JavaScript

Question: What is an array in JavaScript??

Answer: An **array** in JavaScript is a special type of object used to store multiple values in a single variable. Arrays can hold different types of data, including numbers, strings, objects, and even other arrays.

TypeScript

Question: What is Typescript and why would I use it in place of JavaScript

Answer: TypeScript is a **superset** of JavaScript that adds **static typing** and other features to help developers write better, more maintainable code. It was developed by **Microsoft** and compiles down to JavaScript so it can run anywhere JavaScript runs

HTML5

Question: What is the difference between an attribute and a property in HTML.

Answer: In **HTML**, an **attribute** is something that you define in the markup, while a **property** is what you access and manipulate using JavaScript.

Examples: <input id="myInput" type="text" value="Hello">

let inputElement = document.getElementById("myInput");

console.log(inputElement.value); // Output: "Hello"

CSS

Question: What existing CSS frameworks have you used locally or in production?? How would you change or improve them?

Answer: **Bootstrap** - Slow release cycle. Bootstrap 4 has been in alpha for almost 2 years. Add a spinner button component, as it is widely used.

React

Question: what are the major features of ReactJS?

Answer: The major features of ReactJS are as follows,

1. Component-Based Architecture

* React applications are built using components.
* Each component is reusable and independent, making code easier to manage.
* Components can be functional or class-based

**2. JSX (JavaScript XML) 📝**

* JSX allows you to write **HTML-like syntax** inside JavaScript.
* It makes the UI code more readable and expressive.

**3. Virtual DOM for Fast Performance ⚡**

* React uses a **Virtual DOM** to optimize rendering.
* When the state of a component changes, React **updates only the changed parts**, not the whole page.
* This improves **performance** and **user experience**.

✅ **How Virtual DOM Works:**

1. React creates a **copy of the real DOM** (Virtual DOM).
2. When a change happens, React **compares** the new Virtual DOM with the previous one (Diffing Algorithm).
3. Only the **changed parts** are updated in the real DOM.

import { useRef } from "react";

function TextInput() {

const inputRef = useRef(null);

const handleClick = () => {

inputRef.current.focus(); // Focus on the input field

inputRef.current.style.backgroundColor = "yellow"; // Change background color

};

return (

<div>

<input ref={inputRef} type="text" placeholder="Type here..." />

<button onClick={handleClick}>Focus Input</button>

</div>

);

}

export default TextInput;

React-hooks

Question: How do we access DOM elements in react??

Answer: In React, you generally don't **directly manipulate the DOM** like in vanilla JavaScript (document.getElementById() or document.querySelector()). Instead, React provides **Refs** and **event handlers** to access and interact with DOM elements

**Steps to Use Refs:**

1. Create a **ref** using useRef() (for functional components) or React.createRef() (for class components).
2. Attach the ref to a **DOM element** using the ref attribute.
3. Access or modify the DOM element inside **event handlers** or useEffect().

OOP

Question: What is the difference between procedural and object-oriented programming?

Answer: Procedural programming is based upon the modular approach in which the larger programs are broken into procedures. Each procedure is a set of instructions that are executed one after another. On the other hand, OOP is based upon objects. An object consists of various elements, such as methods and variables.

Redux

Question: What is Redux?

Answer: Redux is a **state management library** for JavaScript applications, commonly used with **React**. It helps manage the application state in a **centralized store** and ensures **predictable state updates**.

✅ **Why Use Redux?**

* Manages **complex application state** efficiently.
* **Avoids prop drilling** (passing data through multiple components).
* Makes **debugging easier** with time-travel debugging.
* Enables **global state management** across components.

Git

Question: What is the difference between a pull request and a branch?

* Answer: A **branch** is just a separate version of the code.
* A **pull request** (PR) is a request to **merge changes from one branch into another**. It allows team members to **review, discuss, and approve changes** before they are merged.

Web Security

Question: What is Vulnerability?

Answer: The **Vulnerability** can be defined as weakness of any system through which intruders or bugs can attack on the system.

If security testing has not been performed rigorously on the system then chances of vulnerabilities get increase. Time to time patches or fixes requires preventing a system from the vulnerabilities

Unit Testing

Question: What is mocking??

Answer: **Mocking** is the process of creating **fake objects, functions, or dependencies** in unit testing. It helps test a component **in isolation** by replacing **real dependencies** (like databases, APIs, or external services) with **mock versions**.

Agile and Scrum

Question: What is agile iteration?

Answer: An **Agile Iteration** is a **time-boxed development cycle** (typically 1-4 weeks) where a team completes a set of planned tasks, delivers a working product increment, and gathers feedback for continuous improvement.