

React Js

1. What is Reactjs?

A. React js is a javascript library, it was developed and maintained by facebook. Mainly using the reactjs we can develop the faster web applications compare to other technologies.

2. What is Dom?

A. Dom Means document object model(DOM)

It is a child object to the window, it is used to add dynamic content to the html pages, and also to add dynamic validations.

When html document is loaded in the browser it becomes a document object.

3. What is Real DOM?

A. Real dom is also called as a physical dom

B. It will work at the browser level

C. Every time it will create a memory instance

D. It will reload entire web application at a time

E. So consume more memory space but gives poor performance

4. What is Virtual DOM?

A. Virtual DOM is a logical dom

B. It will work at the react application level it self so it will occupy entire memory space.

C. It will check the previous state with new state in the component tree structure

D. If any modifications are there then only it will reload only updated component or state

E. So it will consume less memory but it will improve high performance.

5. What is react features?

A. React features are

1. Jsx(Javascript xml)

2. Components

3. One-way-data binding

4. Virtual dom

5. Simplycity

6. Performance

6.What is jsx?

A. Jsx is an javascript xml, it is an extention to the javascript

B. Jsx is nothing but a combination of javascript, html, xml.

Advantages of jsx

It will combine 3 codes,

Javascript, html, xml tages.

It will make the react application more eligent and simple

It is a faster than regular javascript

It makes easier to create templates.

7.what is component?

A :- Component is a piece of code or fundamental building block of a React application.

Every React application contains atleast one root component and n- no of child components.

All components will form like as Tree structure in the browser at the time of running a React application.

Every React component can have their own structure methods as well as API's

There are two types of Components:-

a. Functional Component

b. Class Component

a. Functional Component → Functional Components will look like a normal JavaScript Function.

it is also called as a stateless component because it is not maintain any internal state.

it is use only Dynamic Data Display

b. Class Component → Class Component will look like a OOP'S class but that should extend with React.component.

it is also called as a statefull component because it will maintain state with in the component by using this.state.

it maintains the this.state as well as props also

class components are more complex than functional components because it will follow components life cycle method

8.what is state ?

A:- state means need to maintain the data in with in the component. When ever using the class based component it is a statefull component because it will maintain state with in the component using this.state method .

whenever using the function based component it is called as a state less

component because it will not maintain the data within the component but, using the HOOK methods it will maintain the data in functional components using the useState method

9. what is PROP's?

A:- Props's is nothing but a when pass data from parent component to child component

10. How many ways we can pass data from one component to other components ?

A:- we can pass data in 3 ways 1.Prop's 2.callback function 3.context API

1.Prop's :- Props's is nothing but a when pass data from parent component to child component

2.Callback Function :- it means pass the data from the child component to parent component

3. Context API:- it will pass data from one component to any other component without any relationship. For example I have 20 components we need to pass data from 1st component to 20th component that time using the Context API, Context.Provider , Context.Consumer through that one we can pass the data

11. What is props drilling?

A. Props drilling means it will pass data from parent component to child components.

B. For example I have 20 components, need to pass data 1st component to last component (20th component), that time we can pass data one component to two component two to three component and three to four component any way we can pass 20th component. But if any component is not working means getting error it will not pass data nth components this is the drawback props drilling.

C. To overcome this problem we are using the context api or redux architecture.

12. What is Context Api?

A. Context api means it will pass data from one component to any other component without any relationship using the context.provider and context.consumer.

13. What is re-consolation?

A. Re-consolation in the sense it will update the only updated component.

14. Life cycle methods in react.?

A. There are 4 phases

1. Initial phase
2. Mounting phase
3. Updated phase
4. Unmounting phase

Initial phase : initial phase available only constructor method.

Mounting Phase : in mounting phase

1. Constructor
2. getDerivedStateFromProps
3. Render
4. componentDidMount

Updating Phase:

1. getDerivedStateFromProps
2. shouldComponentUpdate
3. Render
4. getSnapshotFromBeforeUpdate
5. componentDidUpdate

Unmounting phase

1. componentWillUnmount.

15. Functional based lifecycle method?

A. In function based there are 3 lifecycle methods

1. componentDidMount
2. componentDidUpdate
3. componentWillUnmount

ComponentDidMount:

Whenever using the useEffect method it will pass the 2 arguments 1st argument as a callback function 2nd argument as a empty dependency array. That time it will call as a componentDidMount.

```
Function App(){
```

```
  useEffect(()=>{
```

```
    Fetch('url');
```

```
  }, [] )
```

```
  Return(
```

```
    <div
```

```
    </div>
```

```
  )
```

```
}
```

componentDidUpdate: Whenever using the useEffect method it will pass the 2 arguments 1st argument as a callback function 2nd argument as a instead of empty dependency array it will pass state or props that time it will call as a component didUpdate Method.

```
Function App(){
```

```
  useEffect(()=>{
```

```
    Fetch('url');
```

```
  }, [data] )
```

```
  Return(
```

```
    <div
```

```
</div>
```

```
)
```

```
}
```

componentWillUnmount: inside useEffect method one return after one function. It is called as a cleanup operations like remove the DOM.

```
Function App(){
```

```
  useEffect(()=>{
```

```
    Fetch('url');
```

```
    Return function(){
```

```
      //code
```

```
    }
```

```
  }, [] )
```

```
  Return(
```

```
    <div
```

```
    </div>
```

```
  )
```

```
}
```

16. What is Hook Methods?

A. Hooks are nothing but a new features also called as a In-Built functions.

Hooks are introduced in React V16.8.0 version.

