Course: MBA **Semester:** 1

Prerequisite: Proficiency in English language and familiarity with fundamental concepts of functional English grammar.

Rationale : This course builds professional communication and employability skills, bridging academic learning with workplace expectations.

Teaching and Examination Scheme

Teaching Scheme					Examination Scheme					Total
Lecture Hrs/Week	Tutorial Hrs/Week	Lab Hrs/Week	Seminar Hrs/Week	Credit	Internal Marks			Marks EYE		
					T	CE	P	T	P	
1	-	2	-	2	40	-	20	60	30	150

SEE - Semester End Examination, **T** - Theory, **P** - Practical

Course Content

W - Weightage (%) , **T** - Teaching hours

Sr.	Topics	W	T
1	**Self-Introduction – SWOT & SOAR, Body Language and Professionalism**Purpose of Self-Introduction (Academic vs Professional) Structure of an Effective Self-Introduction Understanding SWOT & SOAR (Concept and Use) Professional Appearance and Grooming Basics of Positive Body Language	10	1
2	**Sentence correction, One-word substitution, Para jumbles, Cloze Passages, Vocabulary-Synonyms, Antonyms**Common Grammar Rules for Sentence Correction Parts of Speech & Usage Logical Flow in Paragraphs Contextual Vocabulary Learning Synonyms & Antonyms – Word Relationships	10	2
3	**Writing Skills: Paragraph Writing, Letter Writing - Formal, Email Writing**Paragraph Writing – Structure & Types Letter & Email Writing – Formal Formats & Do's/Don'ts	10	2
4	**Reading Comprehension (Level of Difficulty - Intermediate) – Tips and Practice**Types of Reading (Skimming, Scanning, Intensive) Identifying Main Idea & Supporting Details Inference and Vocabulary from Context Time Management in Reading Tests	10	1
5	**Extempore**Meaning and Importance of Extempore Structure of a Short Speech Handling Stage Fear Time Management Techniques	10	1
6	**Goal Setting**Meaning and Importance of Goals Short-term vs Long-term Goals SMARTER Goal Concept Personal vs Professional Goals	10	1
7	**Team Building and Convincing Skills**Meaning of Teamwork Roles in a Team Qualities of a Good Team Member Basics of Persuasion & Convincing	10	1

	Ethical Convincing vs Manipulation		
8	**Negotiation Skills**Meaning and Importance of Negotiation Types of Negotiation Win-Win vs Win-Lose Approach Negotiation Stages Communication in Negotiation	**10**	**2**
9	**Presentation Skills - Do's and Don'ts - Making effective Presentations**Purpose of Presentations Structure: Introduction–Body Conclusion Do's and Don'ts of Slides Common Presentation Mistakes Visual Aids & Audience Engagement	**10**	**2**
10	**Debate**Format of a Debate Rules and Roles (Proposition/Opposition) Argument Building & Rebuttal Logical Reasoning in Debate	**10**	**2**
	Total	**100**	**15**

Suggested Distribution Of Theory Marks Using Bloom's Taxonomy

Level	Remembrance	Understanding	Analyze	Evaluate
Weightage	10	15	55	20

Reference Books

1.	**Business Communication Today**
2.	**Objective General English. S. Chand,** By Aggarwal, R. S.
3.	**English Grammar in Use** By Murphy, Raymond | Cambridge University Press, Pub. Year 2019
4.	**English Grammar in Use** By Raymond Murphy
5.	**Soft Skills: Key to Success in Workplace and Life.** By Meenakshi Raman
6.	**Basic Communication** By Poe & Fruchling | AITBS
7.	**The 7 Habits of Highly Effective People** By Stephen R. Covey | Free Press
8.	**Personality Development and Soft Skills** By Barun K Mitra | Oxford, 2011
9.	**Communication Skills. Oxford University Press, 2017.** By Raman, Meenakshi, and Sangeeta Sharma.
10.	**Basics of Business Communication** Cambridge University Press, | South Asian Edition

Course Outcomes

At the end of this course Students Will be able to:	
1	Remember key concepts related to self-introduction, SWOT/SOAR analysis, body language, professionalism, and basic vocabulary including synonyms, antonyms, and one-word substitutions.
2	Understand sentence structure and contextual meaning by interpreting para-jumbles, cloze passages, vocabulary usage, and intermediate-level reading comprehension passages.
3	Analyse effective communication techniques through paragraph writing, formal letter writing, email drafting, extempore speaking, debates, and structured self introductions.
4	Analyse communication situations to enhance presentation effectiveness by identifying do's and don'ts, evaluating body language, team dynamics, convincing strategies, and negotiation approaches
5	Evaluate personal communication performance by assessing goal-setting strategies, teamwork contributions, negotiation effectiveness, and overall professionalism for continuous improvement.

List of Practical

1.	**Self-Introduction – SWOT & SOAR, Body Language and Professionalism** 60-second Self-Introduction Practice SWOT/SOAR Worksheet Activity Posture, Eye Contact & Hand Movement Drill Peer Feedback on Professional Presence
2.	**Sentence Correction, One-word substitution, Para jumbles, Cloze Passages, Vocabulary-Synonyms, Antonyms** Sentence Correction Worksheets One-word Substitution Quiz Para-jumble Rearrangement Game Cloze Passage Practice Sets Vocabulary Building through Word Games Daily Word Challenge Activity
3.	**Writing Skills: Paragraph Writing, Letter Writing - Formal, Email Writing** Write a Paragraph from a Given Topic Write a Leave/Application Letter Draft a Professional Email Identify Errors in paragraph, letter and email
4.	**Reading Comprehension (Level of Difficulty - Intermediate) – Tips and Pract** Timed Reading Comprehension Practice Question-type Analysis
5.	**Extempore** Pick-and-Speak Activity 1-minute Extempore Rounds
6.	**Goal Setting** Personal Goal-Setting Worksheet SMARTER Goal Conversion Activity
7.	**Team Building and Convincing Skills** Team Task Challenge Problem Solving Group Activity Convince the Group Exercise
8.	**Negotiation Skills** Role-play: Buyer–Seller Negotiation Win-Win vs Win-Lose Approach Salary Negotiation Simulation
9.	**Presentation Skills - Do's and Don'ts - Making effective Presentations** Presentation Skills – Slide Preparation & Tips Create a Short Presentation Individual Presentation
10.	**Debate** Classroom Debate Sessions

Course: M.Sc. (IT) **Semester:** 1

Prerequisite: : Knowledge of C, C+

Rationale : To acquire the fundamental knowledge of Java programming

Teaching and Examination Scheme

Teaching Scheme					Examination Scheme					Total
Lecture Hrs/Week	Tutorial Hrs/Week	Lab Hrs/Week	Seminar Hrs/Week	Credit	Internal Marks			External Marks		
					T	CE	P	T	P	
3	0	2	0	4	20	20	20	60	30	150

SEE - Semester End Examination, **T** - Theory, **P** - Practical

Course Content

W - Weightage (%) , **T** - Teaching hours

Sr.	Topics	W	T
1	**Introduction to Java Data Types, Operators, Statements:** Paradigms of programming languages, Evolution of OO methodology, Basic concepts of OO approach, Comparison of object oriented and procedure oriented approaches. Concepts of OOP - classes and objects, abstraction and encapsulation, inheritance, polymorphism. Features of the Java language, Java environment, Object oriented programming in Java, Java program structure, Java and unicode, Data types, Variables and arrays - data types in Java, literals, characters, variable declaration, symbolic constants. Type casting operations in Java, Arithmetic operators, Basic assignment operators, Relational operators, Boolean logical operators, Ternary operator, Operator precedence. Control statements - Java's selection statements, switch, nested switch, iteration constructs, continue, return.	20	8
2	**Inheritance, Sub Classing, Package:**Inheritance Concepts – defining sub classes, method overriding, using super keyword, Variable shadowing, Method and variable binding, Using final keyword, Abstract classes and interfaces. Object class, Packages - Creating package, CLASSPATH environment variable, access specifiers, Access Control / visibility.	20	8
3	**Exception, Collection Frameworks:**Exception handling - types of exceptions, Throwable class, Keywords - try, catch, throw, throws and finally, Nested try statements, Java built in exceptions, User defined exceptions. Collection framework – Collections, List, Set, Enumeration, Iterator, ArrayList.	15	8
4	**IO and Multi Threading:**java.io- File class, Creating directory, Input/Output basics, Streams (byte and character), Reading from and writing to console, Reading and writing files, PrintWriter Class, the transient modifier, RandomAccessFile, Introduction to multi-threading, Thread class and execution of thread, Runnable interface, ThreadGroup, Daemon threads, Thread states	15	8
5	**GUI Programming and Applets:**Introduction to AWT and Swing, Fundamentals of applets, Applet class, Applet life cycle, A simple banner applet, getDocumentBase(), getCodeBase(), showDocument(), AppletContext and AppletStub interface, Working with frames, windows, graphics, color, fonts. AWT controls - buttons, checkbox, choice, list and textField. Layout Managers - Flow Layout, Grid Layout and Border Layout.	30	13

	User interface events - event classes and event listener interfaces, Adapter classes.		
	Total	**100**	**45**

Reference Books

1.	**The Complete Reference Java J2SE (TextBook)** By Herbert Schildt | TMH Publishing Company Ltd | 5th Edition
2.	**Core Java Volume 1** By Cay Horstmann and Gary Cornell | Pearson Education | 8th Edition
3.	**The class of JAVA** By Pravin Jain | Pearson

Course Outcome

After Learning the Course the students shall be able to:
1. implement object oriented principles using Java. 2. identify errors and implement exception handling mechanism. 3. develop programs using multi-threading concepts. 4. manage source code in Java packages. 5. design and develop java applications using Applets, Abstract Window Toolkit and Swing API.

