JAVASCRIPT

1) What is Javascript and why is it important to learn?
2) Data types in javascript
3) What are variables and different ways of using them?
4) What are declaration, assignment, statement and an expression in javascript?
5) Difference between block scope variables(let and const) vs functional scope variables (var) in javascript.
6) Functions in javascript.
7) What is hoisting in javascript.
8) Difference between function expression and function declaration in javascript.
9) How does hoisting works with variables and functions in javascript?
10) What is Scope chain(Scope and lexical environment) in javascript
    a) The Scope Chain, Scope & Lexical Environment | Namaste Java…
11) What are closures in javascript
    a) Closures in JS | Namaste JavaScript Episode 10
12) **How javascript code is executed under the hood**
    a) How JavaScript Code is executed? & Call Stack | Namaste JavaS…
13) Event loop in javascript
    a) Asynchronous JavaScript & EVENT LOOP from scratch | Namas…
14) What is the difference between synchronous and asynchronous code in JS.
15) Is Javascript synchronous / asynchronous ?
16) How you can handle asynchronous operations in javascript.
17) **Objects and Object methods in javascript**
    a) Object.create
    b) Object.keys
    c) Object.values
    d) Object.fromEntries
    e) Object.entries
    f) Object.assign
    g) Object.hasOwnProperty
    h) Object.is
    i) Object.freeze
    j) Object.seal

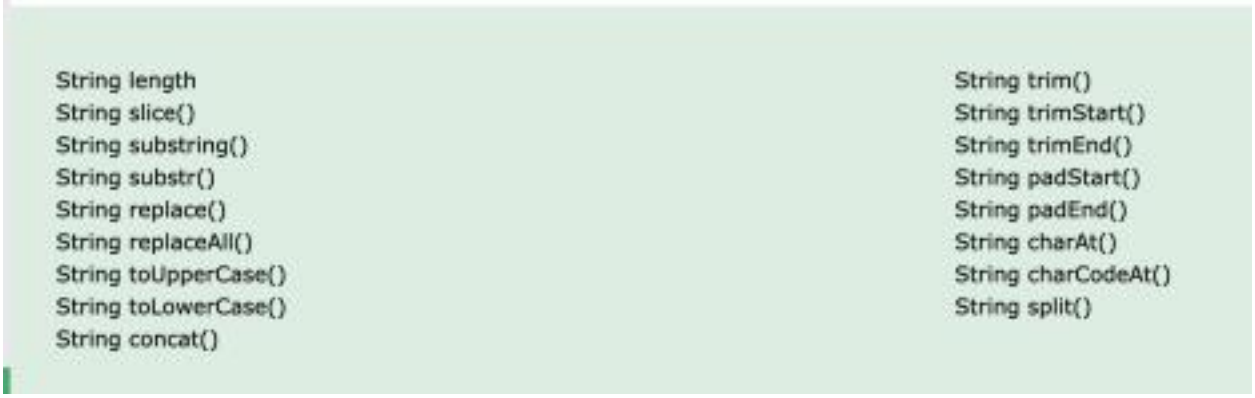
Note: these object methods are mostly used methods in regular coder life. To explore more Object methods visit https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Obje cts/Object

18) **Arrays and Array Methods in javascript**

a) Map
b) Filter
c) Reduce
d) find
e) Shift
f) Unshift
g) Push
h) Pop
i) Slice
j) Splice
k) Includes
l) isArray
m) Join
n) toString
o) Concat
p) Every
q) Some
r) forEach
s) Flat
t) flatMap
u) findIndex
v) Reverse
w) Sort

Note: For more information please visit https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array

19) **String and String methods in javascript(**Essential) → Adding a screenshot of list of string methods**



20) How many ways you can add properties to an object.
21) How to pass dynamic keys to an object and set its value accordingly.
22) **this** keyword in javascript
23) call vs apply vs bind in javascript.
24) Prototypical inheritance in javascript (Using constructor functions and classes)
25) Call by value vs call by reference in javascript
26) Some of ES6 features
    a) arrow functions
    b) let, const
    c) Multi line strings and Template literals
    d) Default parameters
    e) Destructuring
    f) Classes
    g) Promises
    h) Modules etc.
27) Difference between arrow functions and regular functions in javascript.
28) Difference between callbacks , promises and async await in javascript.
29) What are first class functions in javascript
30) Difference between spread operator and rest operator.

Note: Practicing some logical problems covering all these concepts will boost the level of confidence in understanding each concept.

React Js

1. What is ReactJS, and why is it so popular?
2. What is React.createElement().
3. What is ReactDOM
4. What is JSX, and what are its rules?
5. What is state in react?
6. What are props in React?
7. Difference between state and props.
8. What is unidirectional data flow in React?
9. Parent-to-child communication and child-to-parent communication in React
10. Event handlers and different ways of writing event handlers in React
11. Conditional rendering and list rendering
12. Why is a key prop required for list rendering in React?
13. How to apply styles in ReactJS
14. What are hooks in React?
15. rules of hooks.
16. Use cases for the following hooks:
    1. useState()
    2. useReducer()
    3. useEffect() --> with a dependency list and clean-up function
    4. useMemo() --> with a dependency list
    5. useCallback() --> with a dependency list.
    6. useRef()
    7. useLayoutEffect()
    8. useSelector()
    9. useDispacth()
    10. useContext()
17. Are state updates synchronous or asynchronous?
18. What is batching in React?
19. How many ways you can Rerender a component in React?
20. Context API (a solution for prop drilling)
21. Form handling or building forms in react
22. Custom hooks in React
23. React higher-order components vs. custom hooks/react hook
24. What is a virtual DOM?
25. What is reconciliation in React?
26. Difference between class-based components and functional components
27. Life cycle hooks of a class-based component
28. State management with Redux
    1. Data flow in Redux
    2. Redux principles
        1. single source of truth
        2. State is read-only.
        3. Changes should be made with pure functions (reducers).
    3. Handling async operations with Redux using middleware like Thunk

    4. Combining multiple reducers
29. Routing
    1. How to configure routes in React
    2. How to navigate programmatically in React
    3. Dynamic routing
    4. How to access and set dynamic parameters in Route
    5. How to access and set query parameters for a route
    6. Child/Nested routing
    7. How to implement lazy loading and code splitting
30. Migrating React 16 to React 18
31. New features in React 18