

HTML

Q1: What are the differences between html4 and html5?

A: HTML 4 and HTML 5 are the most common and popular versions of HTML. Html4 does not support multimedia but Html5 supports multimedia. Html5 has a few more multiple new tags and simple features more than Html4. Html5 has web sockets for communication if the client and the server. On the other hand, Html4 has no web sockets, so the medium of communication between the server and the client is through streaming and long polling. Html4 has applet tags that have been replaced from Html5 by a new tag called `<object>`. Script attribute is important in Html4 but it is not mandatory for Html5. HTML4 is supported by many third-party elements like Silverlight and flash while html5 is supported by multimedia.

Q2: What are semantic tags in HTML? Give me some examples.

A: In programming, Semantics meaning of a piece of code. And this element clearly describes its meaning to both the browser and the developer. Some examples: `<article>`, `<aside>`, `<details>`, `<figure>`, `<footer>`, `<header>`, `<main>`, `<nav>`, `<section>`, `<summary>`, `<form>`, `<table>`

Q3: What is the purpose of Article, div, section, nav, aside?

A: `<article>` element represents a self-contained composition in a page or document or application. This element does not render anything special in a browser.

`<div>` tag is block-level tag. It is used to make divisions of content in the web page and create a particular section for particular data. This is easily styled by using the class or id attribute.

`<section>` element is a structural HTML element used to group together related elements. Each `<section>` typically includes one or more heading elements and additional elements presenting related content.

`<nav>` tag includes navigation links. It is intended only for major blocks of navigation links. So it provides in the current document or to another document.

`<aside>` is generally used to enhance an article with additional information or highlight parts that can be interesting to the user. This `<aside>` tag is new in HTML5. This tag does not render as anything special in a browser you have to use CSS for that

Q4: Why will you use Meta tag?

A: It defines metadata about an HTML document where metadata is information about data. And it is used to provide such additional information. With this Meta tag web, designers take control over the viewpoint.

Q5: What is the difference between inline, inline-block, and block?

A: Inline – The element doesn't start on a new line and only holds just the width it needs. And can't set the width or height.

Inline-block – It's formatted just like the inline element, where it doesn't start on a new line. But, can set width and height values.

Block – The element will start on a new line and occupy the full width available. And can set width and height values.

Q6: Difference between strong, b, bold, em, i?

A: **** means bold the text, **<i>** means make the text italics style, **** use to mean the text important. **** use for emphasized text.

Q7: What are properties and attributes in HTML?

A: Attributes give Html elements additional information.

Q8: What is a Viewport?

A: Viewport means the area of a web page which is user's visible. It varies with device, like a tablet, mobile phone, or computer.

Q9: Difference between link tag <link> and anchor tag <a>?

A: **<a>** is called anchor tag, used to create a hyperlink to another webpage or to a certain part of the webpage and these links are clickable. And link tag **<link>** defines a link between a document and an external resource but these are not clickable.

Q10: What are HTML Entities?

A: Some characters are reserved in HTML which is entities. It is a piece of text, or string, that begins with an ampersand (&) and ends with a semicolon (;). For example, less than (<) or (>) greater than sign are text but the browser might mix them with tags.

Q11: What is the difference between HTML elements and tags?

A: An element represents a kind of structure or semantics and generally consists of a start tag, content, and an end tag. On the other hand, tags are used to mark up the start and end of an HTML element.

Q12: What is a Style Sheet?

A: A style sheet is a command or set of commands that are used to control the layout of a web page, web page design. Use style sheet we can set the color, margin font size, etc on a web page.

Q13: What is the difference between DIV and SPAN in HTML?

A: **** its inline tag. It is used to group elements for styling purposes by using the class or id attributes. **<div>** tag is block-level tag. It is used to make divisions of content in the web page and create a particular section for particular data. This is easily styled by using the class or id attribute.

Q14: What is <figure> in HTML5?