Hooks are backward compatible with old features of Reactjs which means it does not contain any breaking changes.

When to use Hooks:-

1. using hook methods we will maintain state in function component
2. Fetch and Consume data third party API's from server

3. We can get updated state from Redux store

HOOKS RULES :- we are having mainly two rules in Hooks.

A. call hook methods from react functions only

B. only call hooks at the top level of your react function i.e., don't call hooks inside loops, conditions or nested functions and class component

C. Don't call hooks from regular javascript functions

There are some hook methods :-

1. useState
2. useEffect
3. useMemo
4. useCallback
5. useRef
6. useReducer
7. useContext
8. useSelector
9. useDispatch
10. useStore

Details about Hook Methods:-

1. useState():- useState is a hook method which is used to manage state in functional component.

useState will take two arguments a. state variable

B. Setter method

Example :- `const [name, setName] = useState("veera");`

`const [city, setCity] = useState("atp");`

2. useEffect:- useEffect is a hook method it is equivalent to the 3 methods

1. componentDidMount

2.componentDidUpdate

3.componentWillUnmount

ComponentDidMount:

Whenever using the useEffect method it will pass the 2 arguments 1st argument as a callback function 2nd argument as an empty dependency array. That time it will call as a componentDidMount.

```
Function App(){
```

```
  useEffect(()=>{
```

```
    fetch('url');
```

```
  }, [] )
```

```
  return(
```

```
    <div
```

```
    </div>
```

```
  )
```

```
}
```

componentDidUpdate: Whenever using the useEffect method it will pass the 2 arguments 1st argument as a callback function 2nd argument as an array instead of an empty dependency array. It will pass state or props that time it will call as a componentDidUpdate Method.

```
Function App(){
```

```
  useEffect(()=>{
```

```
    fetch('url');
```

```
  }, [data] )
```

```
  return(
```

```
    <div
```

```
    </div>
```



```
)  
}
```

componentWillUnmount: inside useEffect method one return after one function. It is called as a cleanup operations like remove the DOM.

```
Function App(){  
  useEffect(()=>{  
    Fetch('url');  
    Return function(){  
      //code  
    }  
  }, [] )  
  Return(  
    <div  
    </div>  
  )  
}
```

3. **useMemo():-** useMemo is a hook method whenever using the useMemo it will return as a memorized value

Ex:- I have one component there is a one calculation part is there once application Re-render it will get directly value no need to calculate again and again.

4. **useCallback():-** useCallback is a hook method whenever using the useCallback it will return as a memorized function

Ex:- I have one parent component inside some child components once parent component state is updated it will Re-render the inside child components also need to avoiding that we are using the useCallback

5.useReducer():-useReducer is a hook method it receives a input and based on the dispatch action and value it will give us the modified updated state

Ex:- `const[state,dispatch]=useReducer(state,action){`

`Switchth(action.type){`

`//code //`

`}`

`}`

6.useRef():- useRef is a hook method it is mainly using the focus on the input field

7.useContext():- useContext is a hook method it is used to get and consume data in children elements without using redux.

Mainly using the useContext pass data from one component to any other component with out any relationship

17. Difference between React and Angular?

React	Angular
1.React is JavaScript library	1.Angular is JavaScript Framework
2.React follows one way data binding	2.Angular follows two way data binding
3.React follows View part only in MVC architecture	3.Angular follows MVC architecture
4.React application is fast as compared to Angular	4.Angular application is slow compared to React

18 difference between stateless and statefull components?

A:- stateless component ==> Function based component is called a stateless component because it will not maintain the state within the component.

if we need to maintain the state in function based component we need to use hook methods

Statefull component :- class based component called as a statefull component because it will maintain the state with in the component using the this.state method

19. what is setState?

A:- it is a one of the method it is available in component api it is used to update this.state object
by default setState method it follows asynchronous method

20. what is constructor?

A:- constructor is a special type of method it maintains 4things.

1. super 2.this.state 3.bind handlers 4.bind the reference attribute
2. It will acquire the data from parent component to child component

21. what is super?

A:- super is a method it will acquire the data from parent component to child component

22. what is return?

A:- return is a controlled statement used to return that statement to the calling place. Return keyword is helped to display the browser

23. what is event?

A:- event is a one of the action that will provide interaction between html tags to javascript

24. Difference B/w Controlled & uncontrolled Components

A. Controlled and uncontrolled components mainly using the forms creation

Controlled component it follows the 4 rules

1. This.state : Maintain the initial values
2. This.setState: Updated data input data in text fields

3. Value attribute: read the input data

4. onChange: it will take data when ever change

Controlled component working as a react application it self.