List of Practical

1.	**Basic datatype and looping** Write a program for swapping and find a factorial value. Perform swapping without using third variable.
2.	**Looping and Control structure** Write a program to accept a number from the user through command line and display whether the given number is palindrome or not.
3.	**Array** Write a program to accept an array of integer from the user through command line and find prime numbers from the array.
4.	**Class** Create a class Stack that defines an integer stack that can hold 10 values. Perform push and pop actions in a stack.
5.	**Inheritance** Write a program to create a class Publisher with attributes publisher name and publisher id. Derive a subclass Book with attributes bookname, bookid and author name. All these data should be entered by the user. Create two methods getdata() and showdata() to display the details of book and publisher.
6.	**Method Overloading** Write a program to create a class with two methods with same name addfunc(), one accepting two integer parameters and other accepting two double parameters. When method is called, the appropriate method should be selected depending on parameters passed(method overloading).
7.	**Super and this keyword** Declare a variable called x with integer as the data type in base class and subclass. Make a method named as show() which displays the value of x in the superclass and subclass.
8.	**Final class, abstract class and interface** Write a program to calculate the area, circumference and volume for all shapes. [Perform this application using final class, abstract class and interface]
9.	**Exception handling** Write a program to enter two integers from the command line and display the division of those two numbers. Handle all the

	exceptions (i.e. ArrayIndexOutOfBoundsException, NumberFormatException, ArithmeticException) for invalid arguments passed.
10.	**File Handling** Write a program to perform following actions and store output in file: 1. Accept strings from user, convert it into uppercase and store it in a file. 2. Write double value to a text file and also display the date on which the application was run inside the file. 3. Delete a given file or directory and display appropriate message
11.	**Collection Framework** Write a program for creating an ArrayList, add the elements in array list and then obtain an array from ArrayList and display the contents and sum of those numbers.
12.	**Collection Framework** Create a class named Address and define name, city and state as the data members of this class. Create another class named Maillist and add the elements to the linked list and display the contents using Iterator interface.
13.	**Calendar class** Create a Gregorian Calendar. Display current date and time in default locale and time zone. Find out whether the current year is a leap year or not.
14.	**Regular expression** Write a program to find integers and decimal value from the string by using a pattern of regular expression.
15.	**Thread and Runnable interface** Write a multi-threaded program which sets the priority of threads and gets the name of threads.
16.	**Thread (sleep method)** Write a program to calculate sum and factorial of numbers using sleep method.
17.	**Multi-threading** Write a program to create multi-threaded application to perform banking tasks. [For Example: Withdrawal and Deposit from Joint account.]
18.	**AWT and Swing** Write a program to create a Menubar and create 2 Menus File and Edit. Involve New, Open, Close as items in the File menu and then add a separator, then further add Save, Save As and again add separator and add another menu item named Print. Add all these items in File menu. Add Line, Rectangle and Circle as the menu items and add Radio Button before these menu items and add a separator. Then add Red, Green and Blue as menu items and add Check Box before these menu items.
19.	**Applet and AWT** Create Java Applets to perform following tasks: 1. To display simple calculator 2. To write the content of the text area in the file, whose name is given in the text box using Frame 3. To draw rectangle to the applet when mouse is dragged.
20.	**AWT and Swing** Write a program to create 3 radio buttons named C++, Java and Pascal and add on JPanel. The user is asked to identify which of them is not an OOP language. When the user selects on choice, the program responds by displaying whether the selection is correct or wrong.

Course: M.Sc. (IT) **Semester:** 1

Prerequisite: Knowledge of programming language (C/C++).

Rationale : To provide understanding of basics of data structures and their operation for efficient storage and retrieval of data

Teaching and Examination Scheme

Teaching Scheme					Examination Scheme					Total
					Internal Marks			External Marks		
Lecture Hrs/Week	Tutorial Hrs/Week	Lab Hrs/Week	Seminar Hrs/Week	Credit	T	CE	P	T	P	
4	0	2	0	5	20	20	20	60	30	150

SEE - Semester End Examination, **T** - Theory, **P** - Practical

Course Content

W - Weightage (%) , **T** - Teaching hours

Sr.	Topics	W	T
1	**Introduction:**Primitive and non-primitive data structures, String manipulation and pattern matching, Storage representation of strings, Text handling, Key Word In Context (KWIC) indexing, Arrays, Storage structure for arrays, Special types of arrays – triangular and sparse	10	7
2	**Stack and Queue:Stack, Stack operations, Applications of stack - recursion, polish notations - prefix, infix, postfix, Algorithms of stack applications, Introduction to queue, Algorithms and implementation of simple queue, Circular queue, Double ended queue, Priority queue**	20	12
3	**Linked List:** Linked list, Algorithms and implementation of singly linked list, Doubly linked list, Circular linked list, Operations on linked list, Applications – polynomial representation, addition of two polynomials.	20	8
4	**Trees:**concept and terminologies of tree, General tree, Binary tree and its storage representation, Binary search tree and its operations – create, insert, delete, Traversal of tree - inorder, preorder, postorder, Threaded tree, B tree and B+ tree, Height balanced tree - AVL tree, Rotations in AVL tree, Applications – heap tree, expression tree.	20	12
5	**Graph**Concept and terminologies of graph, Representation of graph - adjacency matrix, adjacency lists, Introduction to graph traversal - Depth First Search (DFS), Breadth First Search (BFS), Introduction to spanning tree.	10	5
6	**Searching, Sorting and Hashing:**Linear search, Binary search, Bubble sort, Selection sort, Insertion sort, Shell sort, Quick sort, Heap sort, Merge sort, Radix sort, Hashing, Hashing functions, Collision resolution techniques.	20	13
	Total	100	57

Reference Books

1.	**An Introduction to Data Structures with Applications** By Jean-Paul Tremblay, Paul G. Sorenson | Tata McGraw-Hill | 2nd Edition, (2007)
2.	**Introduction to Algorithms** By Thomas Cormen, Charles Leiserson, Ronald Rivest, Clifford Stein | PHI publication
3.	**Data Structures using C and C++** By Tanenbaum | PHI
4.	**Expert Data Structures with C** By R. B. Patel
5.	**Theory and Problems of Data Structures** By Seymour Lipschutz | Schaum's Outline Series
6.	**Data Structures Through C++** By Yashavant Kanetkar | BPB

Course Outcome

After Learning the Course the students shall be able to:

1. describe the significance of various linear and non-linear data structures such as arrays, stack, queue, linked list, trees and graph.
2. identify the appropriate data structure for a given problem.
3. construct most suitable data structure to solve a problem by considering various problem characteristics such as data size and various type of operations.
4. design and implement various techniques for searching, sorting and hashing

List of Practical	
1.	**Write a program to perform various stack operations using array**
2.	**Write a program to convert infix expression to prefix and postfix expression using stack**
3.	**Write a program to perform insert and remove operations on following** a. Simple Queue b. Circular Queue c. Priority Queue
4.	**Write a program to perform Double Ended Queue [Input Restricted / Output Restricted]**
5.	**Write a program to create a singly link list in FIFO & LIFO form**
6.	**Write a program to perform following singly link list operations a. insert b. delete c. search d. reverse**
7.	**Write a program to create a doubly link list in FIFO & LIFO form**
8.	**Write a program to perform following doubly link list operations a. insert b. delete c. search d. reverse**
9.	**Write a program to add two polynomials**
10.	**Write a program to perform following circular link list operations a. insert b. delete**
11.	**Write a program to create a binary search tree and print its element in a. Inorder b. Preorder c. Postorder**
12.	**Write a program for insertion of a node in B tree / B+ tree**
13.	**Write a program to create a graph in a adjacency list structure traverse it in a. DFS b. BFS**
14.	**Write a program to perform following sort** a. Bubble Sort b. Selection Sort c. Insertion Sort d. Shell Sort e. Quick Sort f. Heap Sort g. Merge Sort h. Radix Sort
15.	**Write a program to search an element using a. Linear Search b. Binary Search**

Course: M.Sc. (IT) **Semester:** 1

Prerequisite: Basic Operating System Knowledge

Rationale : Linux Operating System is Open source and freely distributed Operating System (O.S). Apart from the fact that it's freely distributed, Linux's functionality, adaptability, and robustness make it highly suitable for the server platform. The course aims to provide knowledge in the basics of Linux, shell, and command line essentials.

Teaching and Examination Scheme

Teaching Scheme					Examination Scheme					Total
					Internal Marks			External Marks		
Lecture Hrs/Week	Tutorial Hrs/Week	Lab Hrs/Week	Seminar Hrs/Week	Credit	T	CE	P	T	P	
3	0	2	0	4	20	20	20	60	30	150

SEE - Semester End Examination, **T** - Theory, **P** - Practical

Course Content

W - Weightage (%) , **T** - Teaching hours

Sr.	Topics	W	T
1	**Introduction to the UNIX/Linux Environment** History and evolution of the UNIX and GNU/Linux operating systems and their derivatives The present landscape of UNIX/Linux derivatives Comparison between free and proprietary software Pros and cons of using Linux, Different types of licenses and IPR issues A comparison of currently popular Linux distributions An overview of the Linux environment, Introduction to the Bourne Again SHell (bash) The vim editor	20	8
2	**Introduction to The CLI** Shell environment, commands, syntax, options, getting help Basic commands and utilities File system navigation and manipulation Process management Command line processing, I/O redirection and filters	25	8
3	**Basic Shell Scripting** Basics of shell scripting ,The built-in constructs of the shell , The basics of filters and regular expressions. Some utility programs Combining the available facilities to produce new ones	25	8
4	**Advanced Shell Scripting – I** Using advanced features of the shell The grep filter More utility programs The sed filter Examples	10	8
5	**Linux System Administration** System structure**,** Installation on and using different media, live systems Partitioning, formatting, file systems, Devices and file system management User management, Managing basic file system permissions Configuring the system - display, network, desktop, etc. System configuration files, System startup and shutdown, runlevels Updating the system and installing and updating packages, Using along with other operating systems, Backup and system recovery Tools and shell scripting for the administrator An introduction to common applications and file formats on Linux An introduction to tools for Linux development	20	8
	Total	100	40

Reference Books

1.	**Unix Shell Programming** By Yashwant Kanetkar | BPB publications
2.	**Essential System Administration: Tools and Techniques for Linux and Unix Administration** By Eleen Frisch
3.	**Linux Administration: A Beginner's Guide** By Wale Soyinka
4.	**The Design of UNIX Operating System** By Bach M J | Prentice Hall India | First Edition

Course Outcome

After Learning the Course the students shall be able to:

Students will be able to:
1. Explain the basic commands of linux operating system and can write shell scripts.
2. Create file systems and directories and operate them.
3. Create processes background and fore ground etc. by using system calls
4. Create shared memory segments, pipes ,message queues and can exercise inter-process communication.
5. Describe the Process Management algorithms, Process Management system calls, Inter Process Communication and CPU Scheduling algorithms

List of Practical

1.	**To Study basic & User status Unix/Linux Commands**
2.	**Study & use of commands for performing arithmetic operations with Unix/Linux.**
3.	**Create a file and display how many lines, words and characters are present in that file using shell command**
4.	**Give two files each of which contains names. use shell command to display only those names that are found on both the files.**
5.	**Implement shell command to find out the inode number of any desired file.**
6.	**Study & use of the command for changing file permissions.**
7.	**Write a pipeline of commands, which displays on the monitor as well as saves the information about the number of users using the system at same time**
8.	**Execute shell commands through vi editor**
9.	**Installation, Configuration & Customizations of Unix and Linux**
10.	**Write a shell script that accepts any number of arguments and prints them in the reverse order**
11.	**Write a shell script to find the smallest of three numbers that are read from the keyboard.**

Course: M.Sc. (IT) **Semester:** 1

Prerequisite: Knowledge of DBMS.

