END SEM

Wednesday, June 21, 2023 9:59 AM

1. Scripts and Programs:
- A script is a set of instructions written in a scripting language that can be interpreted and executed by a computer.
- Scripts are generally shorter and simpler than full-fledged programs and are often used for automating specific tasks or performing repetitive actions.

2. Origin of Scripting:
- Scripting languages evolved from the need for simpler, more accessible programming languages that could be used for quick prototyping and automation.
- Early scripting languages, such as UNIX shell scripting, focused on automating system administration tasks and controlling command-line operations.

3. Scripting Today:
- Today, scripting languages have become more versatile and widely used in various domains.
- Modern scripting languages are capable of handling complex tasks and are employed in web development, software automation, data analysis, and more.
- They offer a balance between ease of use and flexibility, allowing developers to quickly write and execute code.

4. Characteristics of Scripting Languages:
- Interpreted: Scripting languages are typically interpreted rather than compiled, allowing for immediate execution without the need for a separate compilation step.
- Dynamically Typed: Scripting languages often employ dynamic typing, allowing variables to hold values of different types without explicit type declarations.
- High-level: Scripting languages provide abstraction and simplify complex operations, allowing developers to focus on the logic rather than low-level details.

5. Uses for Scripting Languages:
- Automation: Scripting languages excel at automating repetitive tasks and streamlining workflows.
- Rapid Prototyping: They are ideal for quickly developing prototypes and proof-of-concepts due to their ease of use and faster development cycles.
- Web Development: Scripting languages like JavaScript, PHP, and Python are extensively used for client-side and server-side web development.
- Data Analysis and Processing: Scripting languages offer powerful libraries and frameworks for handling data analysis, manipulation, and visualization.

6. Web Scripting:
- Web scripting involves using scripting languages to create interactive and dynamic web pages.
- JavaScript is the primary scripting language used for client-side web scripting, allowing manipulation of HTML, CSS, and interaction with the user.
- Server-side scripting languages like PHP, Python, and Ruby are used for server-side processing and generating dynamic content.

7. The Universe of Scripting Languages:
- The scripting language landscape is vast, with numerous languages available for different purposes.
- Some popular scripting languages include JavaScript, Python, Ruby, Perl, PHP, PowerShell, and Shell scripting languages like Bash.
- Each scripting language has its strengths, community support, and specific use cases.

1. Concept of WWW:
- The World Wide Web (WWW) is a system of interconnected documents and resources accessible through the internet.
- It provides a graphical interface for navigating and accessing information using web browsers.
- The WWW is based on the principles of hypertext and hyperlinks, allowing users to navigate between web pages by clicking on links.

2. Internet and WWW:
- The Internet is a global network of interconnected computers that enables communication and data transfer between devices.
- The WWW is a service provided by the internet, allowing users to access and browse web pages using web browsers.

3. HTTP Protocol: Request and Response:
- HTTP (HyperText Transfer Protocol) is the protocol used for communication between web browsers and web servers.
- When a user requests a web page, the web browser sends an HTTP request to the server, specifying the URL and other parameters.
- The web server processes the request and sends back an HTTP response, containing the requested web page or relevant data.

4. Web browser and Web servers:
- A web browser is a software application used to access and view web pages.
- It interprets HTML, CSS, and JavaScript code and renders the web page accordingly.
- Web servers are software or hardware systems that store and serve web pages and other resources upon receiving HTTP requests.

5. HTML Basics:
- HTML (Hypertext Markup Language) is the standard markup language used to structure and present content on the web.
- HTML documents consist of HTML tags that define the structure and formatting of elements on a web page.
- Elements can include headings, paragraphs, lists, links, images, tables, forms, and more.

6. Linking in HTML:
- HTML allows for creating hyperlinks to connect different web pages or resources.
- The anchor tag (<a>) is used to create links, with the "href" attribute specifying the destination URL.

7. Image Maps:
- Image maps allow dividing an image into clickable regions, each associated with a specific URL or action.
- HTML provides the <map> and <area> tags to define and specify the clickable regions on an image.

8. Meta Information:
- Meta information provides additional information about the HTML document.
- The <meta> tag is used to include meta information, such as the document's title, author, description, keywords, and character encoding.

9. Layouts, Tables, Audio, and Video Support with HTML:
- HTML provides various elements and techniques for creating layouts and structuring content.
- Tables (<table>) are used for organizing data into rows and columns.
- HTML supports embedding audio (<audio>) and video (<video>) content, allowing playback within web pages.

10. Interactive Layout with Frames:
- HTML frames allow dividing a web page into multiple independent sections or frames, each displaying a separate HTML document.
- The <frame>, <frameset>, and <iframe> tags are used to define and implement frames.

