Java Script Advanced (LEVEL 6)

Session 1: Advanced Functions and Callbacks

1. Anonymous Functions

- Definition: Functions that do not have a name, often used in functional programming for short-lived operations.

- Importance: Useful for concise code and passing as arguments to other functions.

2. Callback Functions

- Definition: Functions passed as arguments to other functions or returned by functions, allowing for asynchronous operations and modular code.

- Use Cases: Handling asynchronous tasks, event handling, and functional programming patterns.

3. Scope and Block Scope `{}`

- Understanding Scope: The concept of variable visibility and lifetime within different parts of the code.

- Block Scope: The scope defined by curly braces `{}`, particularly with variables declared using `let` and `const`, limiting their accessibility to the block in which they are declared.

4. Array Functions

- map: Iterates over an array and applies a given function to each element, returning a new array with the results.

- filter: Iterates over an array and returns a new array containing elements that satisfy a specified condition.

- reduce: Accumulates array elements into a single value by applying a function, often used for summing or combining values.

- forEach: Executes a provided function once for each array element, primarily used for side effects rather than returning a value.

Session 2: DOM Manipulation

1. Template Literals

- Definition: A feature allowing embedded expressions and multi-line strings using backticks (` `), facilitating dynamic string creation.

- Use Cases: Constructing strings with embedded variables or expressions, improving readability and maintainability.

2. What is DOM?

- Definition: The Document Object Model (DOM) is a programming interface for web documents, representing the structure of a document as a tree of objects. It allows JavaScript to access and manipulate HTML and CSS, dynamically updating content and style.

3. Get Element by Id

- Functionality: A method to access a single HTML element using its unique `id` attribute, enabling direct manipulation of that element's properties or content.

4. Query Selector & Query SelectorAll

- Definition: Methods for selecting elements using CSS selectors. `querySelector` returns the first matching element, while `querySelectorAll` returns a NodeList of all matching elements, providing a powerful way to access elements by class name, tag name, or other attributes.

5. Inner Text

- Definition: A property used to set or get the text content of an HTML element, excluding any HTML tags.

6. Inner HTML

- Definition: A property used to set or get the HTML content within an element, allowing for the insertion of HTML tags and content.

7. Style

- Definition: An object representing the inline styles of an HTML element, allowing for the direct manipulation of CSS properties via JavaScript.

8. classList

- Definition: A property that provides methods to manipulate the class attribute of an element, including adding, removing, and toggling classes. It is useful for dynamically changing the appearance and behavior of elements.

Session 3: Advanced DOM Manipulation

1. Create Element

- Definition: A method for dynamically creating new HTML elements, enabling the addition of new content to the DOM.

2. Append Child

- Definition: A method for adding a node to the end of a parent element's child nodes, commonly used to insert new elements into the DOM.

3. Remove Element

- Definition: A method for removing a child node from the DOM, useful for dynamically managing content on a webpage.

4. Events

- Definition: Mechanisms to handle user interactions with HTML elements, such as `onClick`, `onMouseOver`, and `onMouseOut`, enabling responsive and interactive web applications.

5. Alert, Prompt, Confirm

- Definitions:
  - Alert: Displays a message box with a single "OK" button, used for notifying users.
  - Prompt: Displays a dialog box prompting the user for input.
  - Confirm: Displays a dialog box with "OK" and "Cancel" buttons, used to confirm an action.

Session 4: Local and Session Storage

1. Local Storage

- Definition: A web storage mechanism that allows data to be stored in the browser persistently, even after the browser is closed. It is commonly used for storing user preferences or application state.

2. Session Storage

- Definition: Similar to local storage, but the data is only available for the duration of the page session. It is cleared when the page session ends, making it useful for temporary storage of data.

3. Examples

- Local Storage: Storing user settings, authentication tokens, or other persistent data.
- Session Storage: Storing temporary data such as form inputs or session-specific information.

4. Designing a Mobile Store

Session 5: CRUD Operations - Create and Display

1. Create

- Definition: The process of adding new items or data entries to a storage system, such as creating new products in an inventory.

2. Display

- Definition: The retrieval and presentation of stored data, typically by displaying it on a web page or within an application interface.

Session 6: CRUD Operations - Update and Delete

1. Update

- Definition: The modification of existing data entries in a storage system, such as editing the details of a product.

2. Delete

- Definition: The removal of data entries from a storage system, such as deleting a product from an inventory.