Rationale : This course is intended to give students advanced concepts of Relational Database Management System, security aspects of databases and introduction to other databases. Also concepts and practical aspects of data manipulation using stored procedures and triggers will be given.

Teaching and Examination Scheme

Teaching Scheme					Examination Scheme					Total
					Internal Marks			External Marks		
Lecture Hrs/Week	Tutorial Hrs/Week	Lab Hrs/Week	Seminar Hrs/Week	Credit	T	CE	P	T	P	
3	-	2	-	4	20	20	20	60	30	150

SEE - Semester End Examination, **T** - Theory, **P** - Practical

Course Content

W - Weightage (%) , **T** - Teaching hours

Sr.	Topics	W	T
1	**Object Relational and Extended Relational Database** Database design for an ORDBMS, Nested relations and collections, Storage and access methods, An overview of SQL3, Systems comparison of RDBMS, OODBMS and ORDBMS.	20	11
2	**PL/SQL, Cursor and Trigger** Basic code structure, Variables, Conditional statements, looping (loop statements, while loops, for loops, cursor FOR loops), Triggers.	11	5
3	**Stored Procedures** Understanding the main features of stored procedures, stored procedure architecture, and advantages of using procedures, stored procedures – functions, procedures and packages.	17	8
4	**Database Tuning, Indexing and Security** Physical Database Design in Relational database, Overview of the database tuning and relational systems. Types of single-level ordered indexes, Primary index, cluster index, secondary index, multilevel index. Database security and its issues, Granting and Revoking privileges, Role-based access control for multilevel security, encryption and PKI, challenges in database security.	16	7
5	**Introduction to NoSQL and MongoDB Basics**Introduction to NoSQL and MongoDB Basics Overview of NoSQL Features Of NoSQL Types of NoSQL Differences between RDBMS and NoSQL Introduction to JSON and jQuery JSON Introduction, JSON vs XML JSON example JSON Object JSON Array, JSON Comments JQuery : Introduction JQuery Installation JQuery Syntax JQuery Selectors JQuery Event-blind, blur, changeJQuery Effects-clear Queue, Delay, Animate	20	10
6	**MongoDB and CRUD with Node.js** Introduction, MongoDB data type, Installation, MongoDB Data modelling,	16	7

	Operators, commands, CRUD Operations Connectivity, Node.js with MongoDB.		
	Total	**100**	**48**

Reference Books

1.	**SQL, PL/SQL – The Programming Language** By Ivan Bayross | BPB Publications
2.	**Database System Concepts** By Silberschatz, Korth, Sudarshan | McGraw Hill Publication | 4th Edition
3.	**Fundamentals of Database Systems** By Ramesh Elmasari, Shamkant B. Navathe | Pearson Education

Course Outcome

After Learning the Course the students shall be able to:

1. Explain the design principles of ORDBMS and compare the architectures of RDBMS, OODBMS, and ORDBMS. Analyze nested relations, collections, and SQL3 enhancements for efficient database modeling. Develop PL/SQL programs using variables, conditional and loop statements, and apply cursors and triggers to automate database operations.
2. Demonstrate the creation and execution of stored procedures, functions, and packages, and explain their architectural and performance benefits. Apply techniques for physical database design and indexing. Evaluate database tuning strategies and implement security mechanisms including roles, privileges, encryption, and PKI. Explain NoSQL concepts, features, and types.
3. Construct and manipulate JSON data and implement client-side operations using jQuery.
4. Implement MongoDB data models and perform CRUD operations using Node.js for developing full-stack data applications.

List of Practical	
1.	**Write a PL/SQL block using nested IF to grade a student.**
2.	**Write a PL/SQL program to find the factorial of a number.**
3.	**Write a SQL query to create a table named employee with the following structure: Write INSERT queries to add the following records into the employee table:** • emp_id: Number (Primary Key) • emp_name: Character field up to 50 characters • department: Character field up to 30 characters • salary: Numeric field with up to 10 digits, including 2 decimal places emp_id emp_name department salary 101 Ravi Sharma HR 25000.00 102 Asha Mehta IT 45000.00 103 John Dsouza Finance 15000.00 104 Neha Patel IT 95000.00 **Note:Based on this data perform other practicals mentioned below.**
4.	**Create a PL/SQL Block to find Bonus Calculation Based on Salary and Display employee name and calculate 10% bonus.**
5.	**Create a Cursor to display List Employees with Salary greater than 20000.**
6.	**Create a cursor to count total employees department-wise. [Hint: Display total number of employees in each department using an implicit cursor with GROUP BY.]**
7.	**Create a Cursor to Display Employee Names Starting with N. [Hint: Use cursor to display names of employees beginning with N]**
8.	**Create a BEFORE trigger to stop inserting/updating low salaries.**
9.	**Create FUNCTION to Calculate Annual Salary. [Hint: To create a function that returns annual salary of an employee based on employee ID.]**
10.	**Create PROCEDURE to Update Salary by Percentage. [Hint: Write a procedure to increase salary by a given percentage.]**
11.	**Create a package to hold related employee operations (function + procedure).**
12.	**WAP to implement database Connectivity using MongoDB**
13.	**WAP to implement CRUD operations using MongoDB**
14.	**WAP to implement JSON object and Array**
15.	**WAP to implement an event o click by using jQuery.**
16.	**WAP to implement to select all elements and change the background colour.**

Course: MCA **Semester:** 2

Prerequisite: Knowledge of Mathematics up to graduate level.

Rationale : The course provides introductory Numerical and statistical techniques, which will be useful for forecasting about software and hardware in computer field.

Teaching and Examination Scheme

Teaching Scheme					Examination Scheme					Total
Lecture Hrs/Week	Tutorial Hrs/Week	Lab Hrs/Week	Seminar Hrs/	Credit	Internal Marks			External Marks		
					T	CE	P	T	P	
3	1	0	-	4	20	20	-	60	-	100

SEE - Semester End Examination, **T** - Theory, **P** - Practical

Course Content

W - Weightage (%) , **T** - Teaching hours

Sr.	Topics	W	T
1	**Error, Solution of Algebraic & Transcendental Equations** Error, Type of errors, Significant digits, Round-off, Error in the approximation of function Descartes's rule of sign, Intermediate theorem, Bisection, False position, Secant method, Newton Raphson method, Rate of convergence of methods without proof.	12	6
2	**Matrix and Solution of System Linear Equations** Direct Methods: Gauss Elimination with Pivoting, LU decomposition Method, Iterative method:Gauss Jacobi method, Gauss Seidel method.	11	5
3	**Interpolation** Newton's Forward Difference Interpolation, Newton's Backward Difference Interpolation, Lagrange's Interpolation and its inverse interpolation, Newton's Divided Difference Interpolation.	12	6
4	**Numerical Integration** Newton cote's formula, Trapezoidal rule, Simpson's 1/3 and 3/8 rule.	6	3
5	**Solution of First Order Ordinary Differential Equations** Euler's method, Modified Euler's method and Runge – Kutta method.	6	3
6	**Curve Fitting** Least squares approximation, Linear least square approximation, Nonlinear least square approximation of higher order polynomial.	9	4
7	**Descriptive Statistics** Introduction to Statistics, Frequency distribution, Charts, Mean, Median, Mode, Percentiles, Variance, Standard deviation, Coefficient of variation, correlation coefficient.	10	5
8	**Probability & Probability Distribution** Introduction to probability, Experiments, Counting, Rules and assigning probabilities, Events and their probabilities, Some basic relationships of probability, Conditional, probability, Baye's theorem. Random Variables: discrete, continuous, discrete probability distributions, Expected value & variance, Uniform probability distribution, Binomial probability distribution, Poisson probability distribution, Normal probability distribution, Exponential probability distribution, Normal approximation of binomial probabilities.	17	8
9	**Statistical Inference-Testing of Hypothesisχ^2 Test** Introduction, Test of significance for large Samples: Difference between small &large Samples, Hypothesis testing for Single Population- mean, Proportion, Variance, Two-tailed test for Difference between the means of two samples, Standard error of the difference between two standard deviations, Tests of significance for small samples: The assumption of normality, Students' t-distribution, Properties & applications of t-distribution, testing difference between means of two samples (Independent Samples, Dependent Samples), Definition of chi-square, Degrees of freedom, chi-square distribution, Conditions for applying chi square test, Uses of chi-square Test,	17	8

	Misuse of chi-square test.		
	Total	**100**	**48**

Reference Books

1.	**Numerical Methods for Engineers (TextBook)** By Steven C Chapra, Raymond P Canale | Tata McGraw Hill Publication | 5th Edition
2.	**Mathematical Methods** By S.R.K. Iyenger, R.K. Jain | Narosa Publication
3.	**Statistics for business and economics** By Anderson, Sweeney, Williams | Thompson Publication | 9th edition
4.	**Statistics for Business and Economics** By Anderson, Sweeney, Williams | Ceenage Learning
5.	**Statistical Methods** By S P Gupta | S Chand & Sons | 30th edition

List of Tutorial

1.	**Error, Solution of Algebraic & Transcendental Equations**
2.	**Solution of System Linear Equations**
3.	**Interpolation**
4.	**Numerical Integration**
5.	**Solution of First Order Ordinary Differential Equations**
6.	**Curve Fitting**
7.	**Descriptive Statistics**
8.	**Probability & Probability Distribution**
9.	**Statistical Inference-Testing of Hypothesisχ^2 Test**

Course: M.Sc. (IT) **Semester:** 1

Prerequisite: Understanding of Object Oriented Design, UMl

Rationale : To analyze the requirements of end users, Understands the software development life cycle (SDLC) along with many other process models, gain understanding of test strategies and different types of testing, gain understanding of different agile processes

Teaching and Examination Scheme

Teaching Scheme					Examination Scheme					Total
Lecture Hrs/Week	Tutorial Hrs/Week	Lab Hrs/Week	Seminar Hrs/Week	Credit	Internal Marks			External Marks		
					T	CE	P	T	P	
3	0	2	0	4	20	20	20	60	30	150

