

**Syllabi and Scheme of Examinations
for
Skill Enhancement Courses for Under Graduate Programs
(Single Major / Multidisciplinary Programs)**

(Based on Curriculum and Credit Framework for UG Programs under NEP)



With Effect from the Session 2024-25

**MAHARSHI DAYANAND UNIVERSITY
ROHTAK (HARYANA)**

Syllabi and Scheme of Examinations for Skill Enhancement Courses for Under Graduate (Single Major/Multidisciplinary/ Single Major after 2nd Semester of Multidisciplinary Programs)of Computer Science w.e.f. 2024-25

Skill Enhancement Course (SEC)	Nomenclature of Course	Course Code	Credits Distribution			Total Credits	Workload			Total Workload	Marks			Total Marks
			L	T	P		L	T	P		Theory		Practical	
						External				Internal				
SEMESTER 1 (2024-25)														
SEC 1 @ 3 credits	Web Development-I	24CSC401SE01	1	0	2	3	1	0	4	5	20	05	50	75
SEMESTER II (2024-25)														
SEC 2 @ 3 credits	Web Development-II	24CSC402SE01	1	0	2	3	1	0	4	5	20	05	50	75
SEMESTER III (2025-26)														
SEC 3 @ 3 credits	Java Programming	25CSC403SE01	1	0	2	3	1	0	4	5	20	05	50	75
SEMESTER VI (2026-27)														
SEC 4 @ 2 credits (offered only in case of Single Major Program)	Machine Learning and Python Programming	26CSC406SE01	0	0	2	2	0	0	4	4	-	-	50	50
SEMESTER VII1 (2027-28)														
SEC 5 @ 4 credits (if offered as an option)	MERN Stack Development	27CSC407SE01	2	0	2	4	2	0	4	6	35	15	50	100
SEMESTER VIII1 (2027-28)														
SEC 6 @ 4 credits (if offered as an option)	Advanced Machine Learning and Generative AI (Option-1)	27CSC408SE01	2	0	2	4	2	0	4	6	35	15	50	100
	OR													
	Social Media Analytics	27CSC408SE01	3	0	1	4	3	0	2	5	50	25	25	100

L: Lecture; T: Tutorial; P: Practical

Syllabi for Skill Enhancement Courses for Under Graduate Programs (Single Major / Multidisciplinary Programs) of Computer Science

Session: 2024-25

First Semester

Name of the Program	Skill Enhancement Courses (Single Major / Multidisciplinary Programs)	Program Code	----
Name of the Course	Web Development-I	Course Code	24CSC401SE01
Hours/Week	1+0+4	Credits(L:T:P)	1:0:2
Max. Marks	Theory: 25 (20+5) Practical: 50	Time of end term Examination	3 Hours
Note: The examiner has to set nine questions in all by setting two questions from each Unit and Question No. 1 consisting of short-answer type questions covering the entire syllabus. Student will be required to attempt five questions in all by selecting one question from each Unit and Question No. 1, which is compulsory.			
Course Objective: In this course, students will learn fundamental aspects of web server, browser and web development. The course will also provide foundational programming skills related to web development. Basic programming skills including HTML, CSS and Java Script language useful for designing the websites will also be covered. By the end of the course, students will gain practical skills and knowledge applicable to real-world scenarios, equipping them to tackle a variety of programming challenges related to website development.			
Course Outcomes: By the end of the course the students will be able to: CO1: Gain knowledge about various aspects related to web development CO2: Understand server side and client side scripting CO3: Develop proficiency in writing codes for web development using HTML and DHTML. CO4: Demonstrate an understanding of various interactive tools useful in developing interactive sites CO5: Learn different style layouts to be applied during web sites development.			
Unit – I			
Introduction to Internet: Overview of Internet, World Wide Web, Evolution and History of WWW; Basic Features; Evolution of Web development. Web Browsers: Web Servers; Hypertext Transfer Protocol; URLs; IP Addresses; Domain Names; Searching and Web- Casting Techniques; Search Engines and Search Tools; Internet Security; The Web Programmers; Toolbox. Web Technologies: Introduction Web Technologies; Introduction to HTML, CSS, and JavaScript; Client-Side vs. Server- Side Scripting.			
Unit– II			
Web Publishing: Hosting your Site; Internet Service Provider; Planning and designing your Web Site; Steps for developing your Site; Choosing the contents; Home Page; Domain Names; Creating a Website and the Markup Languages (HTML, DHTML). Web Development: Introduction to HTML; Hypertext and HTML; HTML Document Features; HTML command Tags; Creating Links; Headers; Text styles; Text Structuring; Text colors and Background; Formatting text; Page layouts; Lists, Tables; meta element; New HTML5 Form input Types; input and data list elements; auto complete Attribute; Page-Structure Elements; Introduction to DHTML and its features.			
Unit–III			

