Unit-I: Introduction to web Technolgies

1. Explain the evolution of the web from Web 1.0 to Web 3.0 with examples.
2. Describe the Client-Server Architecture with a neat diagram and its relevance in web development.
3. Differentiate between website and web application with suitable examples.
4. Explain the role of DNS and its hierarchical structure in internet communication.
5. Compare and contrast client-side and server-side scripting languages with examples of each.
6. Explain the HTTP request-response cycle with a diagram. Discuss the significance of HTTP status codes.
7. What is the difference between HTTP and HTTPS? Explain how HTTPS ensures secure communication.
8. Describe the functions of a web browser and a web server. How do they interact in a client-server model?
9. Discuss the importance of search engines in the context of the World Wide Web. How do they work?
10. Explain URL structure and its components. How does DNS resolve a URL to an IP address?

Unit:2 HTML

1. Explain the basic structure of an HTML5 document with the help of a diagram. Include the role of <!DOCTYPE>, <html>, <head>, and <body> tags.
2. Differentiate between ordered, unordered, and description lists in HTML. Provide an example of a nested list.
3. What are semantic HTML elements? Explain the use of <header>, <footer>, <nav>, <article>, and <section> with examples.
4. Describe the various types of hyperlinks in HTML. How can you link to an external page, an internal section, and an email address?
5. Explain the <table> structure in HTML. How can you merge rows and columns? Provide a simple example.
6. Design a complete HTML form for a student registration page. Include fields for name, email, date of birth, course selection (dropdown), gender (radio buttons), address (textarea), and submit button. Use appropriate form methods and input types.
7. Compare and contrast and elements. Also explain the purpose of <iframe> and <canvas> with suitable examples.
8. Create an HTML page that includes an image, an audio player, and a video player. Explain the attributes used for each media element.
9. What are the differences between HTML4 and HTML5? Discuss at least five new features introduced in HTML5 with examples.
10. Explain the concept of HTML tags, elements, and attributes with examples. How do global attributes differ from specific attributes?

Unit:3 Cascading Style Sheet

1. Explain the CSS Box Model with a diagram. Describe how margin, border, padding, and width/height interact to affect an element's total size.
2. Differentiate between internal, external, and inline CSS with examples. Discuss the advantages of using external stylesheets.
3. What are CSS selectors? Explain the types of selectors (element, class, ID, grouping, pseudo-class) with examples.
4. Describe the principle of CSS specificity. How is specificity calculated? Provide an example to illustrate.
5. What is responsive web design? Explain how media queries help in creating layouts for different screen sizes.
6. Compare CSS Flexbox and CSS Grid layouts. Provide examples of when to use each and demonstrate a basic layout using both.
7. Design a responsive webpage layout using media queries. The layout should adapt from a single column (mobile) to two columns (tablet) to three columns (desktop). Include CSS code.
8. Explain the role of CSS preprocessors like SASS/LESS. Discuss key features such as variables, nesting, and mixins with examples.
9. Create a styled navigation bar using CSS. Include hover effects, proper spacing, and centered content. Ensure it is responsive and collapses into a hamburger menu on mobile.
10. What is the importance of CSS in modern web development? Discuss how CSS enhances separation of content and presentation, with examples of text styling, colors, and link formatting.

Unit:4 JavaScript Basics

1. Explain the different ways of embedding JavaScript in a webpage (inline, internal, external) with examples. Discuss the advantages of each method.
2. Describe JavaScript variables, data types, and operators. How does type coercion work in JavaScript? Provide examples.
3. Explain the Document Object Model (DOM) and its significance in web development. How can JavaScript access and manipulate DOM elements?
4. What are JavaScript functions? Differentiate between function declarations, function expressions, and arrow functions. Discuss scope in JavaScript.
5. Compare the different loop structures in JavaScript (while, for, do-while). Provide examples where each would be most appropriate. Write a JavaScript program that validates a simple registration form. The form should check for: empty fields, valid email format, password strength, and matching confirm password. Use appropriate dialog boxes for feedback.
6. Explain event handling in JavaScript. Demonstrate how to handle events using inline event handlers, DOM property assignment, and addEventListener() with examples of onClick, onSubmit, and onLoad.

