w. e. f. Academic Year:	2025-26
Semester:	4
Category of the Course:	**Core Courses**

Prerequisite:	Basic programming knowledge recommended; familiarity with mathematics or logic will be helpful but not mandatory.
Rationale:	The rapid growth of digital technologies demands professionals who are skilled in foundational computing, programming, and modern IT practices. These courses are designed to provide learners with essential knowledge and hands-on experience in web development, software engineering, version control, databases, artificial intelligence, and cloud computing. By combining theory with practical application, the curriculum ensures that students not only understand core concepts but also develop industry-relevant skills, preparing them for advanced studies, professional certifications, and real-world project implementation.

Course Outcome:

After Completion of the Course, Student will able to:

No	Course Outcomes	RBT Level
01	By the end of this course, learners will be able to build and style simple web pages using HTML and CSS, apply JavaScript fundamentals to add interactivity, and demonstrate their skills by creating a functional rock-paper-scissors game that integrates structure, design, and logic.	**A**
02	By completing this course, learners will understand the core principles of software engineering, gain hands-on experience with version control, and confidently use GitHub to create and manage branches for effective team collaboration and project development.	**N**
03	By the end of this course, learners will be able to use Git and GitHub for version control and collaboration, understand and apply the fundamentals of SQL with relational databases, and explore the basics of NoSQL and Database-as-a-Service (DBaaS) to manage and work with diverse data storage solutions.	**C**
04	By completing this course, learners will gain a foundational understanding of AI concepts and machine learning, explore practical applications across domains including biomedical fields, apply mathematical optimization techniques to solve business problems, and develop awareness of responsible and ethical considerations in AI development and deployment.	**C**
05	By completing this course, learners will understand the fundamentals of cloud computing, gain practical knowledge of IBM Cloud essentials, and acquire hands-on skills to build and deploy their first server using server-side JavaScript.	**C**

**Revised Bloom's Taxonomy (RBT)*

Teaching and Examination Scheme:

Teaching Scheme (in Hours/week)			Total Credits L+T+ (PR/2)	Assessment Pattern and Marks				Total Marks
				Theory		Tutorial / Practical		
L	T	PR	C	ESE (E)	PA / CA (M)	PA/CA (I)	ESE (V)	
5	0	0	5	100	0	0	0	100
*Total Lecture Hrs. (L) =75			Total Practical Hrs. (PR) =00			Total Hours =75 Hrs		

Course Content:

Unit No.	Content	No. of Hours	% of Weightage
1.	**Foundations of Web Development with HTML, CSS & JavaScript – Introduction to Web Development**, **HTML Basics**-Tags, elements, attributes, Structuring web pages, Forms, lists, tables, media elements **CSS Fundamentals**-Selectors, properties, colors, typography, Box model, margins, padding, borders, Layout techniques: Flex box (intro). Applying Design Principles, Visual hierarchy, spacing, alignment, Responsive design basics JavaScript Basics-Variables, data types, operators, Conditionals, loops, events, Updating HTML content dynamically, Hands-on Implementations, Creating and styling a complete web page, Building a simple Rock–Paper–Scissors game using JS logic, user interaction, and DOM updates.	15	20
2.	**Foundations of Software Engineering and Version Control** – Overview of Software Development Life Cycle (SDLC) (Requirement → Design → Development → Testing → Deployment) **Basics of Software Engineering**-Code readability: naming, comments, indentation, Introduction to debugging using browser Dev Tools, Introduction to Version Control, Need for version control, Repository, ommit, push, pull concepts, Basic Git Commands, git in it, Git add, Git commit, Git t status, Git log, Working with GitHub, Creating a repository, Uploading files, Viewing commit history	15	20