Brief Introduction to Interactivity tools: CGI; Features of Java; Java Script; Features of ASP; VBScript; Macromedia Flash; Macromedia Dreamweaver; JavaScript: The JavaScript execution environment; The Document Object Model ; Element access in JavaScript; Events and event handling ; Handling events from the Body elements, Button elements, Text box, and Password elements ; The DOM 2 event model ; The navigator object ; DOM tree traversal and modification;
Unit– IV
Introduction to CSS: Introduction to CSS, Block and Inline Elements, Inline Styles, using internal CSS, using external CSS, How CSS rules cascade, inheritance, external style sheets.CSS3 Basics: CSS selectors, color: foreground color, background color, contrast, opacity; text: Typeface terminology, Specifying Typefaces, fonts; list tables and forms: list-style, table properties, styling forms, styling text input. Layout and positioning: layout: key concepts in positioning elements, controlling the position of elements: relative positioning, absolute positioning, fixed positioning, z-index, float, clear, creating multi column layout with float, fixed width layout, liquid layout, layout grids, Images: controlling size of images in CSS, aligning images using CSS, centering images using CSS, background images, gradients, Media Queries.
Suggested Readings: 1. Thomas A. Powell: Web Design: The Complete Reference, 4/e, Tata McGraw-Hill 2. Wendy Willard: HTML Beginners Guide, Tata McGraw-Hill. 3. Web Design: The Complete Reference , 4/e, Thomas A. Powell, Tata McGraw-Hill 4. Deitel and Goldberg: Internet and World Wide Web, How to Program, PHI. 5. Any other book(s) covering the contents of the subject. Note: Latest and additional good books may be suggested and added from time to time.
List of Practical/Programs/Lab Assignments
1. WAP that explains the HTML style Attributes to specify the style properties for the HTML element. 2. WAP that illustrates the HTML heading tags. 3. WAP that illustrate the use of <p> and <hr> HTML tags 4. WAP that illustrate the use of different text formatting HTML tags 5. WAP that illustrate the use of RGB color values to style our html elements. 6. WAP that Creating a simple table in HTML using a table tag 7. WAP that demonstrates an ordered list () with five programming languages listed sequentially, each item numbered automatically by the browser. 8. WAP for Implementation of Unordered List horizontally. 9. WAP that illustrate the use of CSS color, font family and font size properties. 10. WAP that illustrate the use CSS Table properties. 11. WAP to create a 3D Groove border using CSS. 12. WAP to dynamically change color by percentage CSS. 13. WAP to set the order of the flexible items using CSS. 14. WAP to create reflection effect using HTML and CSS. 15. WAP to create Image Folding Effect using HTML and CSS. 16. WAP to design a Parallax Webpage using HTML & CSS. 17. WAP to create RGB color generator using HTML CSS and JavaScript. 18. WAP to use Google material icon as list-style in a webpage using HTML and CSS. 19. WAP to design a calendar using HTML and CSS? 20. WAP to design Background color changer using HTML CSS and JavaScript. 21. WAP to create Image Stack Illusion using HTML and CSS. 22. WAP to create Hover animations on Button. 23. Create a web page that contains the image of a human. Form an image map such that the user clicks on any part of the body, a web page showing information on that part of the body is displayed. 24. Create a registration form for your institution which will ask first name, last name, age, sex, date of birth, class, address, contact number, email id from the candidate. When you submit the form it should display the confirmation message that “You details are saved” (use JavaScript for displaying the

- confirmation message).
25. Create an HTML page with a paragraph. Apply a style directly using inline styles to change the color and font size. Then use internal CSS to change the background color of the paragraph.
 26. Create a nested HTML structure (e.g., a `<div>` containing a `<p>` and an `<a>`). Apply CSS styles to the parent `<div>` and observe how styles are inherited by child elements. Modify styles on child elements to see how inheritance works.
 27. Create a webpage that includes an iframe displaying another webpage. Ensure the iframe has a set width and height, and add a border around it.
 28. Create an HTML page and use different types of CSS selectors (universal, type, class, ID, attribute, descendant, child, sibling). Apply styles using these selectors and Design an HTML table and use CSS to style it. Apply borders, padding, background colors, and hover effects
 29. Create an HTML page with a background image and a foreground text. Use CSS to set the opacity of the background image .
 30. Design an HTML page with overlapping elements. Use CSS to apply different z-index values to control the stacking order of the elements.
 31. Create a HTML form that has number of Textboxes. When the form runs in the Browser fill the textboxes with data. Write JavaScript code that verifies that all textboxes has been filled. If a textboxes has been left empty, popup an alert indicating which textbox has been left empty
 32. Create an HTML page and use CSS media queries to apply different styles for different screen sizes (e.g., mobile, tablet, desktop).
 33. Create a form having number of elements (Textboxes, Radio buttons, Checkboxes and so on). Write JavaScript code to count the number of elements in a form
 34. Write a JavaScript program that creates an array of 5 numbers. Use array methods to add a number to the end, remove a number from the beginning, and find the index of a particular number.
 35. Write a JavaScript function that changes the background color of a div element when a button is clicked. Provide multiple buttons to change to different colors.

Any other Practical/Lab Assignments/Programs assigned by the teachers.