Uncontrolled Component:

Uncontrolled component it will not follow the 4 rules

It will works as a DOM (Document Object Model)

Mainly using the uncontrolled component focus on the input fields

25. What is React Fragment

A. React fragment means it will not create the Extra node

Example: we are creating div it will create the extra node instedof div we are writing the React.fragment

26. What is Pure Component

A. Pure component means it will stop the unnecessary re-renderings

Ex: we have one parent component inside some child components have

Once the parent component is updated it will Rerender the child components also, so we need to stop the re rendering inside child components.

27. What is React.Memo

A. React.Memo Means It will Stop the unnecessary re-renderings in functional based component.

Ex: we have one parent component inside some child components have

Once the parent component is updated it will Re-render the child components also, so we need to stop the re rendering inside child components.

28. what is SPA(Single Page Application) ?

A.SPA means Single Page Web Application it will create the only one HTML file inside one root id. Root id will collect the all components data and it gives that collected data to the browser it will displayed in UI.

29.React router how to install and how it's work ?

A. React Router we can't able to use directly in React. It is a Third party library we need to install librarie i.e.,

command:-npm i react-router-dom

Mainly we are using the React router dom is Navigation Purpose like one page to another page navigation.

Whenever creating the React router dom we need to use Browser router,Routes,Route

```
<BrowserRouter>
  <Routes>
    <Route path="/" element={<App />}>
    <Route path="Home" element={<Home />} />
    <Route path="About" element={<About />} />
  </Routes>
</BrowserRouter>
```

30.Difference b/w Axios and fetch method ?

Axios	Fetch
1.Need to install 3rd party libraries npm i axios	1.No need to install any 3rd party libraries
2.No need to convert the data in JSON format	2.Need to convert the data in JSON format
3.we can able to easily handle the Errors in Axios	3.it is very difficult to handle the Errors in Fetch

31.what is pure component ?

A.Pure component is used in only Class based component. To stop the Unnecessary Re-rendering

Ex:- I have one parent component inside some child components once parent component state is updated inside child components also Re-render need to

avoid that Re-rendering we can go for Pure componen

```
1  import React, {Component} from 'react'
2  import Normalcmp from './Normalcmp';
3  import Purecmp from './Purecmp';
4
5  class Purecomp extends Component {
6  constructor(){
7      super();
8      this.state={
9          name: "Ravi"
10     }
11 }
12
13 componentDidMount(){
14     setInterval(()=>{
15         this.setState({
16             name: "Raju"
17         })
18     })
19 }
20
```

```

23     render() {
24         return (
25             <div>Purecomp
26
27             <h2> {this.state.name}</h2>
28
29             <Normalcmp value={this.state.name} />
30             <Purecmp value={this.state.name}/>
31             </div>
32
33         )
34     }
35 }
36 export default Purecomp;
37

```

32. what is React.memo?

A. React.memo is used in only Function based component. To stop the Unnecessary Re-rendering

Ex:- I have one parent component inside some child components once parent component state is updated inside child components also Re-render need to avoid that Re-rendering we can go for React.memo

Ex :- export default React.memo (<About/>)

33. what is Higher Order Component (HOC)?

A. Higher order component is a special component it take input as a existing component return as a new component.

import React from 'react';

const withLogger = (WrappedComponent) => {

class WithLogger extends React.Component {

componentDidMount() {

```

    console.log('Component mounted');
  }

  componentWillUnmount() {
    console.log('Component unmounted');
  }

  render() {
    return <WrappedComponent {...this.props} />;
  }
}

return WithLogger;
};

// Usage

class MyComponent extends React.Component {
  render() {
    return <div>Hello, World!</div>;
  }
}

const MyComponentWithLogger = withLogger(MyComponent);

export default MyComponentWithLogger;

```

34. What is Redux and Redux Principles ?

A :- Redux is an architecture it is also called as predictable state container and it is also called as open source javaScript library

=> it is mainly used for the managing the whole application state in unidirectional.

=> it maintains global store. Which contains an updated state

Principles :- Predictability of Redux is determined by 3 most important

principles

1. Single source of truth.
2. State is Read only.
3. Changes are made with pure functions

Redux Architecture:- it is an architecture or predictable state container which is mainly used to manage whole application state in uni-directional flow.

It contains 5 main components :-

- 1.Action
- 2.Reducer
- 3.Store
- 4.Updated state
- 5.View components (or) UI Components

1.**Action :-**

=>Action is a pure JavaScript object or payload of the information

=> using Action.Creators method we will create an action object

=> using store.dispatch (action) method we will dispatch an action object

2.**Reducer :-**

=>Reducer is a pure JavaScript function

=>it will take previous state and dispatched action object as a two arguments

=> Based on the inputs it will update a new state

3.**State :- I**

=>if you want to get updated state from Redux store to your React components.