A: The **<figure>** tag marks self-contained content, like diagrams, illustrations, photos, code listings, etc.

Q15: What is white space in HTML?

A: Whitespace is any string of text composed only of spaces, line breaks. It makes to code format and makes it easily readable by users.

CSS

Q1.What Flex layout? Difference Flex and grid layout?

A: Flexbox layout designed with one dimension, either a row or a column. But the Grid layout is designed with two dimensions, rows and columns designed at the same time. For small-scale layouts flexbox layout is suitable and for large-scale layout, suited grid layout.

Q2.Explain CSS position property? What are some differences between absolute position and relative position?

A: CSS position property set the position of an element in the document. It setting left, right, top, bottom and for this, it uses some methods like static, relative, absolute, fixed, or sticky.

The absolute position creates a position absolutely to its first parent position. And relative position creates a position relative to its normal position.

Q3.What is a box model? And what are the different elements of a box model?

A: The box model describes the document's layout and design. The elements of the box model are margins, borders, padding, and the actual content.

Q4.What is a Hover effect? What is the purpose of the active class?

A: When the mouse is moved over an element or link, it will be selected and this is the hover effect. It is usually used to link, unvisited, and visit pages, in order to be effective.

The active selector is used to select and style the active link, which becomes active when it is clicked.

Q5.What are the different types of Selectors in CSS?

A:

- Simple selectors (select elements based on name, id, class)
- Combinator selectors (select elements based on a specific relationship between them)
- Pseudo-class selectors (select elements based on a certain state)
- Pseudo-elements selectors (select and style a part of an element)
- Attribute selectors (select elements based on an attribute or attribute value)

Q6.What is CSS Specificity?

A: If two or more CSS rules are applied to the same element, then the browser determines it through a rule and its mean specificity.

Q7.What is a CSS Preprocessor? What are some benefits of Sass?

A: CSS processors add different logical syntax to simplify the functionality of vanilla CSS as seen in a normal programming language. Preprocessors take code written with this new and versatile syntax and then compile it into traditional CSS that the browser can work with.

Sass for Syntactically Awesome Stylesheet. Sass help to write clean, easy, and less CSS in programming and contains fewer codes can write CSS quicker. It provides nesting so can use nested syntax and useful functions like color manipulation, math functions, and other values.

Q8.What is a Pseudo element? What is pseudo-class?

A: A CSS pseudo-element is set the style of a specific part of the element which you selected.
A CSS pseudo-class is a keyword added to a selector that specifies a special state of the selected element.

Q9.How will you use media queries to make a website responsive?

A:It uses the @media rule to include a block of CSS properties only if a certain condition is true.
For example:If the windows is 639px or smaller,the background color will be orange.

```
@media only screen and (max-width: 639px) {  
  body {  
    background-color: orange;  
  }  
}
```

Q10.How will you make font size responsive?

A: Make font size responsive with a vw unit, which menaces the viewpoint width. Then the text size will follow the size of the browser window.

Again also make font size responsive with media queries.

Q11.What are gradients in CSS?

A: Gradients in CSS mean smooth transformation between two or more than two specific colors to set display. Linear Gradient and Radial Gradient are two types of gradients in CSS.

Q12.Differentiate between the ID and class??

A: The id can be used as an element because it can uniquely identify. So one element on the page should have a particular style applied to it. And class can be used more than one element. and it helps to set the same style more than one elements

Q13.What is the overflow property in CSS used for?

A: The overflow property specifies whether to clip the content or to add scrollbars when the content of an element is too big to fit in the specified area.

Q14.What does CSS selector mean?

A: A string equivalent of HTML elements by which declarations or a set of it, is declared and is a link that can be referred for linking HTML and Style sheet is CSS selector.

Q15. In CSS when you will use CSS float?

A: In CSS, I will use CSS float when I want to make an element of your page be pushed to the right or left and make other elements wrap around it.

Q16.What are the benefits of using CSS?