Session: 2024-25

Second Semester

Name of the Program	Skill Enhancement Courses (Single Major / Multidisciplinary Programs)	Program Code	-----
Name of the Course	Web Development-II	Course Code	24CSC402SE01
Hours/Week	1+0+4	Credits(L:T:P)	1:0:2
Max. Marks	Theory: 25 (20+5) Practical: 50	Time of end term Examination	3 Hours
Note: The examiner has to set nine questions in all by setting two questions from each Unit and Question No. 1 consisting of short-answer type questions covering the entire syllabus. Student will be required to attempt five questions in all by selecting one question from each Unit and Question No. 1, which is compulsory.			
Course Objective: The objective of this course is to acquire knowledge and Skills for creation of Web Sites. Also to acquire the knowledge regarding creation of Web applications using tools and techniques used in industry and how to design a basic web site using HTML & CSS to demonstrate responsive web design.			
Course Outcomes: By the end of the course the students will be able to: CO1: Understand the fundamental concepts of XML and its structure CO2: Work with XML schema and Web casting techniques for Web development. CO3: Learn the concept of Website planning. CO4: Get acquaintance with different web technologies like AJAX, JSON, Node.js, MongoDB etc.. CO5: Learn fundamentals and essentials of PHP programming.			
Unit-I			
XML: Introduction, Syntax, Document structure, Document Type definitions, Namespaces. XML Schemas: Displaying raw XML documents, Displaying XML documents with CSS, XSLT style sheets, XML Processors, Web services. ECMA Script: ECMA Script versions, ES5 Features, ES6 introduction, Var Declarations and Hoisting, let declaration, Constant declaration, function with default parameter values, default parameter expressions, unnamed parameters, the spread operator, arrow functions, object destructuring, array destructuring, sets and maps, Array, find(), Array, findIndex(), template strings, Javascript classes, callbacks, promises, async/await.			
Unit-II			
AJAX: Introduction, Need for AJAX, Working of AJAX, Handling Ajax request and response, data formats: XML, JSON, Working with JSON data, Loading HTML with Ajax, Loading XML with Ajax, Loading JSON with Ajax. JQuery: Introduction, A basic JQuery example, Need of JQuery, finding elements, JQuery selection, getting element content, updating elements, changing content, inserting elements, adding new content, getting and setting attributes, getting and setting CSS properties, using each(), events, event object, effects, animating CSS properties, using animation, traversing the DOM, working with forms, JavaScript libraries, JQuery and Ajax.			
Unit-III			
Web Servers: Introduction, HTTP Transactions, Multitier Application Architecture, Client SideScripting versus Server-Side Scripting, Accessing Web Servers. Server Side Scripting with Node.js: Getting to know node, node.js changed JavaScript forever, features of node, when to use and not use node, asynchronous callbacks, the NoSql movement, node and MongoDB in the wild, Hello World in Node, package.json, modules, Built-in Modules: FS Module, HTTP Module, Events; Node Package Manager(npm), web server using http, node.js with express, middleware, routing in express, CRUD operations in express, web server using express, making it live on Heroku			

Node.js with MongoDB: Basics of MongoDB, MongoDB CRUD Operations, Building a datamodel with MongoDB and Mongoose, Defining simple mongoose schemas, build node express app with MongoDB.
Unit-IV
Introduction to PHP: Basic Knowledge of websites ; Introduction of Dynamic Website ; Introduction to PHP ; Why and Scope of PHP ; XAMPP and WAMP Installation PHP Functions ; PHP Functions Creating an Array ; Modifying Array Elements ; Processing Arrays with Loops ; Grouping Form Selections with Arrays ; Using Array Functions ; Using Predefined PHP Functions ; Creating User- Defined Functions PHP Programming Basics ; Syntax of PHP ; Embedding PHP in HTML ; Embedding HTML in PHP ; Introduction to PHP Variable ; Understanding Data Types ; Using Operators ; Using Conditional Statements ; If(), else if() and else if condition Statement ; Switch() Statements ; Using the while() Loop ; Using the for() Loop.
Suggested Readings: 1. Bing Liu, Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data, Second Edition, Springer. 2. Ramesh Sharda, Dursun Delen, Efraim Turban, Business Intelligence and Analytics: Systems for Decision Support, Pearson Education. 3. Reza Zafarani, Mohammad Ali Abbasi, and Huan Liu, Social Media Mining: An Introduction, Cambridge University Press. 4. Bing Liu, Sentiment Analysis and Opinion Mining, Morgan & Claypool Publishers. 5. Nitin Indurkha, Fred J. Damerau, Handbook of Natural Language Processing, Second Edition, CRC Press. 6. Matthew A. Russell, Mining the Social Web, Second Edition, O'Reilly Media. 7. Thomas A. Powell, Web Design: The Complete Reference, Fourth Edition, Tata McGraw-Hill. 8. Deitel and Goldberg, Internet and World Wide Web: How to Program, PHI. 9. Any other book covering the contents of the subject. Note: Latest and additional good books may be suggested and added from time to time.
List of Practical/Programs/Lab Assignments
1. Create an XML document to store information about a collection of books, including elements for title, author, publication year, and genre. Write a simple HTML page that uses JavaScript to load and display the XML data. 2. Create an XML file and an XSLT stylesheet to transform the XML data into an HTML table. Use the transformNode or XSLTProcessor methods to apply the transformation and display the result in a web page. 3. Write an HTML page with a script that loads an XML file containing a list of employees. Parse the XML data with JavaScript and display the employee names and departments in a list on the web page. 4. Develop a form that allows users to submit their contact information (name, email, message). Use AJAX to send the form data to a server-side script (e.g., a PHP script) that echoes the received data. Display the server's response on the web page. 5. Write an Ajax script to get player details from xml file when user select player name in any game. Also create xml file to store details of player(name,country,wickets and runs) 6. Write an AJAX request to a URL that does not exist. Implement error handling to display a user-friendly message when the request fails. 7. Write a jQuery code to select values from a JSONObject. Also Write a JQuery code to remove all CSS classes from an Applications. 8. Create a PHP script that defines variables of different data types (string, integer, float, boolean, array). Output the values and types of these variables using echo and var_dump. 9. Develop an HTML form to upload a file. Write a PHP script to handle the file upload, save the file on the server, and display a message indicating the file upload status. 10. Create a MySQL database and table to store user information (name, email). Write a PHP script to connect to the database and insert a new user record. Display a success message upon successful insertion. 11. Create a web page for Travel agency using PHP. Any other Practical/Lab Assignments/Programs assigned by the teachers.