=>View components we have follow the 2 concepts

=>we have to useConnect() method to get updated state from redux store to view components

=> we have to use useSelector() and useDispatch () methods to get updated state from Redux store to view components

=> **Store:-**

For entire redux application we have to create single store that will work as a global store

=> we have to use React.createstore() method to maintain updated state.

=> store contains provider to make state is available to all the components in application.

View Components:-

=> view components are the Components here we will get or access from redux store to update UI.

Middlewares:

Redux Thunk and Redux saga

Redux thunk is a middleware, if you want to do asynchronous operations and if you want to get 3rd party API's we have to use Redux thunk.

Redux Saga is a middleware, if you want to do asynchronous operations and if you want to get 3rd party API's we have to use Redux Saga

Pure Function:

A function is a process which takes some input, called arguments, and produces some output called a return value.

=> a function is called pure if it follows below rules

A function returns same inputs always return some output

No side effects, I.e it does not alter input data.

No maintain of local & global variables.

1) Advantages of using ReactJS

- > Reusable components
- > Virtual DOM for efficient updates and rendering.
- > Good performance.
- > Strong developer for efficient community and support
- > Easy integration with other libraries and frameworks.
- > Can be used on the client and server side.

2) Limitations / Disadvantage of React?

- > React is a simple library and not a complete framework hence calls for dependencies.
- > Writing code is completed as it uses JSX and inline template formatting.

3) Event Pooling?

- > This means that the synthetic Event object would be reused and all properties would be nullified after the event handler has been called Event pooling.

4) Lazy loading: Lazy loading in React involves loading components only when they are needed, rather than loading all components upfront. This can improve performance by reducing the amount of data that needs to be loaded and processed.

5. What are synthetic events in React?

Synthetic events are the objects which act as a cross-browser wrapper around the browser's native event. They combine the behaviour of different browsers into one API. This is done to make sure that the events show consistent properties across different browsers.

6) How do you use forms and form validation in React.js?

- >Forms and form validation in React.js are typically implemented using
Controlled components, where the form input values are stored in the state
and updated as the user interacts with the form.
- >Form validation is then performed by checking the values in the state against
a set of rules.

7) Debugging is the process of identifying and resolving errors, or bugs, in a software system. It is an important aspect of software engineering because bugs can cause a software system to malfunction, and can lead to poor performance or incorrect results. Debugging can be a time-consuming and complex task, but it is essential for ensuring that a software system is functioning correctly.

8) State lifting refers to the practice of moving the state of a component upwards in the component tree to share it with other components. In React, each component manages its

own state, but sometimes you need to share state between components that are not directly related in the component tree.

9) How does Server-side rendering work in React.js?

>Server-side rendering in React.js involves rendering the initial HTML on the server, rather than in the browser.

>This can help improve performance, especially for slower devices or low-bandwidth connections.

10) Reusability

>When state is lifted up, lower-level components that need access to that state Can receive it as props. This makes it easier to reuse those components in different parts of the application, as they are not tightly coupled to the state they depend on.

11) Scalability:

>React built-in state management can become cumbersome in large or Complex applications, while Redux provides a more scalable solution.

12) Memoization

>Memoization in React involves caching the results of expensive computations so that they can be reused in the future, rather than recomputing the results each time. This can improve performance by reducing the amount of redundant computation.

13)How do you test React.js components?

>React.js components can be tested using various testing libraries, such as Jest and Enzyme.

>These libraries provide APIs for writing and running unit tests for React components.

Fetch method : The fetch() method in JS used to request to the server and load the information in the webpages. The request can be of any APIs that return the data of the format JSON or XML. This method returns a promise.

Axios library : With Axios, we can easily send asynchronous HTTP requests to REST APIs & perform create, read, update and delete options. Axios can be imported in plain JavaScript or with any library accordingly.

Javascript

1. What is javascript?

A. Javascript is a lightweight client-side as well as serverside Scripting Language that allows you to create the Dinamic webpages.

2. Javascript is a single thread or multi threaded language?

A. Javascript is a single threaded language

Single threaded means it will perform a single task at a time.

3. Javascript follows synchronous or asynchronous?

A. Javascript by default follows a synchronous mechanism. If you need to go for asynchronous we are using the callback or promises.

4. What are the data types?

A. In javascript there are two types of data types.

Primitive data type

Non primitive data type

Primitive data type means it will store only single value.

Example string, number, Boolean, null, undefined these are the primitive data types.

Ex: var a=10;

Var name='ismail';

Non Primitive Data types Means it will store the Multiple Values, Array, Object, function

Ex: var arr = [1,2,6,7,8,9,10]

```
Var obj={  
Name:"ismail",  
City:"anantapur"  
}
```

5. What is Dynamic validation?

A. Dynamic validation in the sense Checking the user entered data weather True or false

6. Var let const difference?

Var is a global scope key word, it will access through out the program.

Var only reassign the values, Var only supported the Hoisting.