SEE - Semester End Examination, **T** - Theory, **P** - Practical

Course Content

W - Weightage (%) , **T** - Teaching hours

Sr.	Topics	W	T
1	**Introduction to System and Software Development**Concept of system, Basic components of system, Information systems categories, Need of information system development. Study of different models, Software characteristics, Components, Applications, Layered technologies, Processes, Methods and tools, Generic view of software engineering, Classical Systems Development Life Cycle (SDLC) method.	15	6
2	**Requirements Analysis & Engineering**Requirement Analysis: Classical Systems Development Life Cycle (SDLC) method, Requirement determination, System Requirement Specification (SRS), Fact finding techniques. Process models - waterfall model, incremental, Evolutionary process models - prototype, spiral and concurrent development model. Requirement Engineering: Problem recognition, Requirement engineering tasks, , Use cases and functional specification, Requirements validation	20	10
3	**Structured System Analysis & Design:**System Analysis: Introduction to Structured System Analysis Development Methodology (SSADM), Tools for analysis - decision trees, decision tables, structured english, Data flow diagram, Entity Relationship (ER) diagram, Data dictionary. System Design: Design concepts, Design modeling, Software architecture, Data design, Architectural styles and patterns, Procedural design, Object oriented design.	20	10
4	**Agile Methodology:** Agile process, Extreme Programming (XP), Brief overview of other agile process models - adaptive software development, Scrum.	20	7
5	**User Interface Design**Concepts of UI, Interface design model, Internal and external design, Evaluation, Interaction and information display software.	5	3
6	**Planning a Software Project:**Management spectrum, People, Product, Process, Project, W5HHPrinciple, Importance of team management, Scope and feasibility, Effort estimation, Schedule and staffing, Quality planning, Risk management- identification, assessment, control,	15	6
7	**Case Tools and Study:**Introduction to CASE, Building blocks of CASE, Integrated CASE environment.	5	3
	Total	100	45

Reference Books

1.	**Software Engineering : A Practitioner's Approach (TextBook)** By Pressman R.S | TMH
2.	**Software Engineering** By Sommerville
3.	**Software Engineering** By Rajiv Mall | PHI
4.	**Software Engineering** By Pankaj Jalote | Wiley India
5.	**Software Engineering, An Engineering Approach** By Peters & Pedrycz | Wile-India
6.	**Software Engineering, Principles and Practice** By Jawadekar | TMH

Course Outcome

After Learning the Course the students shall be able to:
1. describe Software Development Life Cycle. 2. explain software development process in association with its fundamental principles and methodologies. 3. analyze & represent end user requirements and model requirement analysis using Unified Modeling Language. 4. prepare & represent software design and design software model using Unified Modeling Language.

List of Practical

1.	**Analyze a Library Management System and identify its system components — input, process, output, feedback, control, boundary, and environment — and document them in tabular form.**
2.	**Examine a College Examination System and identify the type(s) of Information System (TPS, MIS, DSS, EIS) it uses, with proper justification.**
3.	**Apply the Classical Systems Development Life Cycle (SDLC) by mapping each phase to the problem of an Online Admission System in correct sequence.**
4.	**Conduct requirement elicitation for an Online Voting System by performing one interview and one questionnaire, and document the elicited requirements.**
5.	**Classify and document exactly five functional and five non-functional requirements for an Online Examination System.**
6.	**Prepare a System Requirement Specification (SRS) document for a Library Management System including a minimum of five standard SRS sections.**
7.	**Design a Context Diagram, Level-0 DFD, and one Level-1 DFD for a Student Information System , ensuring proper balancing between levels.**
8.	**Develop an Entity-Relationship (ER) Diagram with at least four entities for a Banking System , clearly indicating attributes and relationships.**
9.	**Formulate exactly five user stories for an Online Food Delivery Application using standard Agile user-story format.**
10.	**Prepare a Software Project Plan for a Mini Software Project including project scope, effort estimation, schedule, and identification of three major risks.**

Course: M.Sc. (IT) **Semester:** 2

Prerequisite: Knowledge of Core Java Programming and HTML.

Rationale : To make the students aware of the data driven web applications, web services, MVC architecture and ORM concepts of modern web application development.

Teaching and Examination Scheme

Teaching Scheme					Examination Scheme					Total
Lecture Hrs/Week	Tutorial Hrs/Week	Lab Hrs/Week	Seminar Hrs/Week	Credit	Internal Marks			External Marks		
					T	CE	P	T	P	
4	0	4	0	6	20	20	20	60	30	150

SEE - Semester End Examination, **T** - Theory, **P** - Practical

Course Content

W - Weightage (%) , **T** - Teaching hours

Sr.	Topics	W	T
1	**Java Database Connectivity**Introduction to JDBC, JDBC architecture. Common JDBC Components - DriverManager, Driver, Connection, Statement, ResultSet, SQLException. Types of JDBC drivers, Establishing connection with database, Submitting queries and obtaining results, ResultSetMetadata Interface, Precompiled statements, CallableStatement, JDBC transactions.	15	8
2	**Java Servlets**Configuring Java Development Environment, Downloading and configuring Apache Tomcat web server. Introduction to servlets, Servlet life cycle, Exploring servlet API, Basic servlet structure, Handling client requests, Generating response, Working with cookies, Session tracking. Servlet filters - Introduction to filters, creating custom filters, mapping filters to servlets, mapping filters to specific URL pattern.	20	10
3	**Java Server Pages**Introduction to JSP, Need of JSP, Advantages of JSP over servlet, Life cycle of a JSP page, JSP basic tags: declaration tag, scriptlet tag, expression tag. JSP implicit objects - out, request, response, application, session, pageContext, page, exception. Introduction to java beans, Reading and writing properties of JavaBeans in JSP pages. JSP 2.0 expression language - Use of EL, Invoking Expression Language, Preventing Expression language evaluation, Accessing scoped variables. Custom tag libraries - tag handlers, tag library descriptor. JSP standard tag library (JSTL): Downloading and installing JSTL in Java web application, Core tags (c:out, c:forEach, c:if, c:choose, c:set,c:remove, c:import, c:catch).	25	12
4	**Java Web Services**Introduction to web services, Architecture of web services, Types of web services - SOAP web services, REST web services SOAP v/s REST, Introduction to REST, Configuring RESTful web service using Jersey framework in java web application, RESTful architectural principles, HTTP method and URI matching. JAX-RS injection - PathParam, QueryParam, FormParam. Introduction to Server responses and exception handling, JAX-RSclient API.	20	12
5	**Introduction to Spring and Hibernate**Introduction to Spring MVC Framework, Spring MVC Architecture and Components, Advantages of Spring MVC, Modules Of Spring Framework. Spring Application Introduction to Hibernate, Architecture of hibernate, Hibernate O/R	20	12

	mapping, Configuring hibernate development environment, Implementing hibernate O/R Mapping, Introduction to Hibernate Query Language (HQL).		
	Total	**100**	**54**

Reference Books

1.	**Java Server Programming Java EE6 Black Book (TextBook)** Dreamtech Press
2.	**Core Servlets and Java Server Pages Volume-1** By Mary Hall and Larry Brown | Prentice Hall | 2nd Edition
3.	**Core Servlets and Java Server Pages Volume-2** By Marty Hall, Larry Brown and Yaakov Chaikin | Prentice Hall | 2nd Edition
4.	**RESTful Java with JAX-RS 2.0** By Bill Burke | O'Reilly | 2nd Edition

Course Outcome

After Learning the Course the students shall be able to:

1. design and develop data driven web applications using JDBC, servlet and JSP API.
2. develop MVC based web applications using Struts.
3. develop and integrate web services and web clients.
4. describe significance of Object Relational Mapping (ORM) in Java context.

List of Practical

1.	**JDBC Connectivity** Develop a java application using concept of JDBC for user login. User will enter username and password. Application will match for the same in database table. If match is found then display message "Successful Login" and if not found then display message "Invalid Username and password ".
2.	**JDBC Connectivity & CRUD Operations** Develop a Menu driven java application for student information which will create the table with appropriate columns. Menu will contain the options like insert, update and delete. Based on the option, data will be inserted or updated or deleted from table based on student id (student_id will be primary key). Display appropriate message for each operation.
3.	**Data Driven GUI Application** Develop an application using GUI for user registration. In first form user will enter the data for registration (e.g. Name, Birthdate, Email_id, Phone_No). When user clicks on submit button, data will be inserted into database and user will be redirected to another form. In this form data will be selected from database and displayed in proper format.
4.	**A Data Driven Servlet Application** Develop a data driven servlet application for user authentication. Also display header values of request object and response object.
5.	**Servlet - Session Management** Develop a servlet application for implementing student grading system. In this system on first page student will enter his data (name, id and department). By clicking on next button he will be redirected to another page. On the second page he will enter marks of all 6 subjects. By clicking on "Generate result" he will be redirected to next page. On this page all information of student will be displayed along with the result (pass or fail) in appropriate color. Also display the name of student on each page he visits. (Use session management)
6.	**Servlet - Request Redirection** Develop an application for the following: User should be redirected to the URL entered in location text box.

7.	**JSP - User Authentication** Develop a data driven JSP application for user authentication.
8.	**JSP - Java Beans** Develop an application for Bank Account information using Java Beans. This application will perform various operation on bank account like withdraw, deposit and displaying the information.
9.	**JSP - Custom Tags** Develop a custom JSP tag which accepts 10 numbers from user and sorts them in specified order.
10.	**JSP - Application Context** Develop an application that will allow maximum 3 users to access the application. If number of users exceeds than limit then user will be redirected to the error page.
11.	**Simple RESTful Service** Develop a simple JAX-RS service that provides currency conversion.
12.	**Data Driven RESTful Service** Develop a RESTful service to accept student registration data sent from a web form.
13.	**RESTful Web Service - CRUD Operations** Develop a JAX-RS service that provides CRUD database operation interface for a phonebook. Call appropriate service from servlet/jsp pages to perform data oriented operations.
14.	**JAX-RS Client** Develop a JAX-RS client that consumes RESTful service developed in Program-11. Utilize the client in UI layer (JSP pages).
15.	**JAX-RS Client** Develop a RESTful service to perform authentication of the user. The JAX-RS client will be utillized in servlet and provide interface between jsp page and web service to perform authentication.
16.	**Struts2 - User Authentication** Develop a struts 2 application that accepts user name and password from user. If both values are valid then user will be redirected to home page otherwise redirected on error page.
17.	**Struts2 - User Registration and Authentication** Develop a data driven Struts 2 application that accepts registration details from user, i.e. name, contact number, address, user id and password. After successful registration, application redirects user to log in page. If user provides valid authentication details then application will redirect user to home page and will display welcome message, otherwise user will be redirected to error page.
18.	**Struts2 - Data Driven Application** Develop a phone book using struts 2. Application accepts data of contacts like name, address, contact number and email id. The application should provide features to update, delete and list data of contacts into/from database.
19.	**Struts2 - Interceptors** Create a logging service using struts 2 to monitor log in and log out timestamp of the users. (Hint: utilize interceptors)
20.	**Struts2 - Hibernate Integration** Develop an application using struts 2 that provides insert, update, delete and list operations for Student data like, name, roll number, semester, course, result (%). The data oriented operations must be performed through hibernate.