SEC 3 @ 3 credits	Java Programming	25CSC403SE01	1	0	2	3	1	0	4	5	20	05	50	75
--------------------------	------------------	--------------	---	---	---	---	---	---	---	---	----	----	----	----

Session: 2025-26

Third Semester

Name of Program	Skill Enhancement Courses	Program Code	
Name of Course	Java Programming	Course Code	25CSC403SE01
Hours/Week	1+0+4	Credits (L: T:P)	1:0:2
Max. Marks	Theory: 25(20+5) Practical: 50	Time of Examination	3 Hours

Note: The examiner has to set nine questions in all by setting two questions from each Unit and Question No. 1 consisting of short-answer type questions covering the entire syllabus. Student will be required to attempt five questions in all by selecting one question from each Unit and Question No. 1, which is compulsory. All questions will carry equal marks.

Course Outcomes:

This course is designed to provide students with foundational knowledge of Java programming, focusing on its environment, syntax, and structure. It introduces the key principles of object-oriented programming (OOP), including classes, inheritance, and interfaces. The course also emphasizes error handling, package creation, and graphical user interfaces to enable the development of robust Java applications.

Course Objectives:

By the end of the course, the students will be able to:

CO1: To introduce the basics features of Java.

CO2: Set up the Java development environment and demonstrate proficiency in fundamental programming constructs, including data types, control flow, and basic syntax.

CO3: Work with arrays, strings, and develop modular programs using classes, objects, and access modifiers.

CO4: Apply inheritance, polymorphism, and interfaces to build flexible and reusable programs.

CO5: Handle exceptions effectively, create packages, and design graphical user interfaces using Java frameworks.

Unit I

Overview of Java: Java Features, Setting Up the Java Development Environment, Writing, Compiling, and Running a Java Program, Java Virtual Machine (JVM) and Bytecode, Java Development Kit (JDK) and Java Runtime Environment (JRE)

Basic Language Elements: Java Syntax and Structure, Identifiers, Keywords, Literals, Comments, Operators Assignments, Data Types, Variables and its types, Constants, Expressions, Control Flow Statements: if-else, switch, loops (for, while, do-while).

Unit II

Array: Initializing, Single Dimensional Array, Operation on String, Mutable & Immutable String, Using Collection, Bases Loop for String, Creating Strings using StringBuffer.

Class Fundamentals: Object, Object Life cycle, Creating and Operating Objects, Constructor & initialization code block, Access Control, Inner Class, Abstract Class, Argument Passing Mechanism, Method Overloading, Recursion, Use of Access Modifiers with Classes & Methods.

Unit III

Introduction to Inheritance: Use and Benefits of Inheritance in OOPs, Types of Inheritance in Java, Inheriting Data members and Methods, Role of Constructors in inheritance, Overriding Super Class Methods,

Use of “super”, Polymorphism in Java. Interface: Purpose of interface, defining an interface, implementing interfaces, Interface reference variables, Interface with variables, Extending interfaces
Unit IV
Exception Handling: Types of Errors in Java, Try-Catch Blocks and Finally Clause, Throw and Throws Keywords, Creating Custom Exceptions. Packages: Package as Access Protection, Defining Package, CLASSPATH Setting for Packages, Import and Naming Convention for Packages.GUI programming.
Suggested Readings: 1. The Complete Reference JAVA, TMH Publication. 2. Beginning JAVA, Ivor Horton, WROX Public. 3. JAVA 2 UNLEASHED, Tech Media Publications. 4. JAVA 2(1.3) API Documentations. 5. Any other book(s) covering the contents of the subject. Note: Latest and additional good books may be suggested and added from time to time.
List of Practical/Programs/Lab Assignments
1. Set up the Java Development Kit (JDK) and compile & execute a basic Java program. 2. Write a Java program to demonstrate variables, data types, and type casting. 3. Implement control flow structures (if-else, switch-case, loops) in Java. 4. Develop an array-based Java program to perform basic operations (insert, delete, search). 5. Write a program to manipulate strings using String and StringBuffer classes. 6. Implement a Java class with constructors and object instantiation. 7. Demonstrate the use of method overloading and recursion. 8. Create a simple calculator program using user-defined functions and access modifiers. 9. Implement single and multilevel inheritance using Java classes. 10. Implement polymorphism and method overriding in Java. 11. Develop an application that demonstrates the working of interfaces. 12. Write a program to handle exceptions using try-catch-finally blocks. 13. Create and handle user-defined exceptions in Java. 14. Write a Java program to define and use custom packages. 15. Implement file handling in Java (read/write operations using FileReader/FileWriter). 16. Develop a basic GUI-based Java program using AWT components (Button, Label, TextField). 17. Create an event-driven Java GUI application. 18. Build a basic student record management system using OOP concepts. 19. Implement collection framework classes such as ArrayList and HashMap. 20. Design a multithreaded Java application to perform parallel computation. <i>Any other Practical/Programs/Lab Assignments given by the teachers.</i>

