**React Js HandsOn-9**

**1.List the features of ES6**

**let and const** – New ways to declare variables with block scope.

**Arrow functions** – Shorter syntax for writing functions.

**Classes**– Object-oriented programming with class syntax.

**Template literals** – Using backticks (`) for string interpolation.

**Default parameters** – Assign default values to function parameters.

**Destructuring** – Easily extract values from arrays or objects.

**Modules (import/export)** – Better code organization with modular structure.

**Promises** – For easier handling of asynchronous operations.

**Rest and Spread operators (... )** – For flexible function arguments and object/array manipulation.

**New Data Structures (Map, Set)** – More efficient ways to store and manage data.

**2. Explain JavaScript let**let is used to declare block-scoped variables (only accessible within {} where defined).

Prevents redeclaration in the same scope.

Supports hoisting but is not initialized (accessing before declaration causes ReferenceError).

**3. Identify the differences between var and let**

| Feature | var | let |
| --- | --- | --- |
| Scope | Function-scoped | Block-scoped |
| Redeclaration | Can be redeclared | Cannot be redeclared |
| Hoisting | Hoisted & initialized as undefined | Hoisted but not initialized |
| Use Case | Old JavaScript (legacy code) | Modern JS (preferred) |

**4. Explain JavaScript const** -> const is used to declare block-scoped constants.

-> Value cannot be reassigned after initialization.

-> Must be initialized at the time of declaration.

-> For objects and arrays, the reference can’t change, but properties/elements can.

**5. Explain ES6 class fundamentals**

* Introduced class syntax for object-oriented programming.
* Classes are templates for objects.
* Contain constructor() for initializing objects.
* Contain methods for functionality.

**6. Explain ES6 class inheritance**

* Classes can inherit properties and methods from another class using extends.
* The super() keyword is used to call the parent class constructor.

**7. Define ES6 arrow functions**

A shorter syntax for writing functions:  
const add = (a, b) => a + b;  
 **Features:**

* No function keyword needed.
* Implicit return for single-line functions.
* Do **not bind their own**this (lexical this).

**8.Identify set(), map()**

* **Set()**
  + Stores **unique values** (no duplicates).
  + Example:

let numbers = new Set([1, 2, 2, 3]);   
 console.log(numbers);

* **Map()**
  + Stores **key-value pairs** (like objects, but keys can be any type).
  + Example:

let user = new Map();   
 user.set('name', 'Rasmita');   
 user.set('age', 21);   
 console.log(user.get('name'));