1)

How to manage the theme?

Which approach is better Figma design or Material UI

What are the things to keep in mind for creating multiple theme

How to use a chatbot to create a theme to overwrite the existing CSS

Previous Project role and responsibilities explanation

What are the different hooks?

Difference between useEffect, useMemo and useCallback

Given a string S containing alphanumeric characters, The task is to calculate the sum of all numbers present in the string.

Input:  1abc23   
Output: 24   
Explanation: 1 + 23 = 24

console.log(1)   
let promise= new Promise((resolve,reject)=>{   
         console.log(2)   
   setTimeout(() => {   
     resolve('foo');   
   }, 10);   
})   
console.log(3)   
promise   
.then(data=>{   
         console.log(4,data)   
})   
.catch(data=>{   
         console.log(5,data)   
})   
console.log(7)   
setTimeout(()=>{   
console.log(6)   
},0)

Program: Counting duplicate characters   
Input: “adsjfdsfsfjsdjfhacabcsbajda”   
Output: { a: 5, b: 2, c: 2, d: 4, f: 4, j: 4, s: 5 }

console.log(1)   
let promise= new Promise((resolve,reject)=>{   
   console.log(2)   
   setTimeout(() => {   
console.log(12)   
     resolve('foo');   
   }, 10);   
})   
console.log(3)   
promise   
.then(data=>{   
         console.log(4,data)   
})   
.catch(data=>{   
         console.log(5,data)   
})   
console.log(7)   
setTimeout(()=>{   
console.log(6)   
},0)

Create a function in JS which will check a string is using duplicate value or not, and return "duplicate found" or "duplicate not found"

Create a function which will give 3rd largest number from an array without sort method

Create a button and a modal, on click of button display modal, modal has button to close the modal, make the modal responsive using HTML, CSS, JS

Exposure on Vite and Webpack

How to send between two component

What is context

What is useMemo

Lifecycle Method in functional component

What is flexbox

why width in % and height in vh

Authentication in React

fetch VS axios

2)

 React Pearson – Arjun Balsubramanian - 15/04/2025

* Recent project in depth questions
* Team size, roles and responsibilities
* how do u stay updated with js
* How do u handle time zone differences between you and the client.
* when aria should be used in an attribute in html

**Uncomment inside function Wrapper to view the case one by one:**

import React, { useEffect, useRef, useState } from "react";

import { AddAPhotoOutlined } from "@material-ui/icons";

const books = [

  { id: 1, title: "Chemistry" },

  { id: 2, title: "History" },

  { id: 3, title: "Alchemy" },

  { id: 4, title: "Physics" },

  { id: 5, title: "Political Science" },

  { id: 6, title: "Media" },

];

function Wrapper() {

  // return <BookList />; // Question 1

  // return <Counter />; // Question 2

  // return <Button />; // Question 3

  // return <Input />; // Question 4

  // return <ApiCall />; // Question 5

  // return <BookDetail />; // Question 6

  return <Calculation />; // Question 7

}

// 1 - What is wrong in this example?

function BookList() {

  const [st, setSt] = useState();

  return (

    <ul>

      {books.map(({ title }) => (

        <li>{title}</li>

      ))}

    </ul>

  );

}

// 2 - What is wrong in this example?

function Counter() {

  const countRef = useRef();

  countRef.current = 0;

  return (

    <div>

      <span>{`count is ${countRef.current}`}</span>//answer : updated count doesnt reflect on ui so store it in state

      <div>

        <button onClick={() => (countRef.current += 1)}>increment</button>

      </div>

    </div>

  );

}

// 3 - What is wrong in this example?

function Button() {

  return (

    <div>//answer : unnecessary div

      <button onClick={() => {}}>//answer : add aria

        <AddAPhotoOutlined />

      </button>

    </div>

  );

}

// 4 - What is wrong in this example?

function Input() {

  return (

    <div

      style={{

        display: "flex",

        flexDirection: "column",

        alignItems: "flex-start",

      }}

    >

      <label>Enter your name</label>

      <input type="text"></input>//answer : label should point to the input

    </div>

  );

}

// 5 - What is wrong in this example?

function getBooks() {

  return books;

}

function ApiCall() {

  const [bookState, setBookState] = useState([]);

  useEffect(() => {

    const books = getBooks();

    setBookState(books);

  });//answer : no dependency array

  return (

    <ul>

      {bookState.map(({ title }) => (

        <li>{title}</li>//answer : no key

      ))}

    </ul>

  );