Session: 2026-27
Sixth Semester
(Offered only in case of Single Major Program)

Name of the Program	Skill Enhancement Courses	Program Code	----
Name of the Course	Machine Learning and Python Programming	Course Code	26CSC406SE01
Hours/Week	0+0+4	Credits(L:T:P)	0:0:2
Max. Marks	Practical:50	Time of end term examination	3 Hours
Course Objective:			
The course is designed to impart knowledge of important concepts of Machine learning and one of the latest and most powerful programming languages – Python. Python programming is intended for software engineers, system analysts, program managers and user support personnel who wish to learn the Python programming language.			
Course Outcomes:			
By the end of the course the students will be able to: CO1: Develop problem-solving skills and critical thinking in Python using basic programming constructs including variables, operators and data types. CO2: Will be able to learn the automating repetitive tasks using loop and conditional controlled statements. CO3: Understand the complex data types including lists, tuples, dictionaries and Function packages. CO4: Identify and use libraries for algorithmic thinking to implement various data structures. CO5: Will be able to implement important concepts of database programming.			
Unit-I			
Introduction to Python: History and Features of Python Programming, Basics of Python: Keywords, Variables, Operators, I/O Statements, Indentation, and Comments. Python Basic Data Types, Data Types Declaration, and Implementation.			
Unit-II			
Flow Control Statement: if statement, if-else statement, nested-if statement, if-elif-else ladder, While loop, range() Function, For Loop, Nested Loops, Infinite Loop, Break Statement, Continue Statement, Pass Statement.			
Unit-III			
Python Complex data types: String Data Type, String Manipulation Methods and implementation using Python Programming, List and Dictionary Data Type, Declaration, and Implementation using Various built-in Functions and Libraries.			
Unit-IV			
Python File Operations: Reading Files, Writing Files in Python, Understanding Read Functions: read(), readline(), readlines(), Understanding Write Functions: write() and writelines() Manipulating file pointer using seek Programming, using file operations.			
Database Programming: Connecting to a Database, Creating Tables, INSERT, UPDATE, DELETE and READ operations, Transaction Control, Disconnecting from a database, and Exception Handling in Databases.			

Suggested Readings:

1. Allen B. Downey, Think Python: How to Think Like a Computer Scientist, 2nd Edition, Green Tea Press.
2. Al Sweigart, Automate the Boring Stuff with Python.
3. Charles Dierbach, Introduction to Computer Science Using Python, 1st Edition, Wiley India Pvt. Ltd.
4. Wesley J. Chun, Core Python Applications Programming, 3rd Edition, Pearson Education India.
5. Roberto Tamassia, Michael H. Goldwasser, and Michael T. Goodrich, Data Structures and Algorithms in Python, 1st Edition, Wiley India Pvt. Ltd.
6. Reema Thareja, Python Programming Using Problem Solving Approach, Oxford University Press.
7. Charles R. Severance, Python for Everybody: Exploring Data Using Python 3, 1st Edition, Shroff Publishers.
8. Any other book(s) covering the contents of the subject.

Note: Latest and additional good books may be suggested and added from time to time.

List of Practical/Programs/Lab Assignments

1. Find the sum of given numbers using function with default arguments.
2. Demonstrate default constructor or no argument constructor.
3. Generate prime numbers between the given two numbers.
4. Perform arithmetic operations using Inline function.
5. Accept a three digit number and display it in words eg. 123 be printed out as One Two Three.
6. Swap two values using methods of passing arguments in function
7. Prepare a student Record using class and object.
8. Find the area of geometric shapes using function overloading.
9. Illustrate the use of Friend function.
10. Demonstrate parameterized constructor.
11. Demonstrate copy constructor with constructor overloading.
12. Demonstrate destructors and constructor using "this: pointer.
13. To Perform Addition Subtraction Multiplication Division
14. To Check Whether a Character is Alphabet or Not
15. To Check Whether a Character is Vowel or Not
16. To Count Zeros, Positive and Negative Numbers
17. To Remove all Non Alphabet Characters from a String
18. To Find Average of Numbers Using Arrays
19. To Find Size of Variables using sizeof Operator
20. To Check Leap Year.

Any other Practical/Lab Assignments/Programs assigned by the teachers.