7. Describe JavaScript arrays and their common methods. Write a program that demonstrates array operations like adding/removing elements, sorting, filtering, and mapping.
8. What are control flow statements in JavaScript? Explain if-else, switch, break, continue, and label statements with practical examples.
9. Create a dynamic webpage that demonstrates DOM manipulation. The page should include: adding/removing elements, changing styles/content, and responding to user events. Provide the complete HTML and JavaScript code.

Unit: 5 Advanced JavaScript Concepts

1. Explain the key ES6+ features: let vs const, arrow functions, and template literals. Provide examples to illustrate their advantages over older JavaScript syntax.
2. What are JavaScript Promises? Explain the states of a Promise and how .then(), .catch(), and .finally() are used for handling asynchronous operations.
3. Describe the structure and syntax of JSON. How is it different from JavaScript objects? Provide an example of JSON data and how to parse it in JavaScript.
4. What are regular expressions in JavaScript? Explain common patterns and methods like test() and exec() with examples.
5. Compare async/await with traditional Promise chaining. When would you prefer one over the other?
   Write a JavaScript function using async/await that fetches data from a public API (e.g., JSONPlaceholder) and displays it in the console. Include proper error handling using try-catch.
6. Explain AJAX and its role in modern web applications. Demonstrate how to make an AJAX request using the fetch() API to send and receive JSON data.
7. Create a JavaScript module that validates user input (email, phone number) using regular expressions. Include error messages for invalid inputs using structured error handling.
8. Compare ES5 and ES6+ object handling. Discuss object destructuring, spread/rest operators, and enhanced object literals with examples.
9. What is the Fetch API? Write a complete example demonstrating a GET request, a POST request with JSON data, and handling different HTTP status codes.

Unit:6 Introduction to JS framework and Liabries

1. Explain why jQuery was popular in web development and its main advantages over vanilla JavaScript for DOM manipulation. Provide examples of common jQuery methods.
2. Describe the fundamental concepts of jQuery, including the $() function, DOM ready event, and method chaining. Provide examples of selecting and manipulating elements.
3. Compare the three major front-end frameworks: React, Angular, and Vue.js in terms of architecture, learning curve, and community support.
4. How does jQuery simplify DOM manipulation compared to vanilla JavaScript? Demonstrate with examples of event handling and element selection.

5. What factors should be considered when choosing between a library (like jQuery) and a modern framework (like React, Angular, or Vue.js) for a web project?
6. Write a jQuery script that demonstrates DOM manipulation including: selecting elements, modifying content and styles, handling click events, and implementing simple animations. Compare this with equivalent vanilla JavaScript code.
7. Create a comprehensive comparison table of React, Angular, and Vue.js covering: virtual DOM, data binding, component architecture, state management, and build tools. Explain which framework would be best for different project types.
8. Explain the decline of jQuery in modern web development and the rise of component-based frameworks. Discuss situations where jQuery might still be appropriate versus when a modern framework is necessary.
9. Design a scenario-based question: A startup wants to build a dynamic single-page application. Compare the pros and cons of using jQuery versus React/Angular/Vue for this project, considering performance, scalability, and development speed.
10. What are the key architectural differences between jQuery (imperative programming) and modern frameworks like React (declarative programming)? How does this affect code maintainability and performance?

Unit:7 Web Design Concept

1. Explain the key principles of User Interface (UI) design. How do these principles contribute to creating effective and user-friendly interfaces?
2. What are wireframes in UI/UX design? Differentiate between low-fidelity and high-fidelity wireframes with examples of when each is used.
3. Describe the importance of typography and color theory in web design. How do they impact user perception and usability?
4. What is responsive web design? Explain how it differs from adaptive design and why it's essential in modern web development.
5. Explain the concept of user stories in UX design. How do they help in creating user-centered interfaces?
6. Design a wireframe for an e-commerce product page (mobile and desktop versions) and explain your design decisions. Include annotations for key UI elements and responsive behaviors.
7. Discuss the importance of accessibility in web design. Outline WCAG guidelines and provide specific techniques to make websites more accessible to users with disabilities.
8. How does Search Engine Optimization (SEO) relate to web design and development? Explain on-page SEO techniques that designers and developers should implement.
9. Compare and contrast UI design and UX design. How do they work together to create successful digital products? Provide examples.
10. Create a comprehensive checklist for evaluating a website's usability and accessibility. Include items related to navigation, content readability, color contrast, and keyboard accessibility.

****