}

// 6 - What is wrong in this example?

function BookDetail() {

  const [bookState, setBookState] = useState([]);

  const books = getBooks();

  useEffect(() => {

    setBookState(books);

    const handleClick = () => {};

    document

      .querySelector("#removeBook")

      .addEventListener("click", handleClick);

  }, []);//answer : dependency array with books, cleanup function in useeffect

  return (

    <div>

      <h3>Book Detail</h3>

      <div>{bookState[0].title}</div>//answer : add optional chaining so that ui doesnt break

      <button id="removeBook">Remove book</button>

    </div>

  );

}

// 7 - How can this be optimized?

function calculation(num) {

  let result = 0;

  for (let i = 0; i < 100000; i++) {

    result += num \* i;

  }

  return result;

}

function Calculation() {

  const [count, setCount] = useState(0);

  const value = calculation(count);//answer : memo for expensive computations

  return (

    <div>

      <p>Count: {count}</p>

      <p>Calculated Value: {value}</p>

      <button onClick={() => setCount(count + 1)}>Increment Count</button>

    </div>  )

}

export default Wrapper;

3)

what is promise

Spread operator

what is use effect

what are hook and custom hooks

what is use memo and why we used that

why we use context and how we will pass data when their is nested component

what is fragments

have you familiar with graphql

have you familiar with material Ui

how you will display a todo list, provide syntax

have you worked with AWS, s3 bucket, lambda

the Interviewer showed me a page and asked me to explain how I would plan, start the development

4)

Explain about yourself and experience on UI.

What are the the challenges that I have faced in my previous projects.

What are CSS pseudo classes and pseudo elements.

What is accessibility and how to take care of accessibility issues?

How to test accessibility?

How to build Responsive Web Design and what are standard breakpoints?

How does JavaScript handle asynchronous function calls?

What is Microtask and Macro tasks

What is event loop in JavaScript

Questions on Use Memo, Use Callback.

How to handle performance issues in React JS

What will you consider while peer reviewing?

Differences between Next JS and React JS and advantages of Next JS over React JS.

Coding Question in React JS

const { useState } = React;

function App() {

  const [users] = useState([

    { id: 1, name: "Alice", active: true },

    { id: 2, name: "Bob", active: false },

    { id: 3, name: "Charlie", active: true },

    { id: 4, name: "Diana", active: false }

  ]);

const [isActive, setIsActive] = useState(true);

  return (

    <div style={{ padding: "20px" }}>

      <h1>User List</h1>

      <ol>

        <li>Only display "Active" users, then</li>

        <li>

          Display "Active" or "Inactive" users based on the dropdown selection

        </li>

      </ol>

      <label>

        Show:

        <select

          value={isActive ? "active" : "inactive"}

          onChange={(e) => null}

          style={{ marginLeft: "8px" }}

        >

          <option value="active">Active</option>

          <option value="inactive">Inactive</option>

        </select>

      </label>

      <ul>

        {users.map((user) => (

          <li key={[user.id](http://user.id/)}>

            {[user.name](http://user.name/)} {user.active ? "(active)" : "(inactive)"}

          </li>

        ))}

      </ul>

    </div>

  );

}

ReactDOM.render(<App />, document.getElementById("root"));

**Solution:**

const { useState } = React;

function App() {

  const [users] = useState([

    { id: 1, name: "Alice", active: true },

    { id: 2, name: "Bob", active: false },

    { id: 3, name: "Charlie", active: true },

    { id: 4, name: "Diana", active: false }

  ]);

  const [isActive, setIsActive] = useState(true);

  const handleChange = (e) =>{

    if(e.target.value == "active"){

      setIsActive(true)

    }

    else{

      setIsActive(false)

    }

  }

  return (

    <div style={{ padding: "20px" }}>

      <h1>User List</h1>

      <ol>

        <li>Only display "Active" users, then</li>

        <li>

          Display "Active" or "Inactive" users based on the dropdown selection

        </li>

      </ol>

      <label>

        Show:

        <select

          value={isActive ? "active" : "inactive"}

          onChange={handleChange}

          style={{ marginLeft: "8px" }}

        >

          <option value="active">Active</option>

          <option value="inactive">Inactive</option>

        </select>

      </label>

      <ul>

        {users.filter((item)=>item.active == isActive).map((user) => (

          <li key={[user.id](http://user.id/)}>

            {[user.name](http://user.name/)} {user.active ? "(active)" : "(inactive)"}

          </li>

        ))}

      </ul>

    </div>

  );