Session: 2027-28 Seventh Semester

Name of the Program	Skill Enhancement Courses	Program Code	-----
Name of the Course	MERN Stack Development	Course Code	27CSC407SE01
Hours/Week	2+0+4	Credits(L:T:P)	2:0:2
Max. Marks	Theory: 50 (35+15) Practical:50	Time of end term examination	3 Hours
Note: The examiner has to set nine questions in all by setting two questions from each Unit and Question No. 1 consisting of short-answer type questions covering the entire syllabus. Student will be required to attempt five questions in all by selecting one question from each Unit and Question No. 1, which is compulsory. All questions will carry equal marks.			
Course Objective: The objective of this course is to equip students with the skills and knowledge necessary to develop modern web applications using the MERN stack. Students will learn about each component of the stack - MongoDB, Express.js, React.js, and Node.js - and understand their roles in web development.			
Course Outcomes: By the end of the course the students will be able to: CO1: Analyze complex requirements and design scalable web applications using the MERN stack. CO2: Develop RESTful APIs using Express.js and integrate them with MongoDB for data storage. CO3: Build interactive and responsive user interfaces using React.js and implement state management solutions. CO4: Implement server-side logic and event-driven programming using Node.js for robust backend development. CO5: Apply error handling and middleware techniques in Express.js applications for reliable and maintainable server-side functionality.			
Unit-I			
The MERN stack: overview of MongoDB, Express.js, React.js, and Node.js. Role of each in modern web development. Setting up development environment: Installation, configuration- tools and dependencies for MERN stack development.			
Unit-II			
Building a basic MERN application: Creating a simple web application from scratch using MongoDB for data storage, Express.js for server-side logic, React.js for client-side rendering, and Node.js for backend development. Server-Side Development: Node.js and event-driven programming, asynchronous nature of Node.js, eventdriven architecture.			
Unit-III			
Creating RESTful APIs with Express.js: Designing and implementing RESTful APIs using Express.js framework for handling HTTP requests and responses. Data modelling with MongoDB and Mongoose: defining MongoDB schemas, overview of Mongoose ORM, models development using Mongoose ORM, Interacting MongoDB database.			
Unit-IV			
Express.js applications: Error handling- error handling middleware, middleware in Express.js- implementing and using middleware for request processing. Client-Side Development: introduction to React.js, its components, component-based architecture – fundamentals of React.js library, component-based user interface development.			

Suggested Readings:

1. S. Subramanian, Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node, Apress.
2. K. Chinnathambi, Learning React: A Hands-On Guide to Building Web Applications Using React and Redux, Addison-Wesley Professional.
3. M. Casciaro, Node.js Design Patterns: Master best practices to build modular and scalable server-side web applications, Packt Publishing.
4. E. Hahn, Express.js in Action, Manning Publications.
5. S. Bradshaw, E. Brazil, and K. Chodorow, MongoDB: The Definitive Guide, O'Reilly Media.
6. M. Riva, React Design Patterns and Best Practices: Build easy to scale modular applications using the most powerful components and design patterns, Packt Publishing.
7. Accomazzo, N. Murray, and A. Lerner, Fullstack React: The Complete Guide to ReactJS and Friends, Fullstack.io.
8. Dayley and B. Dayley, Node.js, MongoDB, and AngularJS Web Development, Addison-Wesley Professional.
9. Mead, Learning Node.js Development: Learn the fundamentals of Node.js, and deploy and test Node.js applications on the web, Packt Publishing.
10. S. Grubor, Deployment with Docker: Apply continuous integration models, deploy applications quicker, and scale at large by converting applications into Docker containers, Packt Publishing.
11. Any other book(s) covering the contents of the paper in more depth.

Note: Latest and additional good books may be suggested and added from time to time.

List of Practical/Programs/Lab Assignments

1. Create a new React project using create-react-app to render "Hello World" on the screen using Node.js and npm.
2. Implement a simple REST API routes for CRUD operations on MongoDB using Express.js and Mongoose.
3. Implement event handlers to update state of a form based on user input for a form component created using React.
4. Implement client-side routing in a React application using React Router.
5. Implement user authentication in a MERN stack application using JSON Web Tokens (JWT).
6. Demonstrate the ability to install and configure the MERN stack components and ensure they work together correctly.
7. Create a functional web application, demonstrating the integration of all four technologies.
8. Demonstrating the handling of HTTP requests and responses.
9. Enhance the "To-Do List" application by adding middleware for logging requests and handling errors. Implement custom error handling middleware to manage different types of errors (e.g., 404 Not Found, 500 Internal Server Error) and Implement middleware and error handling in Express.js applications to ensure robust and maintainable server-side functionality.
10. Develop a user interface for a "Movie Review" application using React.js. Implement components for displaying a list of movies, adding new reviews, and filtering movies by genre. Use state management techniques to manage the application state and objective is to Build interactive and responsive user interfaces using React.js, applying component-based architecture and state management solutions.

Any other Practical/Lab Assignments/Programs assigned by the teachers.

Session: 2027-28 Eighth Semester

Name of the Program	Skill Enhancement Course	Program Code	----
Name of the Course	Advanced Machine Learning and Generative AI	Course Code	27CSC408SE01
Hours/Week	2+0+4	Credits (L:T:P)	2:0:2
Max. Marks.	Theory: 50 (35+15) Practical: 50	Time of Examination	3 Hours

Note: The examiner has to set nine questions in all by setting two questions from each Unit and Question No. 1 consisting of short-answer type questions covering the entire syllabus. Student will be required to attempt five questions in all by selecting one question from each Unit and Question No. 1, which is compulsory. All questions will carry equal marks.

Course Outcomes:

By the end of the course the students will be able to:

- CO1: To understand the fundamental principles, theory and approaches of deep neural networks.
- CO2: To learn the main variants of deep learning (such as recurrent and convolutional architectures).
- CO3: To understand the key concepts, issues and practices when training and modeling with deep architectures.
- CO4: Design and train neural networks for classification, regression, and generative tasks.
- CO5: Build generative AI systems (e.g., text, image synthesis) using modern frameworks.