Let : let is a functional scope variable , with in the function only accessable,

Let we can't re assign the values, let not supported the Hoisting.

const : const is a functional scope variable , with in the function only accessable,

Const we can't re assign the values, const is a constant, const not supported the Hoisting

7. Difference between == and === ?

A. == means it will compare only values

=== means it will compare the both values and both data type also

8. Difference between null and Undefined?

Null is an assignment value. It can be assigned to a variable as a representation of no value

Undefined in javascript undefined means a variable has been declared but has not yet been assigned a value

9. Type of Null ?

A. Object

10. What is Object?

A. Object Is a non primitive datatype

Object is a physical entity or real world entity or instance of a class that can have states and behaviour

It will work on keys and values Pair

11. How many ways we can create Object?

Objects creates in 3 ways

1. By using object literals
2. By using new keyword with parametarised object constructure
3. By using new keyword with non-parametarised object constructure

12.How We can copy one object to another Object?

Use the spread (...) syntax

Shallow copy : Use the Object.assign() method

Shallow copy means it will not disconnet from original object it will copy of new object

Deep Copy: Use the JSON.stringify() and JSON.parse() methods

Deep copy means it will disconnect from original object and it will copy of new object

13.what is Object.assign() and Object.create() ... ?

object assign is an object method which is used to copy all the values from one or more source object to a target objects

Object are assigned and copied by reference it will return the target object.

Object create

create() methord is used to create a new object with the specified prototype object and properties

14.what is the difference between Deep copy and shallow copy in JavaScript ... ?

A **deep copy** means that all of the values of the new variable are copied and disconnected from the original object

A **shallow copy** means that certain (sub-)values are still connected to the original Object

15.what is spread operator ... ?

A **Spread is a operator**

It is used to copy data from one object to another object or one array to another array. It is represented by {...} in curly braces.

```
Ex. Var arr = [1,2,3,4]
```

```
Var arr1=[5,6,7,8,9]
```

```
Var arr2 = [arr, arr1];
```

```
O/P => [1,2,3,4,5,6,7,8,9]
```

16.what is rest parameter ... ?

A rest is a parameter

Its Allows multiple parameters or arguments into an array

It is represented by (...) in parantasis brakets.

```
Function rest(a,b, ...rest){
```

```
  Console.log(rest)
```

```
}
```

```
Rest(1,2,3,4,5,6,7,8,9)
```

17.What is promises?

A. Promises represents the completion of an asynchronous operation with its results it can be either successful completion of the promise, or its failure, but eventually completed. Promise uses a then() which is executed only after the completion of the promise resolve

Code :

```
let payment = new Promise(function(resolve, reject)
```

```
{
```

```
  let send = false
```

```
  if(send)
```

```
  {
```

```
    resolve("payment is done");
```

```
  }
```



```

else
{
    reject("payment is cancel");
}
});

payment.then(( d ) => { console.log(d)} )
    .catch(( d ) => { console.log(d)} )

```

18.What is Closers?

Closers are created whenever a variable that is defined outside the current scope is accessed from within some inner scope

Code

```

function makeCounter () {
    var count = 0;
    return function () {
        count += 1;
        return count;
    }
}

var x = makeCounter();
console.log(x());
console.log(x())

```

19. Call Apply Bind ?

Call invokes the function immediately and allows you to pass in arguments one by one

Apply invokes the function immediately and allows you to pass in arguments as an array.

Bind returns a new function, and you can invoke/call it anytime you want by invoking a function.

Call:

```
var name1 = {firstName: "ravi", lastName: "kiran"};
var name2 = {firstName: "veera", lastName: "pasupula"};
function say(greeting) {
    console.log(greeting + ' ' + this.firstName + ' ' + this.lastName);
}
//say.call(name1, 'Hello');
say.call(name2, 'Hello');
```

apply:

```
var person1 = {firstName: 'Jon', lastName: 'Kuperman'};
var person2 = {firstName: 'Kelly', lastName: 'King'};
function say(greeting) {
    console.log(greeting + ' ' + this.firstName + ' ' + this.lastName);
}
say.apply(person1, ['Hello', "dfdf"]);
say.apply(person2, ['Hello']);
```

bind:

```
var person1 = {firstName: 'Jon', lastName: 'Kuperman'};
var person2 = {firstName: 'Kelly', lastName: 'King'};
function say() {
    console.log('Hello ' + this.firstName + ' ' + this.lastName);
}
var sayHelloJon = say.bind(person1);
var sayHelloKelly = say.bind(person2);
sayHelloJon();
```

```
sayHelloKelly();
```

20. What is Async Await ?

Async await both are the keywords which are used to make promises easier to write

Async and await keywords are used to perform asynchronous operations in javascript.

Async- makes a function return a promise,

Async function always returns a promises value.

```
const data = async ()=>
```

```
{
```

```
  var data = "This is Async"
```

```
  return data;
```

```
}
```

```
data().then(x=> console.log(x))
```

Await- makes a function wait for a promise,

It could be used within the async block only,

It makes the code wait until the promise return a result,

It only makes the async block wait

```
async function myfunction() {
```

```
  let myPromise = new Promise(function(resolve) {
```

```
    setTimeout(function() {resolve("Success");}, 3000);
```

```
  });
```

```
  let results = await myPromise;
```

```
  alert(results)
```

```
}
```

21. What is Object.freeze vs Object.Seal

	create	read	update	delete
object.freeze	No	Yes	No	No
object.seal	No	yes	Yes	No

22.What is Debouncing?

function will be executed only after the specified time once the user stops firing the event.