}

ReactDOM.render(<App />, document.getElementById("root"));

5)

1. Hooks, custom hooks - what are all hooks i have worked so far with the brief explaination.
2. Promises - when we will use promises and its purpose
3. Reactjs vs Typescript
4. GraphQL - any exposure in GraphQL
5. How much understanding on DB select Queries
6. Any idea on Creating the Rest API
7. Any Exposure on creating the Services using javascript
8. Challenges you faced in your previous projects and how did you handle it
9. javascript vs java
10. Is javascript Single threaded or multithreaded and why
11. Spread operator
12. Code splitting
13. How to fetch data from API and syntax of fetching the list in UI
14. While doing code review, what are the points you will mainly focus
15. Have you worked on micro front end
16. Have you worked on material UI

6)

1. What is JavaScript and OOPS?

2, What is Hooks?

3, What is useEffect ? Explain with one example?

4. What is JavaScript and java?

5, What is Redux explain briefly with examples?

6. Promises - when we will use promises?

7. JavaScript is Single threaded or multithreaded?

8, What is useContext explain briefly with examples?

9. While doing code review, what are the points you will mainly focus

10. What is call, apply, bind in JavaScript?

11. Challenges you faced in your previous projects and how did you handle?

12. The Interviewer showed me a page and asked me to explain how I would plan, start the development, create components, and determine the logic and functionality needed for that page.

7)

Previous project experience. Questions from previous projects mentioned in Resume

As a lead what difficulties I had face.

AWS experience (lambda)

State and props

Rest API’s

what are hook and custom hooks?

what is use Memo?

why we use context?

Do you know about Virtual Dom?

Material UI (theme component)

Explain about micro frontend architecture

JS usage, and explain about accessibility things

The Interviewer showed me a page and asked me to explain how I would plan, start the development.

Do you have any questions about the project?

8)

1. Introduce yourself, roles and responsibilities

2. Advantages of react

3. How authentication and authorization handled in your project

4. what is the value of this, when you click the button

<!DOCTYPE html>

<html>

  <head>

    <title>Hello, World!</title>

    <link rel="stylesheet" href="styles.css" />

  </head>

  <body>

     <button id="myWorld">Click Me</button>

      <script>

        console.log(this)         //window

        let id = document.getElementById("myWorld");

        id.addEventListener("click", function() {

          console.log(this)     //button

        })

      </script>

  </body>

</html>

5. what is the value of this on click of button in the below example

<!DOCTYPE html>

<html>

  <head>

    <title>Hello, World!</title>

    <link rel="stylesheet" href="styles.css" />

  </head>

  <body>

     <button id="myWorld">Click Me</button>

      <script>

        let id = document.getElementById("myWorld");

        id.addEventListener("click",() => {

          console.log(this) // window, because of arrow function

        })

      </script>

  </body>

</html>

6.Stop the setTimeout after count is 1

import React from 'react';

import { useState, useEffect } from 'react'

const App = () => {

  const [count, setCount] = useState(0);

  useEffect(() => {

    setTimeout(() => {

      setCount((count) => count + 1);

    }, 1000);

  }); // pass the empty[], so that useeffect will be stopped after first call

  return (

    <div>

      <div>

        <h1> {count}</h1>

      </div>

    </div>

  )

}

export default App

7. what are the different hooks you have used

8. what is useMemo

9. what is useCallback

10. what is useReducer

11. Why the count value is not updating on click of Ref count button

import React from 'react';

import { useState, useRef } from 'react'

function App() {

  const [count, setCount] = useState(0)

  const refCount = useRef(0);

  const styles = {

    main: {

      padding: '20px',

    },

    title: {

      color: '#5C6AC4'

    },

  };

  return (

    <div style={styles.main}>

      <h1 style={styles.title}>State Count {count}</h1>

      <h1 style={styles.title}>Ref Count {refCount.current}</h1>

      <div>

        <button onClick={() => setCount((count) => count + 1)}>

          State count

        </button>

        <button onClick={() => {refCount.current = refCount.current + 1 }}>

          Ref count

        </button>

      </div>

    </div>

  )

}

export default App

12. Why the below code gives error

import React from 'react';

import { useState, useRef } from 'react'

function App() {

  const [count, setCount] = useState(0)

  const refCount = useRef(0);

  const styles = {

    main: {

      padding: '20px',

    },

    title: {

      color: '#5C6AC4'