Unit-I

Introduction: Machine learning, Examples of Machine Learning Problems, Structure of Learning, learning versus Designing, Training versus Testing, Characteristics of Machine learning tasks, Predictive and descriptive tasks, Machine learning Models: Geometric Models, Logical Models, Probabilistic Models. Learning via Uniform Convergence. The VC-Dimension.

Linear Predictors: Half spaces, Linear Regression, Logistic Regression, Boosting: Weak Learnability, AdaBoost, Linear Combination of Base Hypotheses, AdaBoost for Face Recognition
Model Selection and Validation: Model Selection Using SRM, Validation.

Unit-II

Probabilistic Model: Normal Distribution and Its Geometric Interpretations, Naïve Bayes Classifier, Discriminative learning with Maximum likelihood, Probabilistic Models with Hidden variables: Estimation-Maximization Methods, Gaussian Mixtures, and Compression based Models.

Recurrent Neural Networks (RNN): Concepts, Bidirectional RNNs, Encoder-Decoder Models, Backpropagation Through Time (BPTT), and applications. Advanced RNN Architectures: LSTMs, GRUs and Seq2Seq Models. **Convolutional Neural Networks (CNN):** Basics, building blocks, backpropagation, dropout layers, optimizers (Momentum, RMSProp, Adam), and architectures (LeNet, AlexNet, VGG16, ResNet).

Unit-III

Generative Model Fundamentals: Theoretical foundations of generative AI: generative modelling paradigms and applications, Introduction to Generative Adversarial Networks (GANs): architecture, training dynamics, and loss functions.

Variational Autoencoders (VAEs): understanding the latent space and applications in data generation, Variational Autoencoders (undercomplete, regularized, sparse, denoising), Representational power, layer, size and depth of autoencoders.

Unit-IV

Applications of Generative AI: Implementing generative models using deep learning frameworks (e.g., TensorFlow, PyTorch) for practical applications, Generating realistic images (e.g., faces, landscapes) using GANs and exploring creative applications (art generation).

Text generation and manipulation with generative models (e.g., creating realistic dialogue, machine

translation), Investigating the potential of generative AI in various domains (e.g., drug discovery, material science)

Suggested Readings:

1. Goodfellow, Y. Bengio, and A. Courville, Deep Learning. Cambridge, MA: The MIT Press.
2. R. O. Duda, P. E. Hart, and D. G. Stork, Pattern Classification. New York, NY: John Wiley & Sons.
3. F. Chollet, Deep Learning with Python. Shelter Island, NY: Manning Publications.
4. S. Raschka and V. Mirjalili, Python Machine Learning: Machine Learning and Deep Learning with Python, scikit-learn, and TensorFlow 2, 3rd ed. Birmingham, U.K.: Packt Publishing.
5. J. Brownlee, Deep Learning for Computer Vision: Image Classification, Object Detection, and Face Recognition in Python. California, USA: Machine Learning Mastery.
6. Goodfellow and Y. Bengio, Learning Deep Architectures for AI. Hanover, MA: Now Publishers Inc.
7. T. Hastie, R. Tibshirani, and J. Friedman, The Elements of Statistical Learning: Data Mining, Inference, and Prediction, 2nd ed. New York, NY: Springer.
8. Géron, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems, 2nd ed. Sebastopol, CA: O'Reilly Media.
9. D. Foster, Generative Deep Learning: Teaching Machines to Paint, Write, Compose, and Play. Sebastopol, CA: O'Reilly Media.
10. J. Hawkins, S. Blakeslee, A Thousand Brains: A New Theory of Intelligence. New York, NY: Basic Books.

Note: Latest and additional good books may be suggested and added from time to time.

List of Practical/Lab Assignment

1. Implement a simple linear classifier using Python. Train it on a small dataset (e.g. Iris dataset) and evaluate its performance.
2. Visualize the decision surface of a linear classifier on a 2D dataset using matplotlib. Show how the decision surface changes with different parameters.
3. Implement gradient descent optimization for a simple regression problem. Plot the cost function over iterations to visualize the convergence.
4. Implement a simple MLP using a deep learning framework (e.g., TensorFlow or PyTorch). Train it on the MNIST dataset and evaluate its performance.
5. Implement and compare different optimizers (SGD, Momentum, RMSProp, Adam) on a simple MLP.
6. Implement early stopping and dropout in training a deep neural network. Train it on the CIFAR-10 dataset and evaluate the impact on performance and overfitting.
7. Implementing a Simple RNN for Time Series Prediction
8. Implement LSTM for Text Generation
9. Implement the CNN with Inception Module for Image Classification.
10. Implement a simple residual network (ResNet) and train it on the CIFAR-10 dataset.
11. Implement Hyperparameter Tuning with Keras Tuner in python
12. Create a program using Adam Optimizer for Training a Neural Network
13. Use python programming to have a basic GAN implementation
14. Write python code implementing VariationalAutoencoder (VAE) for Data Generation
15. Write python code to implement GAN for Image Generation.

Any other Practical/Lab Assignments/Programs assigned by the teachers.

