)

Programme Name: B.Sc. (Computer Application)		Semester: V
Course Category / Vertical: Vocational Skill Course (VSC)		
Name of the Dept: Science and Technology		
Course Title: MERN Practical		
Course Code: BCM512		Course Level: 5.5
Type: Practical		
Course Credit: 2 credits		
Hours Allotted: 60 Hours		
Marks Allotted: 50 Marks		
Course Objectives: 1. To develop backend applications using Node.js, Express, MongoDB, and Socket.IO, including CRUD operations, real-time communication, file handling, and module creation. To build frontend applications using React and Angular, including form handling, validation, API integration, and dynamic data rendering.		
Course Outcomes: After completion of the course learners will be able to CO1: develop server-side applications using Node.js, Express, MongoDB, including CRUD operations, real-time chat, and file uploads. CO2: To build interactive front-end applications using React and Angular with form validation, API integration, and component-based design.		
Description of the course:	This course introduces students to full-stack web development using modern JavaScript technologies. It covers the development of server-side applications using Node.js, Express.js, and MongoDB, including CRUD operations, file uploads, and real-time chat applications using Socket.IO. Students will also learn how to create and publish custom Node modules. On the front-end, the course focuses on building dynamic user interfaces using React and Angular. Students will develop applications such as TODO lists using components and hooks, implement form validation, fetch data from APIs, and perform user data operations through API integration. The course provides practical knowledge for developing modern, scalable web applications.	

Syllabus: NEP 2020 w.e.f 2026-27

Pract.No.	Content (List of the Practicals)	Hours
1	Create Chat application by using Socket.IO	60
2	a. Perform CRUD operations by using Express with MongoDB b. Perform File (image/doc) Upload operation by using MongoDB Creating and Publishing Your Own Node Module	
3	a. Create TODO list by using react hooks with different components.	
4	b. Create and validate the user form in React.	
5	Fetch the API data and render the list on React app.	
6	In Angular create TODO list with different components.	
7	Create and validate the user form in Angular (Reactive Form/Template driven Form).	
8	Fetch the API data and render the list on Angular app.	
9	In Angular create and update the user data by using API	

Scheme of Examination

Course with Credit	External Examination	Internal Examination	Total
Credit 4	60 marks	40 marks	100 marks
Credit 2	30 marks	20 marks	50 marks

Internal Examination Structure(Theory)

Internal examination	40 marks	20 marks
Project Presentation/Case Study /Quiz/Group Discussion	10 marks	5 marks
Assignment /Active class Participation/Attendance	10 marks	5 marks
Class test	20 marks	10 marks
Total	40 marks	20 marks

Structure for Class Test(For 10 Marks)

For 10 Marks	
Q1. Multiple Choice Questions	5 Marks
Q2. Answer in one or two sentences	5 Marks

Structure for Class Test(For 20 Marks)

For 20 Marks	
Q1. Multiple Choice Questions	10 Marks
Q2. Answer in one or two sentences	10 Marks

External Examination (For 60 Marks)

Q. No.	External	Marks: 60
Q.1	Answer the following questions (Any 3) A B C D E F	15 Marks
Q. 2	Answer the following questions (Any 3) A B C D E F	15 Marks
Q. 3	Answer the following questions (Any 3) A B C D E F	15 Marks
Q. 4	Answer the following questions (Any 3) A B C D E F	15 Marks

External Examination (For 30 Marks)

Q. No.	External	Marks: 30
Q.1	Answer the following questions (Any 3) A B C D E F	15 Marks
Q. 2	Answer the following questions (Any 3) A B C D E F	15 Marks

Scheme of Practical Examination (Under NEP)

Under Graduate:

Practical External Exam: 50 marks

A Certified copy journal is essential to appear for the practical examination.

1	Practical Question 1	20
2	Practical Question 1	20
3	Journal	5
4	Viva Voce	5

OR

1	Practical Question 1	40
3	Journal	5
4	Viva Voce	5