    },

  };

  return (

    <div style={styles.main}>

      <h1 style={styles.title}>State Count {count}</h1>

      <h1 style={styles.title}>Ref Count {refCount.current}</h1>

      <div>

        <button onClick={setCount((count) => count + 1)}>

          State count

        </button>

        <button onClick={() => {refCount.current = refCount.current + 1 }}>

          Ref count

        </button>

      </div>

    </div>

  )

}

export default App

13. What is event loop, microtask and macrotask queue

14. How you will make network call

15. What is the team size in your previous project

9)

* Recent project in depth questions.
* Team size, roles and responsibilities?
* How do u stay updated with the JS latest features?
* Have you used any AI in your project?

**Uncomment inside function Wrapper to view the case one by one:**

import React, { useEffect, useRef, useState } from "react";

import { AddAPhotoOutlined } from "@material-ui/icons";

const books = [

  { id: 1, title: "Chemistry" },

  { id: 2, title: "History" },

  { id: 3, title: "Alchemy" },

  { id: 4, title: "Physics" },

  { id: 5, title: "Political Science" },

  { id: 6, title: "Media" },

];

function Wrapper() {

  return <BookList />; // Question 1

  // return <Counter />; // Question 2

  // return <Button />; // Question 3

  // return <Input />; // Question 4

  // return <ApiCall />; // Question 5

  // return <BookDetail />; // Question 6

  // return <Calculation />; // Question 7

}

// 1 - What is wrong in this example?

function BookList() {

  const [st, setSt] = useState();

  return (

    <ul>

      {books.map(({ title }) => (

        <li>{title}</li>

      ))}

    </ul>

  );

}

// 2 - What is wrong in this example?

function Counter() {

  const countRef = useRef();

  countRef.current = 0;

  return (

    <div>

      <span>{`count is ${countRef.current}`}</span>

      <div>

        <button onClick={() => (countRef.current += 1)}>increment</button>

      </div>

    </div>

  );

}

// 3 - What is wrong in this example?

function Button() {

  return (

    <div>

      <button onClick={() => {}}>

        <AddAPhotoOutlined />

      </button>

    </div>

  );

}

// 4 - What is wrong in this example?

function Input() {

  return (

    <div

      style={{

        display: "flex",

        flexDirection: "column",

        alignItems: "flex-start",

      }}

    >

      <label>Enter your name</label>

      <input type="text"></input>

    </div>

  );

}

// 5 - What is wrong in this example?

function getBooks() {

  return books;

}

function ApiCall() {

  const [bookState, setBookState] = useState([]);

  useEffect(() => {

    const books = getBooks();

    setBookState(books);

  }, []);

  return (

    <ul>

      {bookState.map(({ title }) => (

        <li>{title}</li>

      ))}

    </ul>

  );

}

// 6 - What is wrong in this example?

function BookDetail() {

  const [bookState, setBookState] = useState([]);

  const books = getBooks();

  useEffect(() => {

    setBookState(books);

    const handleClick = () => {};

    document

      .querySelector("#removeBook")

      .addEventListener("click", handleClick);

  }, [books]);

  return (

    <div>

      <h3>Book Detail</h3>

      <div>{bookState[0].title}</div>

      <button id="removeBook">Remove book</button>

    </div>

  );

}

// 7 - How can this be optimized?

function calculation(num) {

  let result = 0;

  for (let i = 0; i < 100000; i++) {

    result += num \* i;

  }

  return result;

}

function Calculation() {

  const [count, setCount] = useState(0);

  const value = calculation(count);

  return (

    <div>

      <p>Count: {count}</p>

      <p>Calculated Value: {value}</p>

      <button onClick={() => setCount(count + 1)}>Increment Count</button>

    </div>

  );

}

export default Wrapper;

10) 1. How do you PR code review, what is the checklist.  
2. How will you achieve Responsive Web Design.  
3. Explain about Performance Optimization.  
4. Challenges faced in previous project and how we handled it.  
5. Explain about Promises and it’s use case.  
6. Explain about useMemo & useCallback.  
7. Idea about GraphQL.  
8. Shared few code snippets and asked to fix the mistake or missed line of code. 11)

1. How to improve performance in React applications? (usecallback/usememo/memo/code splitting)
2. Can we use Context API in place of Redux? When should we use Redux?
3. What is the use of Webpack?
4. Any exposure CI/CD pipeline
5. Any exposure to AWS services?
6. What is Mirco-frontend?
7. Code reusability in React (HOC/Custom hooks)
8. How to manage component life cycle methods in functional component?