In the debouncing technique, no matter how many times the user fires the event, the attached function will be executed only after the specified time once the user stops firing the event.

ex : searching in the internet.

23.What is Throttling ?

executes the function at specified intervals.

Throttling is a technique in which, no matter how many times the user fires the event, the attached function will be executed only once in a given time interval.

ex : Making Api call is best.

24.Debouncing Vs Throttling?

The main difference between throttling and debouncing is that throttling executes the function at a regular interval, while debouncing executes the function only after some cooling period.

Both can be implemented with the help of the setTimeout web Api.

25.Event capturing and Event bubbling?

Event capturing is the event starts from top element to the target element.

It is the opposite of Event bubbling, which starts from target element to the top element.

26.What is event delegation in JavaScript?

Image result for event delegation in javascript

Event Delegation is basically a pattern to handle events efficiently. Instead of adding an event listener to each and every similar element, we can add an event

listener to a parent element and call an event on a particular target using the . target property of the event object

27.What is prototype?

prototype is a property of a Function object. It is the prototype of objects constructed by that function. __proto__ is an internal property of an object, pointing to its prototype.

28. Class in Javascript?

A Javascript class is a blueprint for creating objects. A class encapsulated data and functions that manipulate data.

29.Java script Hoisting?

When the program is running time it will move to the top of the scope, only Variable declaration part. Var only supported to the hoisting.

Ex: a=10;

Console.log(a);

Var a;

30.What is strict mode in javascript?

In javascript strict mode means it will through the silent errors while developing the Application;

“use strict”

A=10;

Console.log(a); // error

31.What are the scopes available in javascript?

In javascript 3 scopes

1. Global scope
2. Functional Scope
3. Block Scope

32.What is Function?

Function is a piece of code or fundamental building block which will execute certain task at a time.

2 ways we can create the Functions

Paramitarized function

Non Paramitarized function

33.What is call back function?

Callback function is a plain javascript function. It will pass the argument as a one more anonimus function it is called as a callback function.

34.What is Callback hell issue?

Callback hell means it will pass the auguments as a more call backs that time it will getting issue it is called as a callback hell issue

35.What is pure function?

A pure function in JavaScript is a function that returns the same result if the same arguments(input) are passed in the function.

36.What is Higher order Functions?

A function which takes another function as an argument or returns a function is known as a higher order function

Ex: map. Foreach, reduce, filter.

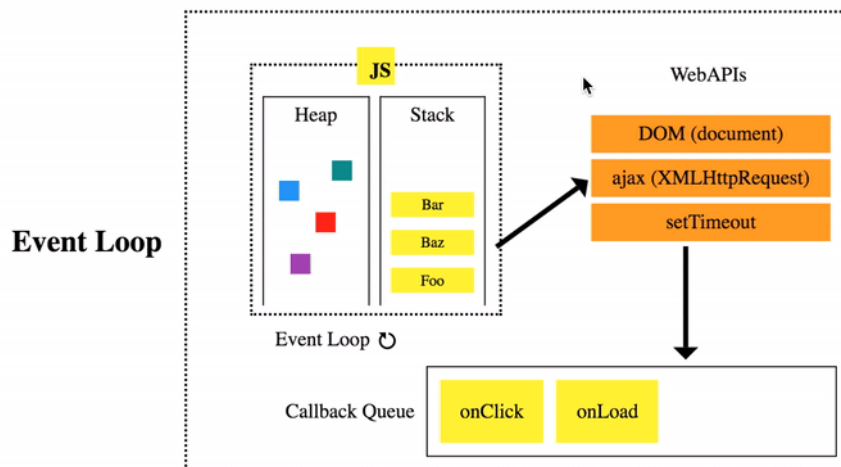
37.What is Event Loop?

Event loop is a mechanisum it will divide the 3 parts, heap memory, stack area, call stack area,

Heap memory in the sense it will storage part like `var a=10; var name="ramu"`

Stack area in the sense it will execution part normal execution part like `alert, console, document.writeln();`

Call stack means it will execution part but here only `settimeouts` and `promises` will execute: boath are same intervals it will first preference `promises`.



38.setTime out SetInterval?

setTime out means it will execute the only one time.

setInterval means it will execute every perticular time interval.

39.What is Template Literals?

Template literals are a new features introduced in Es6. it Provides an easy way of Creating Multiline string and Perform String interpolation.

```
Let str1 = "Hello";
```

```
Let str2 = "World";
```

```
Let str = `${str1} ${str2}`;
```

```
Console.log(str);
```

40.What is destructuring?

Destructuring is a new way to extract elements from an object

```
Let car = {
```

```
  Name:"bmw";
```

```
  Price:"1cr";
```

```
  Model:"BMW123N"
```

```
}
```

```
Const {name, price, model} = car;
```

41.What is carry function?