11. FORMS and Form Control:
- HTML forms (<form>) enable users to input and submit data to the web server.
- Form controls such as text fields, checkboxes, radio buttons, dropdown menus, and buttons are used to collect user input.

1. Introduction to CSS:
- CSS stands for Cascading Style Sheets and is a styling language used to describe the presentation of a document written in HTML or XML.
- CSS allows web developers to control the layout, formatting, and appearance of web pages, ensuring consistency and separation of content and presentation.

2. Benefits of CSS: ① solves a big problem ② saves time ③ provides more attributes
- CSS provides several benefits, such as:
  - Separation of content and presentation, allowing for easier maintenance and updates.
  - Consistent styling across multiple web pages, leading to a unified look and feel.
  - Increased flexibility and control over the appearance of web elements.
  - Improved page load times by reducing code duplication through external style sheets.

 Hello world

3. Types of CSS:
- Inline CSS: Inline CSS is applied directly within the HTML element using the "style" attribute.
- Internal CSS: Internal CSS is placed within the "style" tags in the HTML document's head section.
- External CSS: External CSS is stored in separate style sheets with a .css extension and linked to the HTML document using the "link" tag.

4. Selectors and Types:
- Selectors are used to target specific HTML elements for styling in CSS.
- Types of selectors include:
  - Element selectors: Select elements based on their tag name.
  - Class selectors: Select elements with a specific class attribute value.
  - ID selectors: Select elements with a specific ID attribute value.
  - Attribute selectors: Select elements based on attribute values or presence.
  - Pseudo-classes: Select elements based on a specific state or condition

p { }
.classname { }
#id { }

(e.g., :hover, :first-child).

5. Text Formatting Properties:
- CSS provides various properties for text formatting, including:
  - font-family: Specifies the font for text.
  - font-size: Sets the size of the text.
  - font-weight: Defines the thickness or boldness of the text.
  - color: Sets the color of the text.
  - text-align: Aligns the text horizontally within its container.
  - text-decoration: Adds visual styles like underline or strikethrough to the text.

6. Box Model Concept:
- The box model describes how elements are structured and displayed in CSS.
- It consists of four parts: content, padding, border, and margin.
- The content area contains the actual content (text or images) of the element.
- Padding provides space between the content and the element's border.
- Border defines a visible boundary around the element.
- Margin creates space between the element and other elements on the page.

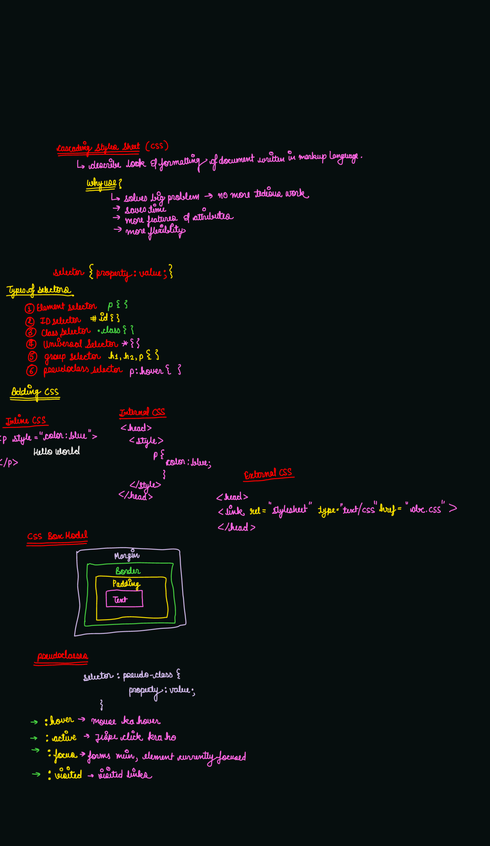
7. CSS Border and Margin Properties:
- CSS border properties allow customization of the element's border, including color, width, and style.
- CSS margin properties control the spacing around an element, setting the margin on each side (top, right, bottom, left).

8. Positioning:
- CSS positioning refers to how elements are placed and positioned on a web page.
- Common positioning values include static, relative, absolute, and fixed.
- Positioning properties like top, right, bottom, and left are used to specify the exact placement of an element.

9. Color Properties:
- CSS provides various properties for defining colors, including:
  - color: Sets the text color.
  - background-color: Sets the background color of an element.
  - border-color: Sets the color of an element's border.

10. Classes in CSS:
- CSS classes allow the grouping of multiple elements to apply the same styles.
- Classes are defined using the "class" attribute and selected using the dot (.) notation in CSS.