12)

1. The interviewer first introduced himself and asked to give a brief introduction about me with the skillset and about recent/last project.
2. Challenges faced in previous projects, and how you overcome them.
3. What is useEffect hook and how you handle lifecycle methods and use cases where you need to use useEffect.
4. What is useMemo hook and where in your project you used useMemo/ give real world scenarios where it needs. Should we use useMemo everywhere if not then what is the drawback using it more.
5. What are the performance techniques you implemented in your project.
6. How you handle responsiveness in your project, what are the strategies you follow.
7. What are the CSS frameworks you have worked on and asked about material UI theming.
8. Name few CSS Pseudo Selector.
9. How you handle forms in ReactJs and what are the challenges faced while implementing forms.
10. Give one scenario where you had to take difficult decision in your project.
11. Have you implemented custom hooks in your projects and give examples where you have implemented.
12. Then Interviewer shared codepen link where based on some condition 3 implementation was there for filtering (1) Using for loop. 2) Using JS reduce method. 3) Using Set) among them choose which one better for readability and performance.
13. One more code pen link shared with ReactJs code with bug (useEffect dependency related)
14. Then Interviewer explained about the project and asked me if I have any questions to ask.

13)

1. Explain about the Projects worked
2. Can we use Context API in multiple updates of a value?
3. What are hooks you worked on? Explain?
4. What is usememo and usecallback?
5. What is the use of this in javascript?
6. Let, var and const
7. Graphql
8. What is difference between typescript and javascript
9. Why the count value is not updating on click of Ref count button

import React from 'react';

import { useState, useRef } from 'react'

function App() {

  const [count, setCount] = useState(0)

  const refCount = useRef(0);

  const styles = {

    main: {

      padding: '20px',

    },

    title: {

      color: '#5C6AC4'

    },

  };

  return (

    <div style={styles.main}>

      <h1 style={styles.title}>State Count {count}</h1>

      <h1 style={styles.title}>Ref Count {refCount.current}</h1>

      <div>

        <button onClick={() => setCount((count) => count + 1)}>

          State count

        </button>

        <button onClick={() => {refCount.current = refCount.current + 1 }}>

          Ref count

        </button>

      </div>

    </div>

  )

}

export default App

1. what is the value of this on click of button in the below example

1. Stop the setTimeout after count is 1

import React from 'react';

import { useState, useEffect } from 'react'

const App = () => {

  const [count, setCount] = useState(0);

  useEffect(() => {

    setTimeout(() => {

      setCount((count) => count + 1);

    }, 1000);

  }); // pass the empty[], so that useeffect will be stopped after first call

  return (

    <div>

      <div>

        <h1> {count}</h1>

      </div>

    </div>

  )

}

export default App

14)

* They began with questions on **state management** in React, including handling complex states and best practices.
* There was a strong focus on **performance optimization**—I was asked about tools I use for debugging and improving performance (e.g., Chrome DevTools, React Profiler), as well as strategies to optimize rendering and reduce unnecessary re-renders.
* They also assessed my experience with **unit testing using Jest**, including test coverage, mocking APIs, and writing test cases for components.
* A good portion of the interview covered **CSS**, including concepts like **Flexbox**, **responsive design**, **pseudo containers**, and tools used for responsive development.
* They presented a **code snippet with a bug in a counter function** and asked me to identify the issue, explain the root cause, and fix it.
* Another problem was around toggling **active/inactive states** and implementing **internationalization** support for the same.

15)

* Asked me to explain the last project.
* What is micro frontend?
* How to implement micro frontend? What are the mechanisms?
* What webpack module federation?
* What are the React hooks that you have worked on?
* What is the use case of useReducer hook?
* Difference between useState and useReducer?
* How to make asynchronous execution?
* How to handle asynchronous execution in useEffect hook?
* How to handle authentication?
* What is oktaAuth?
* Handle refresh token
* How to decode JWT?
* How to handle token at the frontend and at the backend?
* Where should we store the token at frontend?
* How can we handle the token at new tab opening?
* What is sessionStorage? How does it work?
* Gave some code chunks and asked the output and to fix the issues.
* State management?
* How to manage state with Redux?
* How to handle global state management with micro frontend?
* Why do we use GraphQL? Benefits of GraphQL?