REACT INTERVIEW QUESTIONS

1. What is React?

   Ans: React is a free and open-source front-end JavaScript library for building user interfaces based on components.

2. What are the major features of React?
   Ans:

   1. Virtual DOM
   2. One-way data blinding
   3. JavaScript
   4. Speed and efficiency

3. What is JSX?

   Ans: JavaScript XML allows us to write HTML in React. JSX makes it easier to write and add HTML in React.

4. What is the difference between state and props?

   Ans:

   - State: - A object used to store datas inside components.
     - State can be modified.
     - State can be used only in class components.
     - State are passed inside a components.
   - Props: - A object used to share data inside components
     - Props cannot be modified.
     - Props can be used in class and function components.
     - Props are passed outside a component.

5. What is the difference between Real DOM and Virtual DOM?
   Ans:

   - Real DOM: If one of DOM component is change than inner DOM will be changed.
     - Working time will be more.
     - It can be directly updated in HTML.
     - It represents the UI of your application.
   - Virtual DOM: If one of the DOM components is changed then the inner DOM component does not change.
     - Working time can be saved.
     - It cannot be directly updated in HTML.
     - It is representing only in virtual DOM.

6. What are the lifecycle hooks of React?

   Ans:

   1. Mounting phase is when a new component is created and inserted into the DOM.
   2. Unmounting phase is when the component is removed from the DOM.
   3. Updating phase is when the component updates or re-renders.

7. How react is different from angular?

   Ans:

Angular	React
• Cannot change predefined framework.	• Can change in framework
• Real DOM	• Virtual DOM
• Run speed is less of react than angular	• Run speed is more of angular than react
• Component will be class-based components angular	• Component will be function based components react

8. Explain redux?

   Ans:

   - React redux is an advanced state management library for React.
   - A predictable state container designed to help you write JavaScript apps that behave consistently across client, server, and native environments.

9. What is a fragment in react?

   Ans: allows you to return multiple elements from a React component by allowing you to group a list of children without adding extra nodes to the DOM.

10. What is setState()?

    Ans: to create state using API call.

11. What is useState ()?

    Ans: to create state using API call.

12. What is useParams ()?

    Ans: used to access params from url

13. What is useDefault ()?

    Ans: Pass the default value as a parameter to the useState hook for controlled fields.

14. What is the use of render() in React?

    Ans: to display the specified HTML code inside the specified HTML element.

15. What's use effect?

Ans: to access the component at the time of open to component (similar to ngOnit)

16. What is react router?

    Ans: React Router is a JavaScript framework that lets us handle client and server-side routing in React applications.

17. Difference between functional components and class components?

    Ans:

Functional Components	Class Components
• Simple functions.	• More feature rich
• Does not use 'this' keywork	• Maintain their own private date.
• Mainly responsible for UI.	• Has complex UI Logic
• Use functional components wherever possible.	• Provide lifecycle hooks

18. Difference between statefull and stateless component?

    Ans:

Stateful Components	Stateless Components
• The same server processes all the request.	• The different servers deliver information at the same time
• There are no previous requests maintained or stored.	• Previous requests information is stored.
• The requests are independent and do not depend on previous result	• Requests based on the results of other requests
• Server requests are independent and do not depend on previous results	• Based on internet protocols that require state.

19. What are error boundaries in React

    Ans: React components that catch JavaScript errors anywhere in their child component tree, log those errors, and display a fallback UI instead of the component tree that crashed.

20. How do you style React components?

    Ans:

    1. Inline css
    2. Normal CSS

3. CSS in js
4. Styled Components
5. CSS module
6. Sass & SCSS
7. Less
8. Stylable

21. Why do React Hooks make use of refs?

    Ans: useful for accessing and manipulating DOM elements directly.

22. What are the limitations of React?

- It's a library, not a framework.
- It uses JSX.
- Dose not support SEO
- Lack of update documentation
- Fast development Speed

23. What are stateless components?

    Ans:

Stateful Components	Stateless Components
• The same server processes all the request.	• The different servers deliver information at the same time
• There are no previous requests maintained or stored.	• Previous requests information is stored.
• The requests are independent and do not depend on previous result	• Requests based on the results of other requests
• Server requests are independent and do not depend on previous results	• Based on internet protocols that require state.

24. What are stateful components?

    Ans:

Stateful Components	Stateless Components
• The same server processes all the request.	• The different servers deliver information at the same time
• There are no previous	• Previous requests

requests maintained or stored.	information is stored.
• The requests are independent and do not depend on previous result	• Requests based on the results of other requests
• Server requests are independent and do not depend on previous results	• Based on internet protocols that require state.

25. What is the use of react-dom package?

    Ans: to render components and elements on the web.

26. What is flux?

    Ans: a pattern for managing how data flows through a React application.

27. What is Redux Thunk?

    Ans: Redux Thunk middleware allows you to write action creators that return a function instead of an action

28. What is the difference between try catch block and error boundaries?

    Ans:

    - **Try…catch** is used in specific code blocks where you program the functionality of the application.
    - Try…catch deals with *imperative code* while error boundaries*deal with *declarative code*. Imperative programming is *how* you do something and declarative programming is *what* you do.

29. How to add Bootstrap to a react application?

    Ans:

    - Download bootstrap
    - To install: `npm install react-bootstrap bootstrap`

30. Can you list down top websites or applications using react as front end framework?

Ans:

Facebook, Netflix, twitter,whatsapp etc.