Course: M.Sc. (IT) **Semester:** 2

Prerequisite: Basics of computer fundamentals.

Rationale : Demonstration of application layer protocols Discuss transport layer services and understand UDPand TCP protocols Explain routers, IP and Routing Algorithms in network layer Disseminate the Wireless and Mobile Networks covering IEEE 802.11 Standard.

Teaching and Examination Scheme

Teaching Scheme					Examination Scheme					Total
Lecture Hrs/Week	Tutorial Hrs/Week	Lab Hrs/Week	Seminar Hrs/Week	Credit	Internal Marks			External Marks		
					T	CE	P	T	P	
4	-	2	-	5	20	20	20	60	30	150

SEE - Semester End Examination, **T** - Theory, **P** - Practical

Course Content

W - Weightage (%) , **T** - Teaching hours

Sr.	Topics	W	T
1	**Networking Fundamentals**Introduction, Data & Information, Data Communication-Characteristics of Data Communication, Components of Data Communication, Data Representation, Data Flow- Simplex, Half Duplex, Full Duplex, Computer Network- Categories of a network, Protocol- Elements of a Protocol, Networking Standards, Reference Models- OSI Model, TCP/IP Model, Comparison of OSI and TCP/IP Model.	**20**	**12**
2	**The Data Link Layer**Design Issues: Framing, Error Control, Flow Control, Protocols: FDDI, CDDI, Frame Relay, ATM, 802.11, PPP, HDLC. The Medium Access Sub-Layer Multiple Access Protocols: ALOHA, CSMA, Ethernet: Switched Ethernet, Fast Ethernet, Gigabit Ethernet, DLL Switching: Internetworking, Repeaters, Hubs, Bridges, Switches, Routers, Gateways, Virtual LANs	**25**	**12**
3	**The Network Layer**Design Issues, Routing Algorithms: Link State Routing, Distance Vector Routing, Flooding, Routing Protocols: RIP, IGRP, EIGRP, OSPF, Internetworking: Tunneling, Fragmentation, IPV4, IPV6 Basics, BGP. The Transport Layer Protocols: UDP, TCP, Headers.	**20**	**10**
4	**The Application Layer**DNS: The DNS Name Space, Name Servers-Mail: SMTP, POP3, HTTP, FTP, Telnet, Network Management: SNMP	**10**	**6**
5	**Network & System Security**Authentication Applications: Kerberos X.509, Directory Authentication Service, Electronic Mail Security, Pretty Good Privacy (PGP), S / Mime, Security: Architecture, Authentication Header, Encapsulating Security Payloads, Combining Security Associations, Key Management, Web Security: Secure Socket Layer & Transport Layer Security, Secure Electronic Transaction (Set), System Security: Intruders, Viruses, Firewall Design Principles, Trusted Systems	**25**	**14**
	Total	**100**	**54**

Reference Books	
1.	**Computer Networking: A Top-Down Approach** By James F. Kurose, Keith W. Ross.
2.	**Computer Architecture** By Parhami, Behrooz | Oxford publication, Latest Edition ISBN: 978-0-19-808407-5
3.	**Computer Networks : A Systems Approach** By Larry L. Peterson & Bruce S. Davie | Morgan Kaufmann
4.	**"Computer networks"** By Andrew Tanenbaum | Prentice Hall

Course Outcome
After Learning the Course the students shall be able to:
1. Explain fundamental concepts of data communication, networking standards, protocols, and reference models including OSI and TCP/IP. 2. Analyze data link layer and medium access control mechanisms by applying framing, error control, flow control, multiple access protocols, and networking devices. 3. Apply network and transport layer concepts to analyze routing algorithms, internetworking techniques, IP addressing (IPv4/IPv6), and TCP/UDP protocols. 4. Explain application layer services and network security mechanisms including DNS, email protocols, web services, authentication, encryption, and secure communication protocols.

List of Practical	
1.	**Study of different types of Network cables and Practically implement the cross-wired cable and straight through cable using clamping tool.**
2.	**Study of Network Devices in Detail.**
3.	**Study of network IP.**
4.	**Connect the computers in Local Area Network.**
5.	**Study of basic network command and Network configuration commands.**
6.	**Performing an Initial Switch Configuration**
7.	**Performing an Initial Router Configuration**
8.	**Configuring and Troubleshooting a Switched Network**
9.	**Connecting a Switch**

Course: M.Sc. (IT) **Semester:** 2

Prerequisite: Basic knowledge of web technologies (HTML/CSS), introductory understanding of C# or object-oriented programming concepts, familiarity with databases and SQL, and fundamental awareness of client–server architecture.

Rationale : To provide basic understanding of .NET framework and knowledge of developing dynamic and rich web application in conjunction with event handling, state management and data access.

Teaching and Examination Scheme

Teaching Scheme					Examination Scheme					Total
Lecture Hrs/Week	Tutorial Hrs/Week	Lab Hrs/Week	Seminar Hrs/Week	Credit	Internal Marks			External Marks		
					T	CE	P	T	P	
3	0	2	0	4	20	20	20	60	30	150

SEE - Semester End Examination, **T** - Theory, **P** - Practical

Course Content

W - Weightage (%) , **T** - Teaching hours

Sr.	Topics	W	T
1	**Introduction** ASP.NET framework, The origin of.NET Technology, Common Language Runtime (CLR), Common Type System (CTS), Common Language Specification (CLS), Microsoft Intermediate Language (MSIL), Just - in-time compiler, Framework base classes	10	6
2	**ASP.NET Controls, UI Design & Coding** ASP.NET server controls: Controls - buttons, textboxes, labels, checkboxes, radio buttons, list controls and other web server controls, web.config and global.asax files. Programming ASP.NET web pages: Web page, Data types and variables, statements, organizing code, and object-oriented basics of C#.	25	12
3	**ASP.NET Validation Controls, State Management & Navigation** **Validation Control:** Controls, Basic Validation Controls, Validation Techniques, and Using Advanced Validation Controls. **State Management:** Using view state, Using session state, Using application state, Using cookies and URL encoding. **Master Pages:** Creating master pages, content pages, nesting master pages, and accessing master page controls from a content page. **Navigation:** Site navigation: Using site navigation controls.	25	14
4	**Data Access through ADO.NET** Basic features of ADO.NET, Using SQL data sources, Grid view control, Detail view control, Form view control, List view and other controls Using object data sources.	20	10
5	**Advanced ASP.NET** AJAX extension, LINQ, Stored procedure, Working with XML data. ASP.NET MVC: Overview of Model-View-Controller, Model-View-Controller, and ASP.NET Routes	20	12
	Total	100	54

Reference Books

1.	**Professional ASP.Net** By Bill Evjen, Scott Hanselman, Devin Rader | Wrox Publication
2.	**ASP.NET 4 Unleashed** By Stephen Walther, Kevin Hoffman, Nate Dudek | Sams Publication
3.	**NET 4.0 Programming Black Book** Kogent Learning Solutions Incorporation, Dreamtech Press

Course Outcome

After Learning the Course the students shall be able to:
1. Explain .NET framework architecture and its features. 2. Build dynamic web applications to demonstrate event handling, state management and content manipulation using C#. 3. Demonstrate data driven applications using ADO.NET, LINQ and XML. 4. Develop web applications using model-view-controller pattern.

List of Practical

1.	Label and Button controls Create a web application to display "Welcome to the world of ASP.NET" on web browser when user "Click" button is clicked.
2.	Student Marksheet using web controls Create a web page in which user can enter enrollment number, student name and marks for 5 subjects. Calculate total, percentage and grade of the students and display it. [Use text box, label, button control].
3.	Textbox and Button controls Write a program to enable - disable textbox and change width of text box programmatically.
4.	Use of Radio button Create a web page that will take two radio buttons labeled as a gender, if user selects radio button the gender will be changed according to selected radio button.
5.	List box and Image controls Write a program which have list box and image controls, list box is used to display items and when user clicks on item in list box it display the item image in image control.
6.	Registration page Create a web page for registration which have name, dob, gender, email id, contact no, address, city, pin code, hobbies and display all data in same page with proper layout.
7.	Required Field validation Write a program that uses a textbox for a user input and validate it using required field validation.
8.	Registration page with validation Create a web page for registration which have name, dob, gender, email id, contact no, address, city, pin code, hobbies. After the user enters the appropriate values the validate button must validates the values entered and display all data in same page with proper layout.
9.	Custom Validation Write a program to check the length of the string in the text box using custom validation.
10.	Introduction to Master page Create a master page that contain links for home page, about us page and contact page, when user clicks on particular link it will display that particular page.
11.	Use of Master page and Session Create a web application for registration system that contains navigation control to specify link like registration and login. When user click registration link user should be navigated to registration page. When user click on the login link login page will be display that will redirected to registration page if user logged in to system. [Use master page and session to maintain the user identity].
12.	Cookie's Write a program that has a form taking the user's name as input. Store this name in a permanent cookie & whenever the page is opened again, then value of the name field should be attached with the cookie's content.
13.	Database CRUD operations [Gridview] Create student database which contain table stud_registration and have fields like first name, last name, dob, email id, contact number, gender, address, city, pin code. Create web application which will insert, view, search, update, and delete records from stud_registration table. [Use gridview]
14.	Database CRUD operations [Listview] Create product data base which contain product_info table and have fields like product id, product name, product quantity, price . Create a web application with validation which will insert, view, search, update and delete records from table. [Use listview]
15.	Database Operations Create table employee with the following columns and data types. Columns Data Type dept_id int Name nvarchar(20) DOJ datetime Salary float Designation nvarchar(30) ☐ Insert following records into the table ☐ Change the candidate name from "Ansh" to "Alpesh". ☐ Select records from table where salary > 10000 and display it.
16.	LINQ and Stored Procedure Create database named train_master with table named train_info and fields like train_id (PK, AI), train_name, train_type, arrival_time, departure_time, start_location, and end_location. Create web page, traindata.aspx, that allows user to enter train_id, train_name, train_type, arrival_time, departure_time, start_location, end_location and perform following operations ☐ Insert, delete, view, update, and search [Perform this application using LINQ and Stored Procedure]
17.	Ajax Create a Calendar Navigation by using Ajax Calendar Extender control ,in which user can Navigate Year , Month and select date from the Calendar control.
18.	XML Create an application which reads data from an XML file and also write data to the file.

