

## 1. Variables and Data Types:

- Variable declaration (var, let, const)
- Primitive data types (number, string, boolean, null, undefined)
- Complex data types (object, array)
- Type coercion and conversion
- Variable scope and hoisting

## 2. Operators:

- Arithmetic operators (+, -, \*, /, %)
- Assignment operators (=, +=, -=, \*=, /=)
- Comparison operators (==, ===, !=, !==, >, <, >=, <=)
- Logical operators (&&, ||, !)
- Ternary operator (? : )
- Bitwise operators (&, |, ^, ~, <<, >>)

## 3. Control Flow:

- Conditional statements (if-else, switch-case)
- Loops (for, while, do-while)
- Break and continue statements
- Ternary operator for conditional expressions

## 4. Functions:

- Function declaration and expression
- Parameters and return values
- Arrow functions
- IIFE (Immediately Invoked Function Expression)
- Higher-order functions and callbacks

## 5. Arrays:

- Creating and accessing arrays
- Array methods (push, pop, shift, unshift, slice, splice, concat, indexOf, forEach, map, filter, reduce)
- Iterating through arrays (for loop, forEach, map, for...of)

## 6. Objects:

- Creating objects

- Object properties and methods
- Object constructors and prototypes
- Object destructuring
- JSON (JavaScript Object Notation)

## 7. Scope and Closures:

- Global scope and local scope
- Lexical scoping
- Closure concept and examples
- IIFE (Immediately Invoked Function Expression)

## 8. DOM Manipulation:

- Selecting elements from the DOM
- Modifying element properties (textContent, innerHTML, style)
- Handling events (click, submit, keyup, etc.)
- Creating and appending elements dynamically
- Traversing and manipulating the DOM tree

## 9. Asynchronous JavaScript:

- Callbacks
- Promises
- Async/await
- Fetch API for AJAX requests
- Error handling in asynchronous operations

## 10. ES6+ Features:

- let and const
- Arrow functions
- Template literals
- Destructuring assignments
- Spread and rest operators
- Default parameters
- Modules (import and export)
- Classes and inheritance
- Iterators and generators
- Set and Map data structures

## 11. Error Handling:

- try...catch statements
- Throwing custom errors
- Error objects and properties
- Error handling best practices

## 12.Regular Expressions:

- Creating regular expressions
- Matching patterns
- Modifiers and flags
- Common metacharacters and character classes
- String methods for working with regular expressions

## 13.Modules and Bundlers:

- Module systems (CommonJS, AMD, ES6 modules)
- Using module bundlers (Webpack, Rollup)
- Tree-shaking and code splitting

## 14.JavaScript Frameworks and Libraries:

- Introduction to popular frameworks (React, Angular, Vue.js)
- Working with frameworks (components, state management, routing)
- Using libraries (lodash, moment.js, axios)

## 15.JavaScript Design Patterns:

- Singleton pattern
- Factory pattern
- Observer pattern
- Module pattern
- Prototype pattern
- MVC (Model-View-Controller) pattern

## 16.Functional Programming in JavaScript:

- Pure functions
- Immutability
- Higher-order functions
- Function composition
- Avoiding side effects

## 17.Testing and Debugging:

- Debugging techniques (console.log, breakpoints, browser dev tools)

- Unit testing frameworks (Jasmine, Mocha, Jest)
- Test-driven development (TDD)
- Writing testable code

## 18.Security Considerations:

- Cross-Site Scripting (XSS) prevention
- Cross-Site Request Forgery (CSRF) prevention
- Secure coding practices
- Handling user input securely

## 19.Performance Optimization:

- Optimizing code execution
- Reducing memory usage
- Browser rendering optimizations
- Minification and compression techniques

## 20.Browser APIs:

- DOM manipulation
- Web Storage (localStorage, sessionStorage)
- Fetch API for AJAX requests
- Geolocation API
- Web Workers for multi-threading