What is the purpose of currying in JavaScript? Currying in JavaScript transforms a function with multiple arguments into a nested series of functions, each taking a single argument

Code:

```
Function abc(a){
```

```
Return function(b){
```

```
Return function(c){
```

```
Return a+b+c;
```

```
}
```

```
}
```

```
}
```

```
Var res = abc(5)(6)(7);
```

```
Console.log(res);
```

42.What is this key word?

This key word represents the current function or current object.

43.What is local storage and session storage?

Local storage means when the browser is closed it will not expire the session.

Session storage means when the browser is closed it will expire the session.

44.Listed some of the es6 features ?

Let , const,

Arrow function

Spread, rest operators

Template literals

Destructring

Promises

Closers

Array methods

Map, foreach, reduce, filter, splice, slice etc.

45.Normal function Vs Arrow function?

Normal function means when ever creating the normal function keyword, function name curly braces {}, return key word and this key word need to use, while creating the arrow function no need to write function keyword curly braces {} return keyword and this keyword it will write the function expression in single line of code.

46.Difference between map and forEach?

Map and for each both are the es6 features, when ever using the map method it will extract the all elements in the given array, map it will return type as a new array.

forEach method it will extract the all elements in the given array, forEach method it will return type undefined.

47.Difference between splice and slice?

Splice means it will add remove the given array but it will modify the original array.

Slice means it will extract the particular index position elements but it will create the new array.

48.Difference between find and filter?

Find means it will find the particular value

Filter means it based on the condition it will return the elements

49.Generators in javascript?

Generator is the new concept introduced in Es6. Its provides you a new way of working with iterators and functions.

Es6 generator Is a different kind of function that may be paused in the middle either one or many times and can be resumed later

```
Function* mygenerator(){
```

```
Yield 1;
```

```
Yield 2;
```

```
};
```

HTML

1. What is doctype?

A. Doctype provides a version of html 5

2. What is meta tag?

A. Meta tag defines the Metadata about the html Document.

3. What is view port?

A. The viewport is **the user's visible area of a web page**. It varies with the device - it will be smaller on a mobile phone than on a computer screen. You should include the following <meta> element in all your web pages: <meta name="viewport" content="width=device-width, initial-scale=1.0">

4. Difference between inline and block tags?

A. Block elements cover space from left to right as far as it can go. Inline Elements only cover the space as bounded by the tags in the html element. Block elements have top and bottom margins. Inline Elements don't have top and bottom margin.

5. What is semantic tags?

A. The html semantics refers to the tags that provide meaning to an html page rather than just presentation. It makes html more comprehensible by better defining the different sections and layout of webpages.

Eg. header, footer, section, article, aside.

6. What is element and tag difference?

A. Html Tag : html tag starts with < and ends with >

B. Html element: whatever is written within a html tag is an html element.

7. What is attribute?

A. All html elements can have attributes

Attributes Provide Additional Information about element

Attributes are always specified in the start tag

Attributes usually come in Name/Value pairs like, Name="value".

8. What is web worker?

A. A web worker is a JavaScript that runs in the background, independently of other scripts, without affecting the performance of the page

9. Difference between html and html5?

Html:

1. Language in HTML does not have support for video and audio
2. The HTML browser uses cache memory as temporary storage.
3. In HTML5, vector graphics are not supported.
4. Uses cookies to store data.
5. It not supported the geolocation(MAP)

Html5:

1. **HTML5** supports both video and audio.
2. HTML5 has the storage options like: **application cache, SQL database, and web storage.**
3. In HTML5, vector graphics are supported by default.
4. Uses local storage instead of cookies
5. Html5 supports the geolocation(Map)
6. Drag and drop,

10. what is web storage?

A. Web storage in the sence local storage and session storage

Local Storage: when ever using the local storage when the browser is close it will not expire the session.

Session Storage: when ever using the Session storage when the browser is close it will expire the session.

CSS

1. What is Ccss?

- A. **CSS** stands for Cascading Style Sheet. It is a popular styling language which is used with **HTML** to design websites. It can also be used with any **XML** documents including plain **XML, SVG**.

2. What is Box Model?

The CSS box model is used to define the design and layout of elements of **CSS**.

The elements are:

- Margin - It removes the area around the border. It is transparent.
- Border - It represents the area around the padding
- Padding - It removes the area around the content. It is transparent.
- Content - It represents the content like text, images, etc.

3. How many ways to create a styles in css

- A. We can Create 3 ways
1. Inline Css --> it will write the with in the tag
 2. Internal Css -->it will write the head tag in top of the file
 3. External Css -- it will create the new file like style.css their it will write the css

4. What is position properties

A. positions properties

1. static ->Default value. Elements render in order, as they appear in the document flow

2.relative ->The element is positioned relative to its normal position, so "left:20px" adds 20 pixels to the element's LEFT

3.absolute-> The element is positioned relative to its first positioned (not static) ancestor element

4.sticky-> The element is positioned based on the user's scroll position

5.fixed ->The element is positioned relative to the browser window

5. What is flex and grid box

A. flex we can create single dimensional layout row wise or column wise

Properties are -> display:flex, flex-direction :row, flex-direction:column, flex-direction:row-reverse, flex-direction:column-reverse

grid -> it will create the two dimensional layout like row wise and column wise

6. Visibility hide and display none

A. **visibility hide** means it will not display the text but it will occupy space.

display none means it will not display the text as well as it will not occupy the space also.