19.	**Introduction to MVC Create database book_master having table book_info and have fields like book_id (PK, AI), book_name, author, price. Create a web application with validation which will insert, view, search, update and delete records from table. [Use MVC architecture]**
20.	**Mini Project Create a data-driven website system from following list:** 1. College Web Portal 2. E-Shopping 3. Online Student Management 4. Online Inventory 5. Web Time Sheet 6. Complaint Management 7. Alumni Information System 8. Property Management 9. Online Hotel Booking 10. Library Management 11. Hospital Management

Course: MBA **Semester:** 2

Prerequisite: Knowledge of Professional Skills - I

Rationale : This course develops advanced communication and employability skills for effective professional performance and career readiness.

Teaching and Examination Scheme

Teaching Scheme					Examination Scheme					Total
Lecture Hrs/Week	Tutorial Hrs/Week	Lab Hrs/Week	Seminar Hrs/Week	Credit	Internal Marks			Marks EYE		
					T	CE	P	T	P	
1	-	2	-	2	40	-	20	60	30	150

SEE - Semester End Examination, **T** - Theory, **P** - Practical

Course Content

W - Weightage (%) , **T** - Teaching hours

Sr.	Topics	W	T
1	**Picture Perception**Concept of Picture Perception and Interpretation Observation, Assumptions, and Logical Thinking Story Structure: Past–Present–Future	5	1
2	**LinkedIn Profiling and Resume Building – Difference between CV/Resume; Format, Heading and Application Procedure:**Difference between CV and Resume Resume Structure, Headings, and Content LinkedIn Profile Essentials, Key words and Professional Branding	15	2
3	**Writing Skills: Report, Memo & Proposal Writing**Purpose and Types of Reports, Memos, and Proposals Structure, Format, and Language Style Clarity, Objectivity, and Professional Tone	10	1
4	**Decision-Making & Problem Solving through Case Studies**Decision-Making Process and Models Problem Identification and Root Cause Analysis Ethical and Logical Decision-Making	5	1
5	**Reading Comprehension (Level of Difficulty - Advanced & Case Study based)**Advanced Reading Strategies (Inference, Evaluation) Understanding Case Study based Passages Critical Reading and Logical Reasoning	10	1
6	**Organizing Meetings - Agenda, Minutes of Meeting**Purpose and Types of Meetings Agenda Preparation and Meeting Roles Format and Language of Minutes of Meeting	10	1
7	**JAM Sessions**Concept and Purpose of JAM Content Structuring under Time Limit Fluency, Coherence, and Confidence	5	1
8	**Time Management and Stress Management**Importance of Time Management Prioritization Techniques (Urgent vs Important) Pomodoro technique and Eisenhower Matrix	10	1
9	**Group Discussions: Do's and Don'ts - Practice Sessions:**Purpose and Types of Group Discussions Roles, Etiquette, and Evaluation Criteria	10	2

	Common Mistakes in GDs		
10	**Interview Skills – STAR Technique- Mock interview sessions:**Types of Interviews STAR Technique for answering behavioural questions	**20**	**4**
	Total	**100**	**15**

Suggested Distribution Of Theory Marks Using Bloom's Taxonomy

Level	Remembrance	Understanding	Application	Analyze	Evaluate
Weightage	15	15	35	15	20

Reference Books

1.	**The 7 Habits of Highly Effective People** By Stephen R. Covey | Simon & Schuster, Pub. Year 2020
2.	**Basic Business Communication** By Lesikar, Petit &Flately, Lesikar's | Tata McGraw Hill
3.	**Basic Communication** By Poe & Fruchling | AITBS
4.	**Personality Development and Soft Skills** By Barun K Mitra | Oxford, 2011
5.	**1. Sanjay Kumar, Pushp Lata, Communication Skills, Oxford University Press**

Course Outcomes

At the end of this course Students Will be able to:	
1	Explain key formats, structures, and terminology related to professional documents and workplace communication.
2	Apply communication strategies, case analysis, and the STAR technique in group discussions, interviews, and professional situations.
3	Analyse professional documents, case studies, and workplace interactions to identify effectiveness and areas for improvement.
4	Assess personal and peer performance in mock interviews and documentation tasks, and group discussions using defined criteria.
5	Design and develop structured reports, memos, proposals, resumes, CVs, LinkedIn profiles, meeting agendas, and minutes of meetings demonstrating professional standards.

List of Practical	
1.	**Picture Perception** Picture-based Story Writing Group Discussion on Picture Interpretation
2.	**Writing Skills: Report, Memo & Proposal Writing** Writing a Short Report on a Given Case Drafting Internal Memos for Workplace Scenarios Preparing a Proposal
3.	**Decision-Making & Problem Solving through Case Studies** Case Study Analysis in Groups Problem–Solution Presentation Decision Justification Exercise
4.	**Reading Comprehension – Advanced & Case Based** Advanced RC Practice Sets Case study based Group Discussion on Case Conclusions
5.	**LinkedIn and Resume Building – Difference between CV/Resume; Format, Heading and Application Procedure** Resume Drafting Activity LinkedIn Profile Creation/Optimization
6.	**Organising Meetings - Agenda, Minutes of Meeting** Agenda Drafting Exercise Mock Meeting Role-play Writing Minutes of Meeting
7.	**JAM Sessions** Individual JAM Speaking Rounds Topic-based JAM Practice
8.	**Time Management and Stress Management** Time Log and Priority Planning Activity Stress Management Techniques Practice Group Discussion on Personal Challenges
9.	**Group Discussions: Do's and Don'ts - Practice Sessions** Mock Group Discussion Sessions Observation and Feedback Activity
10.	**Interview Skills – STAR Technique- Mock interview sessions** STAR-based Answer Framing Mock Interview Sessions Feedback and Improvement Discussion

Course: M.Sc. (IT)

Semester: 2

Prerequisite: Fundamental knowledge of computer network.

Rationale : The key objectives of this course are to develop an understanding of information assurance as practiced in computer operating systems, networks and representative applications and to gain familiarity with prevalent attacks, defenses against them.

Teaching and Examination Scheme

Teaching Scheme					Examination Scheme					Total
Lecture Hrs/Week	Tutorial Hrs/Week	Lab Hrs/Week	Seminar Hrs/Week	Credit	Internal Marks			External Marks		
					T	CE	P	T	P	
3	1	2	-	5	40	20	20	60	30	150

SEE - Semester End Examination, **T** - Theory, **P** - Practical

Course Content — **W** - Weightage (%) , **T** - Teaching hours

Sr.	Topics	W	T
1	**Cyber Security Foundations and Threat Landscape** Introduction to Cyber Security, Information Security principles, CIA Triad, Security goals and models, Defense-in-depth strategy, Types of cyber threats and attacks (Malware, Phishing, Ransomware, Social Engineering), Risk management concepts, Security policies, Security standards and frameworks (NIST, ISO 27001, CIS Controls). **Cryptography and Secure Communication** Fundamentals of cryptography, Symmetric encryption (AES, DES), Asymmetric encryption (RSA), Cryptographic hash functions (SHA-256, SHA-3), Digital signatures, Message authentication codes (MAC), Public Key Infrastructure (PKI), Introduction to AI Security, post-quantum security.	24	12
2	**Network Security and Monitoring** Network security architecture, Firewalls and access control lists, Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS), Virtual Private Networks (VPN), Network security protocols, Network traffic monitoring and packet analysis, Introduction to Security Information and Event Management (SIEM). **System and Application Security** Operating system security mechanisms, Linux and Windows security architecture, Privilege management and access control, Vulnerability assessment and patch management, Secure configuration and hardening.	20	10
3	**Cloud Security** Introduction to cloud computing and cloud security, Cloud service models (IaaS, PaaS, SaaS), Cloud deployment models (Public, Private, Hybrid), Shared responsibility model, Identity and Access Management (IAM), Multi-Factor Authentication (MFA), Cloud security threats (data breaches, misconfigured storage, insecure APIs, account hijacking, insider threats, DDoS attacks), Data protection techniques including encryption and key management, Backup and disaster recovery, Cloud security monitoring and logging, Introduction to cloud security tools such as AWS Security Hub, AWS CloudTrail, AWS GuardDuty, Azure Security Center, and Google Cloud Security Command Center. **Mobile Security** Introduction to mobile security concepts, Mobile operating system architecture, Android security model, Mobile application vulnerabilities and threats, Malware and spyware in mobile devices, Mobile authentication mechanisms, Secure mobile communication, Mobile device management (MDM), Mobile application security	32	16

	testing and analysis. **Virtualization and Container Security** Introduction to virtualization concepts, Types of virtualization, Virtual machines and hypervisors, Security issues in virtualized environments, Containerization concepts and container security, Security challenges in Docker and Kubernetes environments, Best practices for securing virtualized infrastructures.		
4	**Digital Forensics and Incident Response** Introduction to digital forensics, Types of digital evidence, Rules of evidence and chain of custody, Digital forensic investigation process, Disk forensics, Memory forensics, Network forensics, Log analysis, Incident detection and response process, Documentation and reporting of forensic findings.	**12**	**5**
5	**Advanced Digital Forensics and Investigation Techniques** Forensic acquisition and imaging techniques, File system forensics, Email forensics, Malware forensics, Mobile device forensics, Cloud forensics concepts, Timeline analysis, Evidence preservation and integrity verification, Use of digital forensic tools such as Autopsy, FTK Imager, Volatility Framework, and Wireshark for forensic investigation.	**12**	**5**
	Total	**100**	**48**

Reference Books

1.	**Information systems security (TextBook)** By Nina Godbole | Wiley Publications,2008
2.	**Cyber Security understanding Cyber Crimes, Computer forensics and Legal Perspectives (TextBook)** By Nina Godbole and Sunit Belapure
3.	**Cryptography and Network Security Principles and Practices** By W. Stallings | Prentice-Hall of India, 2006 | 4th Edition
4.	**Information Security: Principles and Practices** By M. Merkow and J. Breithaupt | Pearson Education, 2006

Course Outcome

After Learning the Course the students shall be able to:

1. **Understand and analyze cybersecurity principles, threat landscapes, and security frameworks** used to protect modern information systems.
2. **Apply cryptographic techniques and network security mechanisms** to secure communication, detect attacks, and protect network infrastructure.
3. **Evaluate and implement security practices for systems, cloud platforms, mobile devices, and virtualized environments** to mitigate vulnerabilities and cyber threats.
4. **Conduct digital forensic investigations and incident response activities** by analyzing digital evidence using forensic tools and methodologies.