3.	**JavaScript Fundamentals:** Variables (let, const), data types,Operators & expressions,,Conditionals (if/else, switch),Loops (for, while), Arrays & Objects (Introductory level), Functions, Function declaration, parameters, return,,Basic arrow function concepts, DOM Manipulation (Beginner level),Selecting elements (getElement ById, querySelector),Changing text, styles, attributes, Handling events (click, input), Simple Interactive ,omponents, Toggle button, Basic form validation, Simple JS calculator.	**15**	**20**
4	**Introduction to Artificial Intelligence-**Definition, scope, and evolution of AI,Key AI domains and terminology, Foundations of AI,Problem-solving and search techniques (introductory),Knowledge representation basics **Machine Learning Basics-**Types of ML: supervised, unsupervised, reinforcement (overview),Classification vs. regression,Model training and evaluation concepts **AI Applications-**Business optimization use cases,Industry examples across retail, finance, manufacturing, AI in Biomedical Applications, Diagnostic support systems (intro),Image-based analysis (basic concepts),Hands-on workshop on simple biomedical AI workflows, Ethical and Responsible AI,Bias, transparency, fairness, Data privacy considerations, Responsible use of AI technologies.		
5.	**Cloud Computing Fundamentals** – Introduction to Cloud Computing,Definition, characteristics, benefits, Public, private, hybrid cloud models, Cloud Service Models,,IaaS, PaaS, SaaS (overview and examples) **IBM Cloud Essentials-**IBM Cloud console overview,Key services: compute, storage, networking (introductory),Resource creation and management basics, Cloud Deployment Concepts,Regions, zones, resource groups,Basics of scaling and availability **Introduction to Server-Side Development-**Basics of JavaScript for backend,Creating a simple server using Node.js, Deployment Workshop,Deploying a basic JavaScript server on IBM Cloud,Testing and validating the deployed application.	**15**	**20**
	Total	**75**	**100**

Suggested Specification Table with Marks :

Distribution of Marks (in %)					
R Level	**U Level**	**A Level**	**N Level**	**E Level**	**C Level**
5	**15**	**25**	**20**	**10**	**25**

Where R: Remember; U: Understanding; A: Application, N: Analyze and E: Evaluate C: Create (as per Revised Bloom's Taxonomy)

References/Suggested Learning Resources:

(a) Books:

1. An Integrated Approach to Software Engineering – Pankaj Jalote
2. Introduction to Machine Learning – Ethem Alpaydin
3. Full Stack Development with Spring Boot and React – Juha Hinkula
4. Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow – Aurélien Géron

(b) Open-source software and website:

1. **Python.org** – https://www.python.org/
2. **Docker** – https://www.docker.com/
3. **Kubernetes** – https://kubernetes.io/
4. **React.js Official** – https://react.dev/
5. **Node.js** – https://nodejs.org/
6. **Flask** – https://flask.palletsprojects.com/

Suggested Course Practical List: List of Laboratory/Learning Resources Required: IDE (IntelliJ/Eclipse/VS Code), JDK, Tomcat/Jetty, MySQL/PostgreSQL, Spring Boot, Hibernate, Maven/Gradle, GitHub, Postman, Browser DevTools, Docker.

List of Laboratory/Learning Resources Required: IntelliJ/Eclipse/VS Code, JDK 17/21, Tomcat/Jetty, MySQL/PostgreSQL (Workbench/pgAdmin), Spring Boot, Hibernate, Maven/Gradle, GitHub/GitLab, Postman/Swagger, Browser DevTools, Docker.

Suggested Project List: E-commerce site, Student/Job Portal, Blogging CMS, Online Banking, Food Delivery App, Social Media Platform, Hospital Management System.

Suggested Activities for Students: Coding challenges, hackathons, code reviews, open-source contributions, project demos, debugging practice, resume/GitHub building, cloud deployment.

Skill & Practical Activities to be carried out during Semester						
Sr. No.	**Category of Engagement**	**Describe the activities to be carried out by students in brief**	**Expected Frequency & Duration**	**Mode of Delivery (Online / Offline / Hybrid)**	**Tools / Platforms / Equipment / Machinery to be Used**	**Expected major Learning Outcomes**
1	Master Classes / Expert Lectures by Industry Professionals	Guest lectures on *Modern Web Frameworks (Spring Boot, ReactJS), Microservices Architecture*, and *Industry Best Practices in Java Full Stack Development*.	2–3 sessions (2 hours	Online	Zoom/MS Teams, IntelliJ IDEA, Spring Boot Demo Apps	• Gain industry insights into Java-based full stack development trends. • Understand deployment and scalability practices from experts.
2	Quizzes / Competency-Based Evaluation	Periodic quizzes and coding challenges on *Core Java, JDBC, Spring Boot, REST APIs, ReactJS* and *SQL/NoSQL Databases*.	Every 2 weeks	Online	IBM LMS	• Assess conceptual understanding and coding proficiency. • Strengthen analytical and debugging skills through practice.
3	Hands-on Training / Lab Exercises / Tool-Based Learning	Lab sessions covering *frontend design using HTML, CSS, ReactJS, backend with Spring Boot, database integration using MySQL/MongoDB*, and *project deployment using GitHub & Docker*.	Weekly (3 hours per session)	Online	IBM LMS	Build and deploy Java-based full stack applications. • Gain practical experience with modern frameworks, databases, and version control. • Develop problem-solving skills through end-to-end project implementation.