11. Concept of IDs and Pseudo-classes:
- IDs are used to uniquely identify individual elements in HTML.
- IDs are defined using the "id" attribute and selected using the hash (#) notation in CSS.
- Pseudo-classes are used to select elements based on specific states or conditions, such as
  - :hover for mouseover effects
  - :first-child for selecting the first child element.



1. Overview of JavaScript:
- JavaScript is a high-level, interpreted programming language primarily used for client-side web development.
- It allows adding interactivity and dynamic functionality to web pages.
- JavaScript is supported by all modern web browsers and can also be used for server-side development (Node.js).

2. Features of JavaScript:
- JavaScript is a versatile language with the following key features:
  - Interactivity: JavaScript allows handling user interactions, such as form validation and event handling.
  - Dynamic Content: It enables modifying page content dynamically without reloading the entire page.
  - Manipulating HTML and CSS: JavaScript can manipulate and modify HTML elements and CSS styles.
  - APIs and Integration: JavaScript can interact with various web APIs and services, allowing integration with external resources.
  - Asynchronous Programming: JavaScript supports asynchronous operations, making it suitable for handling network requests and timers.

3. JavaScript Identifiers:
- Identifiers are names used to identify variables, functions, objects, or other elements in JavaScript.
- They must start with a letter, underscore, or dollar sign and can contain letters, numbers, underscores, or dollar signs.
- Identifiers are case-sensitive.

4. Operators:
- JavaScript includes various operators for performing operations on data.
- Arithmetic operators (+, -, *, /, %) are used for basic arithmetic computations.
- Assignment operators (=, +=, -=, etc.) are used to assign values to variables.
- Comparison operators (==, ===, !=, !==, >, <, >=, <=) are used to compare values.
- Logical operators (&&, ||, !) are used for logical operations.

5. Control and Looping Structures:
- JavaScript provides control structures such as if-else, switch-case, and ternary operators for decision-making.
- Looping structures like for, while, and do-while are used for executing code repeatedly.

6. Introduction to Arrays:
- Arrays are used to store multiple values in a single variable.
- JavaScript arrays can contain values of different types.
- Arrays are created using square brackets ([ ]), and individual elements are accessed using their index.

7. Array Methods:
- JavaScript provides various built-in methods to manipulate arrays, such as push, pop, shift, unshift, slice, splice, etc.
- These methods allow adding, removing, and modifying elements in an array.

8. User-defined and Predefined Functions:
- JavaScript allows defining user-defined functions to encapsulate a block of reusable code.
- Predefined functions are built-in functions provided by JavaScript or specific libraries.
- Functions are declared using the function keyword and can accept parameters and return values.

9. Errors and Exceptions:
- JavaScript provides error handling mechanisms to catch and handle runtime errors and exceptions.
- The try-catch statement is used to catch and handle exceptions gracefully, preventing the script from terminating abruptly.

1. Introduction to PHP:
- PHP (Hypertext Preprocessor) is a server-side scripting language used for web development.
- It is embedded within HTML and executed on the server before sending the resulting HTML to the client's browser.
- PHP code is enclosed in special delimiters: <?php and ?>.

2. A First PHP Web Page:
- A basic PHP web page consists of HTML markup mixed with PHP code.
- PHP code is written within <?php and ?> tags, and it can be used to generate dynamic content or perform server-side operations.

3. Variables:
- Variables in PHP are used to store and manipulate data.
- PHP variables start with a dollar sign ($) followed by a name.
- They are dynamically typed and do not require explicit type declarations.
- Variables can hold various data types, such as integers, floats, strings, arrays, etc.

4. Operators and Expressions:
- PHP supports a wide range of operators for performing arithmetic, comparison, logical, assignment, and string concatenation operations.
- Expressions are combinations of variables, values, and operators that result in a value.

5. Control Statements:
- Control statements in PHP allow controlling the flow of execution based on conditions.
- Examples include if-else, switch-case, while, for, and foreach loops.
- They are used for decision-making and repetition of code blocks.

6. Functions:
- Functions in PHP encapsulate a block of reusable code that performs a specific task.
- They are declared using the function keyword and can accept parameters and return values.
- PHP provides built-in functions and allows user-defined functions.

7. Arrays:
- Arrays in PHP store multiple values in a single variable.
- PHP supports both indexed arrays (numeric keys) and associative arrays (string keys).
- Array elements can be accessed using their index or key, and arrays can be manipulated using various built-in functions.

8. String Handling in PHP:
- PHP offers numerous functions for manipulating strings, such as concatenation, length calculation, case conversion, substring extraction, and searching.

9. Using MS-Access/MySQL Databases in PHP Pages:
- PHP provides libraries and extensions to interact with databases such as MySQL or MS-Access.
- These libraries allow connecting to databases, executing queries, fetching results, and performing CRUD operations (Create, Read, Update, Delete) on database records.