List of Practical

1.	**Cyber Security Lab Setup** **Installation and configuration of virtual lab environment using VirtualBox / VMware, installation of Kali Linux, setup of vulnerable machines such as Metasploitable and OWASP BWA, and understanding the role of Kali Linux in cybersecurity and forensic analysis.**
2.	**Network Scanning and Enumeration** **Perform host discovery, port scanning, and service enumeration using Nmap, Netdiscover, and Recon-ng.**
3.	**Packet Capture and Network Traffic Analysis** **Capture and analyze network packets to understand network communication and detect suspicious traffic using Wireshark and Tcpdump.**
4.	**Vulnerability Assessment and Scanning** **Identify system and web vulnerabilities using OpenVAS, Nessus, and Nikto.**

5.	**Web Application Security Testing** **Perform web application penetration testing to detect vulnerabilities such as SQL Injection and Cross-Site Scripting using Burp Suite, OWASP ZAP, and DVWA (Damn Vulnerable Web Application).**
6.	**Password Cracking and Authentication Testing with Mitigation** **Perform password attacks such as dictionary and brute-force using John the Ripper, Hydra, and Hashcat, and analyze prevention techniques such as strong password policies and hashing.**
7.	**Penetration Testing using Metasploit Framework** **Exploit vulnerabilities in a controlled lab environment using Metasploit Framework and analyze post-exploitation techniques.**
8.	**Wireless Network Security Testing** **Analyze wireless network security and perform wireless scanning using Aircrack-ng, Airodump-ng, and Kismet.**
9.	**Cloud Security Monitoring using Free AWS Tools** **Configure and analyze cloud security features using AWS IAM, AWS CloudTrail, and AWS GuardDuty (free tier/trial), including access control and logging.**
10.	**Mobile Application Security Analysis** **Analyze Android applications and identify vulnerabilities using MobSF (Mobile Security Framework), APKTool, JADX, and ADB.**
11.	**Malware Analysis Basics** **Perform static analysis of suspicious files using PEStudio, Strings, and VirusTotal.**
12.	**Digital Forensics Investigation** **Acquire and analyze disk images for forensic investigation using Autopsy, FTK Imager, and The Sleuth Kit.**
13.	**Memory Forensics Analysis** **Analyze memory dumps and detect malicious processes using Volatility Framework.**
14.	**Security Log Analysis and Incident Investigation** **Analyze system and network logs using tools like Splunk or ELK Stack to detect suspicious activities and support incident investigation. (Elasticsearch, Logstash, Kibana).**

Course: M.Sc. (IT)

Semester: 2

Prerequisite: Knowledge of data structures and discrete mathematics.

Proposed Rationale: To provide fundamental knowledge of Artificial Intelligence including problem solving, knowledge representation, and reasoning techniques. To introduce data handling and basic machine learning concepts for real-world applications.

Teaching and Examination Scheme

Teaching Scheme					Examination Scheme					Total
Lecture Hrs/Week	Tutorial Hrs/Week	Lab Hrs/Week	Seminar Hrs/Week	Credit	Internal Marks			External Marks		
					T	CE	P	T	P	
3	1	2	-	5	40	20	20	60	30	150

SEE - Semester End Examination, **T** - Theory, **P** - Practical

Sr.	Topics	W	T
1	**Introduction To AI & Real-World Applications:** History & evolution of AI, Types: ANI, AGI, ASI, AI applications (healthcare, education, business), Turing Test, Introduction to modern AI (Generative AI overview)	**10**	**6**
2	**Intelligent Agents & Environments:** Agents & environment, Types of agents: Simple reflex, Goal-based, Utility-based Learning agents, Rationality & performance measures How AI Agent works	**10**	**6**
3	**Problem Solving & Search Techniques:** Problem formulation & state space, Production systems Informed search (Heuristics, A*, Hill Climbing, Generate and Test, Means Ends Analysis, Constraints Satisfaction Problem) Game playing: Minimax, Alpha-Beta pruning	**20**	**9**
4	**Knowledge Representation & Logic** Knowledge representation: Introduction, Types of Knowledge, Approaches Propositional logic Predicate logic Inference: Forward chaining, Backward chaining	**15**	**6**
5	**Reasoning Under Uncertainty** Probability basics, Bayes theorem, Bayesian networks, Fuzzy logic	**15**	**5**
6	**Data Handling & Feature Engineering** Types of data (structured/unstructured), Data preprocessing (cleaning, normalization), Feature selection & extraction, Handling imbalanced data	**15**	**6**
7	**Introduction To Machine Learning** Learning from data Types: Supervised, Unsupervised, Reinforcement learning Models: Linear regression, Logistic regression, SVM	**15**	**7**
	TOTAL	**100**	**45**

Course Outcomes
After Learning the Course, the students shall be able to:
1. Explain fundamental concepts of Artificial Intelligence, including its types and applications. 2. Analyse intelligent agents and their interaction with different environments. 3. Apply problem-solving and search techniques to AI-based problems. 4. Demonstrate knowledge representation and logical reasoning using appropriate methods. 5. Evaluate uncertainty models and apply basic machine learning techniques with data handling.

List of Practical	
1.	Implement a simple reflex agent for a vacuum cleaner environment using condition–action rules.
2.	Modify the above program to implement a goal-based agent.
3.	Implement a Tic-Tac-Toe game using AI concepts.
4.	Develop a rule-based system for simple medical diagnosis.
5.	Write a Python program to solve the Water Jug Problem.
6.	Perform data cleaning and preprocessing on datasets by handling missing values, normalization, and feature transformation.
7.	Perform Linear Regression for modeling and predicting continuous variables. *Sample Project:* House Price Prediction using Linear Regression OR ANY
8.	Perform binary classification using Logistic Regression on structured data. *Sample Project:* Email Spam Detection OR ANY
9.	Perform classification using Support Vector Machine (SVM) on a dataset and evaluate its performance. *Sample Project:* Breast Cancer Detection using SVM OR ANY
10.	Case Study: Analyze a real-world AI application (e.g., healthcare diagnosis system) and explain how agents, search, or ML techniques are used.

Course: M.Sc. (IT) **Semester:** 2

Prerequisite: Basic Knowledge of Data Analytics and Computation Methods

Rationale : The objective of this course is to provide Conceptual insight about Big Data Analytics, Installation and understanding of Hadoop Architecture and its ecosystems.

Teaching and Examination Scheme

Teaching Scheme					Examination Scheme					Total
Lecture Hrs/Week	Tutorial Hrs/Week	Lab Hrs/Week	Seminar Hrs/Week	Credit	Internal Marks			External Marks		
					T	CE	P	T	P	
3	1	2	0	5	20	20	20	60	30	150

SEE - Semester End Examination, **T** - Theory, **P** - Practical

Course Content

W - Weightage (%) , **T** - Teaching hours

Sr.	Topics	W	T
1	**Overview of Big Data**Introduction to Big Data, Evolution of Big Data, Structuring of Big Data, Fundamentals of Big Data, Big Data Analytics, Career and Future in Big Data	10	5
2	**Discovering the Use of Big Data in Business Context**Big Data in Social Networking, Big Data in Preventing Fraudulent Activities, Big Data in Detecting Fraudulent Activities in Insurance Sector, Use of Big Data in Retail Industry.	10	5
3	**Technologies for Handling Big Data**Distributed and Parallel Computing for Big Data, Introducing Hadoop, Cloud Computing and Big Data, In-Memory Computing Technology for Big Data.	8	4
4	**Understanding Hadoop Ecosystem**History of Hadoop, Hadoop Ecosystem, Analysing Data with Unix tools, Analysing Data with Hadoop, Hadoop Streaming, Hadoop Echo System, IBM Big Data Strategy, Introduction to Infosphere BigInsights and Big Sheets.	12	6
5	**HDFS(Hadoop Distributed File System)**The Design of HDFS, HDFS Concepts, Command Line Interface, Hadoop file system interfaces, Data flow, Data Ingest with Flume and Scoop and Hadoop archives, Hadoop I/O: Compression, Serialization, Avro and File-Based Data structures.	14	6
6	**MapReduce Fundamentals**Anatomy of a Map Reduce Job Run, Failures, Job Scheduling, Shuffle and Sort, Task Execution, Map Reduce Types and Formats, Map Reduce Features.	14	6
7	**Understanding Big Data Technology Foundations, Storing Data in Databases and Data Warehouses**Understanding Big Data Technology Foundations: Exploring the Big Data Stack, Virtualization and Big Data, Virtualization Approaches, Summary, Quick Revise Storing Data in Databases and Data Warehouses: RDBMS and Big Data, Non-Relational Database, Polyglot Persistence, Integrating Big Data with Traditional Data Warehouses, Big Data Analysis and Data Warehouse, Changing Deployment Models in Big Data Era.	16	8
8	**Processing Your Data with MapReduce**Recollecting the Concept of MapReduce Framework,Developing Simple MapReduce Application, Points to Consider while Designing MapReduce.	16	8
	Total	100	48

Reference Books	
1.	**Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's Businesses** By Michael Minelli, Michelle Chambers, and Ambiga Dhiraj | Wiley, Pub. Year 2013
2.	**Big-Data Black Book** By DT Editorial Services | Wiley India
3.	**Hadoop Operations, Eric Sammer, O'Reilley.** By O'Reilley
4.	**Hadoop: The Definitive Guide by Tom White, Third Edition, O'Reilley.** By Tom White

Course Outcome
After Learning the Course the students shall be able to:
1. Work with big data platform and explore the big data analytics techniques business applications. 2. Understand the fundamentals of various big data analytics techniques. 3. Analyse the HADOOP and Map Reduce technologies associated with big data analytics. 4. Have knowledge on accessing, storing and manipulating the huge data from different resources.

