React.js

1. **What is the virtual DOM?**

Virtual DOM concept is a virtual representation of REAL DOM stored in memory and synced with REAL DOM using REACTDOM library.

React uses two virtual DOM s to render user interface.One is to store the current state of the object and other is store the Previous state of the object.

Whenever there is a change in virtual DOM objects ,react compares two virtual DOMs and gets to know about changes and render those changes in the Real DOM instead of rendering whole DOM.

2. **What is the primary difference between the ES6 and ES5 standards?**

ES6 introduced in 2015.It is a major version of Javascript.In ES6 let,const variable defining methods,arrow functions,spread operator,rest operators are introduced..

3. **How do you update the state of a component?**

We use   this.setState()   method in Class components .where as in functional components we use react hook 'useState' in functional components to create and update the state

4. **What is PureComponent?**

the component that do not rerenders when there is change in state or props

5. **What is React.cloneElement?**

6. **What is the difference between mapStateToProps() and mapDispatchToProps() ? Give an example.**

7. **How to add multiple middlewares to Redux?**

8. **What is wrong with this example, and how would you go -**

```
class App extends React.Component {
        constructor(props) {
          super(props);
          this.state = {
            name: this.props.name || 'Anonymous' //error is here we need to
replace this.props with props
          }
        }    render() {
          return (
            <p>Hello {this.state.name}</p>
          );
        }
}
```

9. **Can you explain the differences between all those ways. What happens when you click each of the buttons?**

```
class App extends React.Component {

constructor() {
super();
this.name = 'MyComponent';

this.handleClick2 = this.handleClick1.bind(this);
}

handleClick1() {
   alert(this.name);
}

handleClick3 = () => alert(this.name);
render() {
return (

<div>

<button onClick={this.handleClick1()}>click 1</button> //it throws an error as handleClick1() not returning any value
<button onClick={this.handleClick1}>click 2</button> //shows an alert
<button onClick={this.handleClick2}>click 3</button>
<button onClick={this.handleClick3}>click 4</button> //shows an alert

</div>
);
 }
 }
```

10. **What's the issue with this component. Why? How would you go about fixing it?**

```
class App extends React.Component {state = { search: '' }

handleChange = event => {/**
   * This is a simple implementation of a "debounce" function,
   * which will queue an expression to be called in 250ms and
   * cancel any pending queued expressions. This way we can
   * delay the call 250ms after the user has stoped typing.
   */
  clearTimeout(this.timeout);
  this.timeout = setTimeout(() => {
    this.setState({
      search: event.target.value
    })
  }, 250);
 }
render() {
   return (
    <div>
      <input type="text" onChange={this.handleChange} />
      {this.state.search ? <p>Search for: {this.state.search}</p> : null}
    </div>
   )
 }
}
```

11. **Why is props passed to the super() function in React?**
    if we want to use this in the constructor method ,we pass the props to super() function

12. **What are the benefits of HOC?**
    Higher order components takes a component as input and returns another component.
    the main benefit of using HOC is we can reuse the components easily.

13. **Why do we need a Router to React?**
    To create Navigation between different components in REACT

14. **Give an example of class component and functional component.**
    Before introduction of hooks functional components are called as stateless components
    after the introduction of hooks functional components are very similar to class components.

Functional component

It is like a normal function ,we can define it directly by using arrow functions or function keyword.

We can get the props directly from arguments given to the function

React hooks are used to store state and to use life cycle methods

Class component

it is defined using ES6 class.It extends the Component class in React.

this keyword is used.

we can use life cycle methods directly

15. **Explain the purpose of "render()" in React. Give an example.**

it returns the JSX code in class component.this method is re triggered when there is a change in state value

Node.js

1. **What's the Event Loop?**

2. **What are the advantages of using promises instead of callbacks?**
   Promises are used to handle asynchronous functions .It have different states Pending,fullfilled,rejected.
   we use resolve,reject method in promises.if the promise was resolved we can get data using then() method .if it is rejected we can error or response using catch() method

3. **How do you create a simple server in Node.js that returns Hello World?**

```
const http=require("http")
http.createServer((req,res)=>{
    res.write("Hello World)
    res.end()
}.listen(4000)
```

4. **Which of the following statements is valid to import a module in file?**

a. **var fs = require("fs");   answer**

b. var fs = import("fs");
c. package fs;
d. import fs;

5. **How to make node modules available externally?**

a. module.expose
b. module.spread
c. **module.export answer**

6. **Which of the following are Node.js stream types?**

a. Duplex
b. Readable
c. Writable
d. All of the above

7. What is the default scope of Node.js application?

a. Global
b. Local ANSWER
c. Public
d. Private

8. Which module is used to serve static files in Node.js?

a. static ANSWER
b. node-static
c. http
d. node-http

9. REPL stands for.

a. Research Eval Program Learn
b. Read Eval Print Loop   answer
c. Read Earn Point Learn
d. Read Eval Point Loop

10. Which of the following code print the platform of operating system?

a. console.log('platform : ' + os.platform);
b. console.log('platform : ' + os.platform());
c. console.log('platform : ' + os.getPlatform()); answer
d. None of the above.

MongoDB

1 What makes MongoDB the best?
MongoDb is a NoSQL database.It stores data in the form json documents.we can store/update/change data easily without writing lengthy queries.

2 How to do transactions/locking in MongoDB?

3 When and to what extent does data get extended to multi-slice?

4 Compare MongoDB with Couchbase and CouchbaseDB.

5 When do we use a namespace in MongoDB?

6 If you remove an object attribute, is it deleted from the database?

7 How can we move an old file into the moveChunk directory?

8 Explain the situation when an index does not fit into RAM.

9 How does MongoDB provide consistency?

10 Why is MongoDB not chosen for a 32-bit system?

11 How does Journaling work in MongoDB?

12 How can you isolate the cursors from intervening with the write operations?

13 Define MongoDB.

14 Explain the replica set.

15 What are the key features of MongoDB?

16 What is CRUD?
CREATE
READ
UPDATE
DELETE

17 What is Sharding?

18 What is Aggregation in MongoDB?

19 Define Namespace in MongoDB.

MySQL

1 How do you create a table using MySQL? Please write syntax.
CREATE TABLE nookesh(
        name varchar(100),
        age int);

2 How do you Insert Data Into MySQL? Please write syntax.
INSERT INTO nookesh(name,age)
VALUES
("raju",10);

3 How do you remove a column from a database?
ALTER TABLE nookesh
DROP COLUMN name;

4 How to add columns in MySQL?
ALTER TABLE nookesh
ADD address TEXT;

5 How to create an Index in MySQL?

6 How to Delete Data From a MySQL Table?
DELETE FROM nookesh
WHERE age=10;

7 How do you create and execute views in MySQL?
creating a view

CREATE VIEW myview AS(
SELECT name FROM nookesh
WHERE age=10)

execution

SELECT * FROM myview

8 What are the Numeric Data Types in MySQL?
INTEGER,DECIMAL

9 What are the Temporal Data Types in MySQL?
10 How do you create and execute views in MySQL?