7. Z-index

A. Z-index is a CSS property that defines the order of overlapping HTML elements. Elements with a higher index will be placed on top of elements with a lower index

8. Float Properties

A. Float properties are

1.float left 2.float right

Float left display the text left side

Float right display the text right side

9. Justify properties

A. Justify properties are

Justify-content: space-around

: space-between

: space-evenly

10. How to create a text in center horizontally and vertically

A. Horizontally and vertically center properties are

Display:flex;

Justify-content:center;

Align-items:center;

Height:100vh;

11.Psdo classes and psudo element

Psudo class are :hover, :active, :visited, :link;

Psudo elements are -> ex p::first-letter, p::first-line

P::before, p::after

12. What is media query?

A. Media queries are useful when you want to modify your site or app depending on a device's

13. Display inline vs display block difference

A. **Display Inline** : Display inline means it will not apply the height and width

Display inline block: it will apply the height and width but same row only display

Display Block: it will apply the height and width but it will display the new row

Bootstrap

1. What is Bootstrap?

A. Bootstrap is **a free front-end framework for faster and easier web development**. Bootstrap includes HTML and CSS based design templates for typography, forms, buttons, tables, navigation, modals, image carousels and many other, as well as optional JavaScript plugins.

Mainly we are using the bootstrap all screen devises responsive the web application.

Bootstrap it will divided in to 12 colums based on the requirements we will breack the colums like col-md-4 col-md-5 col-md-2

2. Whichverntion you are usingBootstrap?

A. we are using the bootstarp 4

3. what is break points in bootstrap.?

A. Bootstrap breakpoints are col-lg, col-xl, col-md, col-sm, col-4

4 .Container and container-fluid difference?

Containers are used to pad the content inside of them, and there are two container classes available:

1. The **.container** class provides a responsive **fixed width container**
2. The **.container-fluid** class provides a **full width container**, spanning the entire width of the viewport

5 .What is RWD(Responsive Web Design)

Responsive web design is an approach where a web designer develops a single web page that automatically resizes itself according to the screen's size

Must Know this concepts,

Container, container-fluid, buttons Color Codes, break points, Grid System, Carousel, tables, Alert, Badges, Progress Bar, Card, Collapse, Nav Bar, Forms, Model, Tabs.

azile methodology

every 2 weeks one sprint based on the ticked complexity assign the work

requirements getting the gira board once open the jira board full details about the ticket

if not clear the requirements ask to poc person (client person) requirements given by client

once clear the requirement proceeding the task once complete the task rise the PR(pull request) once internal team approve move to the

Merge to git dev branch

we are checking the dev brance all changes coming or not if all chenges came then move to QA (testing team)QA team pass the ticket move to UAT UAT client side testing do once done move to production in scrum meeting all team members (backend- front end -testing -devops)join and discuss what did yester day what to do today yester day any blockages is their inform to TL once scrum meeting completed based on the priority wise procede the tasks. end of the day one update meeting conduct only fe team we are all discuss completed task.

Here my roles and responsibilities are

1.creating the dash board purpose we are using the html, css, bootstrap, javascript.

2. api integration purpose we are using the axios/fetch librarys getting the data to to the back end to front end displaying in ui. based on the figma design we will display list, grid table format.

3 creating the sigle page web application purpose we are using the react-router-dom through that one creating the navigation menu one page to another page moving.

4. Entire application responsive purpose we are using the bootstarp.

through that one we will do all screen devises like mobile tablet desktop

view we can do the responsive.

5. based on the requirements we are using the state, props, context api,, callbacks passing the data from one component to another component.

6. creating the forms we are using the controlled and uncontrolled components.

7 entire application creating a single storage purpose we are using the redux architecture through that one we can create entire application only one single store.

8 Adding dynamic functionality by creating and dispatching action creators that deploy actions.

9 Optimizing animation logic with optimistic functions to increase rendering performance

health care project

Health care in the sence hospital domain like appolo kims hospital application here different modules like Health information, patient care, Book appointment,

Book health checkup, consult online, find hospital, Health record

hospital employes module , doctors information, patient information

doctor deportments. how many patiens still hospital discharged patients

all data getting data from backend restfull api we are displaying

front-end baseb on the figma design

E commerce application

i am working as a ecommerce application like amazon flipcart.

here i am developing as a products module and cart module.

routing concept also implemented.

inside product module, search, filter, left menu part.

when click read more display detail page and add to cart button click

add products to the cart page.

click the cart page what ever products added cart need to display the cart page.

check out page ask to the delivery address and payments methods like cash on delivery or

pay online after that need to proceed to payment.

after success message display like Your order confirm.