List of Practical	
1.	**(i)Perform setting up and Installing Hadoop in its two operating modes: ☐ Pseudo distributed, ☐ Fully distributed. (ii) Use web based tools to monitor your Hadoop setup.**
2.	**(i) Implement the following file management tasks in Hadoop: ☐ Adding files and directories ☐ Retrieving files ☐ Deleting files ii) Benchmark and stress test an Apache Hadoop cluster**
3.	**Run a basic Word Count Map Reduce program to understand Map Reduce Paradigm. ☐ Find the number of occurrence of each word appearing in the input file(s) ☐ Performing a Map Reduce Job for word search count (look for specific keywords in a file)**
4.	**To understand the overall programming architecture using Map Reduce API**
5.	**Stop word elimination problem: ☐ Input: A large textual file containing one sentence per line A small file containing a set of stop words (One stop word per line) ☐ Output: A textual file containing the same sentences of the large input file without the words appearing in the small file.**
6.	**Write a Map Reduce program that mines weather data. Weather sensors collecting data every hour at many locations across the globe gather large volume of log data, which is a good candidate for analysis with MapReduce, since it is semi structured and recordoriented. Data available at: https://github.com/tomwhite/hadoopbook/tree/master/input/ncdc/all. ☐ Find average, max and min temperature for each year in NCDC data set? ☐ Filter the readings of a set based on value of the measurement, Output t**
7.	**Store the basic information about students such as roll no, name, date of birth , and address of student using various collection types such as List, Set and Map**
8.	**Basic CRUD operations in MongoDB**
9.	**Retrieve various types of documents from students collection**
10.	**To find documents from Students collection**
11.	**Develop Map Reduce Work Application**
12.	**Creating the HDFS tables and loading them in Hive and learn joining of tables in Hive**

Course: M.Sc. (IT) **Semester:** 2

Prerequisite: basic knowledge of HTML, CSS, and JavaScript fundamentals along with familiarity in working with any programming environment or web development tools.

Rationale : This course equips students with advanced JavaScript, modern UI frameworks, ReactJS, API integration, and MongoDB skills needed to build full-stack, industry-ready web applications. It enables learners to develop efficient, responsive, and scalable solutions aligned with current trends in modern web development.

Teaching and Examination Scheme

Teaching Scheme					Examination Scheme					Total
					Internal Marks			External Marks		
Lecture Hrs/Week	Tutorial Hrs/Week	Lab Hrs/Week	Seminar Hrs/Week	Credit	T	CE	P	T	P	
3	1	2	-	5	20	20	20	60	30	150

SEE - Semester End Examination, **T** - Theory, **P** - Practical

Course Content

W - Weightage (%) , **T** - Teaching hours

Sr.	Topics	W	T
1	**Advanced JavaScript Concepts:**Deep Dive into JavaScript Fundamentals (Scope, Hoisting, and Closures, Prototypes and Prototypal Inheritance, **this** Keyword and its Context Execution Context and Call Stack) , Asynchronous JavaScript (Promises and Async/Await) JavaScript Design Patterns (Singleton, Factory, Observer, and Strategy Patterns Module Pattern and Revealing Module Pattern Functional Programming), Memory Management, Functional Programming	15	7
2	**JavaScript Objects & Arrays:**ES6 Features (let/const, Arrow Functions, Template Literals, Destructuring, Spread & Rest Operators), Promises & Async/Await, Fetch API, JavaScript Modules (Import/Export), Error Handling & Debugging, Regular Expressions, Higher-Order Functions (map, filter, reduce), Event Loop & Callbacks, Local Storage & Session Storage, Object-Oriented JavaScript (Classes, Prototypes, Inheritance)	15	7
3	**Advanced UI Design**Material UI, Tailwind CSS, Bootstrap concepts	15	10
4	**ReactJS**Introduction to React, Setting Up a React Environment, React Components, JSX Syntax, Props and State, Handling Events, Conditional Rendering, Lists and Keys, Forms and Controlled Components, Advanced Component Patterns, React Hooks, Performance Optimization, Context API and State Management, Server-Side rendering and Static site generation, Testing and Best Practice	15	7
5	**API Calling & Integration**RESTful API Basics, Fetch API and Axios, Handling HTTP Methods (GET, POST, PUT, DELETE), Authentication and Authorization (JWT, OAuth), Error Handling in API Calls	15	7
6	**Database Connectivity with MongoDB**Introduction to NoSQL and MongoDB, CRUD Operations in MongoDB, Mongoose ODM and Schema Design, Aggregation Framework, Connecting Node.js with MongoDB	20	10
	Total	95	48

Reference Books	
1.	**You Don’t Know JS" (Series)** By Kyle Simpson
2.	**JavaScript: The Good Parts** By Douglas Crockford
3.	**Eloquent JavaScript** By Marijn Haverbeke
4.	**Tailwind CSS: From Zero to Production** By Simon Vrachliotis
5.	**The Road to React** By Robin Wieruch
6.	**Fullstack React** By Anthony Accomazzo
7.	**REST API Design Handbook** By Mike Stowe
8.	**API Design Patterns** By JJ Geewax

Course Outcome
After Learning the Course the students shall be able to:
1: Understand advanced JavaScript concepts including ES6 features, asynchronous programming, and functional programming. 2: Apply modern UI frameworks like Material UI, Tailwind CSS, and Bootstrap to design responsive interfaces. 3: Develop dynamic web applications using ReactJS components, hooks, state management, and best practices. 4: Integrate RESTful APIs and connect applications with MongoDB using CRUD operations and Mongoose

List of Practical	
1.	**Create a function with nested functions demonstrating closures, lexical scope, and use of this in different contexts.**
2.	**Simulate API calls using Promises and rewrite using async/await with proper error handling.**
3.	**Use let, const, arrow functions, destructuring, spread/rest, and template literals in a mini calculation or info display program.**
4.	**Create a task list using map, filter, reduce, and store/retrieve it from local storage.**
5.	**Build a responsive profile or landing page using Tailwind CSS utility classes.**
6.	**Design a product grid using Bootstrap cards with responsive breakpoints.**
7.	**Build a React app using Context API for theme switching and create a custom hook for toggling dark/light mode.**
8.	**Fetch user data from a public API using Axios and display it with proper loading and error messages.**
9.	**Create a simple Express server connected to MongoDB for adding, viewing, and deleting students.**
10.	**Define a Mongoose schema for products and perform an aggregation to group or filter products based on category or price.**

Course: M.Sc. (IT) **Semester:** 2

Prerequisite: Fundamental knowledge of Networking and Computing, Basic Knowledge of Web Services.

Rationale : The key objectives of this course are to provide an understanding of the basic concepts of parallel and distributed computing and their role in cloud computing, to study the concept of virtualization and relevant technologies available in the market, to understand the importance of cloud computing for higher throughput, to make aware about availability of various cloud platforms, to study different application of cloud and cloud management techniques.

Teaching and Examination Scheme

Teaching Scheme					Examination Scheme					Total
					Internal Marks			External Marks		
Lecture Hrs/Week	Tutorial Hrs/Week	Lab Hrs/Week	Seminar Hrs/Week	Credit	T	CE	P	T	P	
3	1	2	-	5	20	20	20	60	30	150

SEE - Semester End Examination, **T** - Theory, **P** - Practical

Course Content

W - Weightage (%) , **T** - Teaching hours

Sr.	Topics	W	T
1	**Introduction to Cloud Computing**Overview of Cloud Computing & Evolution,Cloud Service Models (IaaS, PaaS, SaaS), Cloud Deployment Models (Public, Private, Hybrid, Community), Benefits & Challenges of Cloud Computing,Major Cloud Providers (AWS, GCP, Azure)	7	15
2	**Cloud Computing Models**Cloud Architecture & Components, Multi-Tenancy & Resource Sharing, Load Balancing & Scalability in Cloud, Security & Compliance in Cloud Computing, Pricing Models & Cost Management	7	15
3	**Virtualization**Introduction to Virtualization & Hypervisors, Types of Virtualizations (Server, Network, Storage, Desktop), Virtual Machines vs. Containers, Resource Allocation & Optimization in Virtualization, Performance & Security Considerations in Virtualized Environments	10	20
4	**Cloud Adoption Frameworks**Understanding Cloud Migration Strategies, Cloud Governance & Best Practices DevOps & Cloud Adoption, Compliance & Risk Management in Cloud, Case Studies on Cloud Adoption	7	15
5	**Docker Concepts**Introduction to Docker & Containerization, Docker Architecture & Components, Creating & Managing Docker Images, Networking & Orchestration in Docker, Deploying Applications using Docker	7	15
6	**Google Cloud Platform (GCP)**Introduction to GCP and CORE services, Setting up Virtual Machines on GCP, Serverless Computing with Cloud Function, Running Containers on CloudRun, Security and Cost Optimization in GCP, Hosting on Cloud	10	20
	Total	48	100

Reference Books

1.	**Cloud Computing - A practical approach for learning and implementation** By A.Srinivasan and J.Suresh | Pearson Publications
2.	**Cloud Computing: A practical approach** By Anthony T. Vetle | TMH
3.	**Cloud Computing for Dummies** By Judith Hurwitz, R.Bloor, M.Kanfman, F.Halper | Wiley India Edition
4.	**Cloud Computing: SaaS, PaaS, IaaS, Virtualization, Business Models, Mobile, Security and More** By Kris Jamsa | Jones & Bartlett Learning
5.	**Cloud Computing Bible (TextBook)** By Barrie Sosinsky | Wiley India Pvt Ltd

Course Outcome

After Learning the Course the students shall be able to:
1: Define cloud computing, its applications, and explain the role of virtualization in building cloud environments. 2: Understand and compare the core principles, architectures, and service models (IaaS, PaaS, SaaS) of cloud computing. 3: Familiarize with and evaluate various cloud computing platforms to identify suitable solutions for different scenarios. 4: Demonstrate the ability to manage cloud storage, ensure data security, and deploy services and resources on cloud platforms effectively.

List of Practical

1.	**Creating a Google Cloud Account – Setting up & securing an account**
2.	**Creating a VM (Linux) Environment – Deploying & managing VMs on GCP**
3.	**Introduction to Docker – Understanding Docker architecture & installation**
4.	**Creating an Image using Docker – Writing Dockerfiles & building images**
5.	**Running a Docker Container – Managing containers & executing commands**
6.	**Docker Networks – Creating and configuring Docker networks**
7.	**Running Docker on Cloud or CloudRun – Deploying containers in a cloud environment**
8.	**Cloud Function-Based Practical – Implementing serverless computing**