

# LAB ASSIGNMENT – 3 ANSWERS

## 1. Explain the differences between var, let, and const with respect to scope and hoisting.

**var**, **let**, and **const** are used for variable declaration in JavaScript, but they differ in scope and hoisting.

### Scope:

- **var** has *function scope*, meaning it is accessible anywhere inside the function in which it is declared.
- **let** has *block scope*, meaning it is accessible only inside the block { } where it is declared.
- **const** also has *block scope* and behaves like let in terms of scope.

### Hoisting:

- **var** is hoisted and initialized with *undefined*, so it can be accessed before declaration (though not recommended).
- **let** is hoisted but not initialized. Accessing it before declaration causes a *ReferenceError* due to the Temporal Dead Zone.
- **const** is hoisted but must be initialized at declaration and also has a Temporal Dead Zone.

### Reassignment:

- **var** can be re-declared and reassigned.
- **let** can be reassigned but not re-declared in the same scope.
- **const** cannot be reassigned or re-declared (for objects, only reference is constant, properties can change).

## 2. Describe the various Control Flow statements in JavaScript, specifically highlighting the difference between for, while, and do-while loops.

Control flow statements determine the order in which statements are executed in JavaScript.

Common control flow statements include:

- **if** and **if-else** – used for decision making.
- **switch** – used when multiple conditions must be checked.
- **for**, **while**, and **do-while** – used for looping or iteration.

### Difference between loops:

- **for loop** – used when the number of iterations is known. It has initialization, condition, and increment/decrement in one line.
- **while loop** – condition is checked before each iteration. It is used when the number of iterations is not fixed.
- **do-while loop** – executes the body at least once before checking the condition, so it is *exit-controlled* loop.

Thus, **do-while** always runs once, but **for** and **while** may not execute at all if the condition is false initially.

### 3. What is the Document Object Model (DOM)? Explain how to select elements and modify their content using innerText and innerHTML.

The **Document Object Model (DOM)** is a programming interface for HTML and XML documents. It represents a webpage as a hierarchical tree of nodes where each element, attribute, and text is a node.

Using DOM, JavaScript can:

- Access elements
- Change content
- Modify styles and attributes
- Add or delete elements dynamically

#### **Selecting elements:**

- `document.getElementById('id')`
- `document.getElementsByClassName('class')`
- `document.getElementsByTagName('tag')`
- `document.querySelector(selector)`
- `document.querySelectorAll(selector)`

#### **Modifying content:**

- **innerText** – changes only the text content of an element and ignores HTML tags.
- **innerHTML** – allows adding HTML markup inside an element and interprets HTML tags.

Therefore, DOM enables interaction between JavaScript and the web page structure for dynamic web development.