Name of the Program	Skill Enhancement Courses	Program Code	----
---------------------	---------------------------	--------------	------

Name of the Course	Social Media Analytics	Course Code	27CSC408SE01
Hours/Week	3+0+2	Credits(L:T:P)	3:0:1
Max. Marks	Theory 75 (50+25) Practical:25	Time of end term examination	3 Hours
Note: The examiner has to set nine questions in all by setting two questions from each Unit and Question No. 1 consisting of short-answer type questions covering the entire syllabus. Student will be required to attempt five questions in all by selecting one question from each Unit and Question No. 1, which is compulsory. All questions will carry equal marks.			
Course Objective: The course is designed to impart knowledge of one of the latest and most powerful programming languages – Python. Python programming is intended for software engineers, system analysts, program managers and user support personnel who wish to learn the Python programming language.			
Course Outcomes: By the end of the course the students will be able to: CO1: Understand the concept of Social media. CO2: Understand the concept of social media Analytics and its significance. CO3: Analyze the effectiveness of social media. CO4: Use different Social media analytics tools effectively and efficiently. CO5: Use different effective Visualization techniques to represent social media analytics.			
Unit-I			
Social Media Analytics -An Overview: Core Characteristics of Social Media, Types of SocialMedia, Social media landscape, Need for Social Media Analytics (SMA), SMA in small & large organizations. Purpose of Social Media Analytics, Social Media vs. Traditional Business Analytics, Seven Layers of Social Media Analytics, Types of Social Media Analytics, Social Media Analytics Cycle, Challenges to Social Media Analytics, Social Media Analytics Tools.			
Unit-II			
Social Network Structure, Measures & Visualization: Basics of Social Network Structure - Nodes, Edges & Tie. Networks Measures - Degree Distribution, Density, Connectivity, Centralization, Tie Strength & Trust. Network Visualization-Graph Layout, Visualizing Network features, Scale Issues. Social Media Network Analytics - Common Network Terms, Common Social Media Network Types, Types of Networks, Common Network Terminologies, Network Analytics Tools.			
Unit-III			
Social Media Text, Action & Hyperlink Analytics: Social Media Text Analytics - Types of Social Media Text, Purpose of Text Analytics, Steps in Text Analytics, Social Media TextAnalysis Tools. Social Media Action Analytics –Introduction to Actions Analytics, Common Social Media Actions, Actions Analytics Tools. Social Media Hyperlink Analytics - Types of Hyperlinks, Types of Hyperlink Analytics, Hyperlink Analytics Tools.			
Unit-IV			
Social Media Location & Search Engine Analytics: Location Analytics - Sources of Location Data, Categories of Location Analytics, Location Analytics and Privacy Concerns, Location Analytics Tools. Search Engine Analytics - Types of Search Engines, Search Engine Analytics, Search Engine Analytics Tools.			
Social Media Analytics Applications and Privacy: Social media in public sector - Analyzing public sector social media, analyzing individual users, case study. Business use of Social Media - Measuring success, Interaction and monitoring, case study. Privacy - Privacy policies, data ownership and maintaining privacy online			

Suggested Readings:

1. Seven Layers of Social Media Analytics_ Mining Business Insights from Social Media Text, Actions, Networks, Hyperlinks, Apps, Search Engine, and Location Data, Gohar F. Khan,(ISBN-10: 1507823207).
2. Analyzing the Social Web 1st Edition by Jennifer Golbeck
3. Mining the Social Web_ Analyzing Data from Facebook, Twitter, LinkedIn, and Other Social Media Sites, Matthew A Russell, O'Reilly
4. Charu Aggarwal (ed.), Social Network Data Analytics, Springer, 2011
5. Social Media Analytics [2015], Techniques and Insights for Extracting Business Value Out of Social Media, Matthew Ganis, AvinashKohirkar, IBM Press
6. Social Media Analytics Strategy_ Using Data to Optimize Business Performance, Alex Gonçalves, APress Business Team
7. Social Media Data Mining and Analytics, Szabo, G., G. Polatkan, O. Boykin & A. Chalkiopoulos (2019), Wiley, ISBN 978-1-118-82485-6
8. Any other book(s) covering the contents of the subject.

Note: Latest and additional good books may be suggested and added from time to time.

List of Practical/Programs/Lab Assignments

1. Study various -
 - i) Social Media platforms (Facebook, twitter, youtubeetc)
 - ii) Social Media analytics tools(facebookinsights, google analytics netlyticetc)
 - iii) Social Media Analytics techniques and engagement metrics (page level, Post level, member level)
 - iv) Applications of Social media analytics for business. e.g. Google Analytics
2. Data Collection-Select the social media platforms of your choice(Twitter, Facebook, LinkedIn, Youtube, Web blogs etc) ,connect to and capture social media data for business (scraping, crawling,parsing).
3. Data Cleaning and Storage- Preprocess, filter and store social media data for business (Using Python, MongoDB, R, etc).
4. Text Analytics: Exploratory Data Analysis and visualization of Social Media Data for business.
5. Develop Content(text, emoticons, image, audio, video) based social media analytics model for business. (e.g. Content Based Analysis :Topic , Issue ,Trend, sentiment/opinion analysis, audio, video, image analytics) network classifier to explain decision boundaries.
6. Develop Structure based social media analytics model for any business. (e.g. Structure Based Models - community detection, influence analysis)

Any other Practical/Lab Assignments/Programs assigned by the teachers.