A: There are multiple benefits of using CSS, such as Accessibility, Page reformatting, Site-wide consistency, Bandwidth, Separation of the bandwidth of presentation

Q17.Which property controls the image scroll in the background?

A: Image scroll in the background can be controlled using the background-scroll property.

Q18.What is embedded style sheets?

A: These are the sheets where style sets for the entire HTML document are defined in a single place. For doing this, the style sheet information under the style tags should be embedded into an HTML document.

Q19.What is the difference between physical and logical tags?

A: Logical tags are older as compared to the physical ones and mainly focus on the content. They hardly find any usage in terms of presentation. Logical tags do not find any application in terms of aesthetics while the physical ones find their application in presentation too.

Q20.What are CSS counters?

A: These are variables that can be increased by using CSS that inspect and find the number of times of usage of variables.

JavaScript Part 1

Q1: Is JavaScript Single-threaded or multi-threaded?

A: JavaScript is a single-threaded language.

OR: Is JavaScript Synchronous or asynchronous?

A: JavaScript is synchronous.

Q2: How does JavaScript code is executed in Browser?

A: All in JavaScript is wrapped inside Execution Context, which is an abstract concept that holds the whole information about the environment within which the current JavaScript code is being executed.

Q3: What are the differences between “==” and “===”?

A: Double equal(==) test for abstract equality. It converts the variable value to the same before comparison. Triple equal(===) test for strict equality. It tests value and types both. If it's equal then its be true.

Q4: What is a callback function?

A: A callback is a function passed as an argument to another function. This technique allows a function to call another function. A callback function can run after another function has finished.

Q5: When will you return something from a function?

A: This is a default method in JavaScript. This method applies a method specifying the owner's object.

OR: How will you return more than one value from a function?

A: Using an array

Q6: Tell me about bind, call and apply.

A: The bind() method is reminiscent of call() and apply(). But instead of executing a function immediately, bind() returns a function that can be executed later on. The call () method invokes a function with a specific context. In other words, you can attach a function to an object as if it belonged to the object. The application () is similar to the method (). The difference is that the call () accepts a list of arguments, but the application () accepts an arrangement of arguments.

Q7: What is a Closure in JavaScript? How does it work?

A: A closure gives you access to an outer function's scope from an inner function. Global variables can be made local with closures. In JavaScript, closures are created every time a function is created, at function creation time.

Q8: What does the "this" keyword indicate in JavaScript?

A: 'this' points to a particular object, which is that object depends on how a function that includes the 'this' keyword is being called.

Q9: What is Event bubbling in js? Or How does event delegate work in JS?

A: Event bubbling is a method of event propagation in the HTML DOM API when an event is in an element inside another element, and both elements have registered a handle to that event. It is a process that starts with the element that triggered the event and then bubbles up to the containing elements in the hierarchy. In event bubbling, the event is first captured and handled by the innermost element and then propagated to outer elements.

Q10: Explain hoisting in JavaScript.

A: In JavaScript, Hoisting is the default behavior of moving all the declarations at the top of the scope before code execution. Basically, it gives us an advantage that no matter where functions and variables are declared, they are moved to the top of their scope regardless of whether their scope is global or local.

Q11: What is a recursive function?

A: A recursive function is a function in code that refers to itself for execution. Recursive functions can be simple or elaborate. They allow for more efficient code writing, for instance, in the listing or compiling of sets of numbers, strings, or other variables through a single reiterated process.

Q12: Difference between undefined and null

A: Undefined is a type that means a variable is declared but no value has been assigned a value. And null is an object which can assign to a variable

Q13:What are the different data types in JavaScript?

A: String, Number, Boolean, Undefined, Null are primitive data types in JavaScript. And Object, Array, RegExp are non-primitive data types in JavaScript.

Q14:What is DOM?

A: DOM for Document Object Model. It represents the page so that programs can change the document structure, style, and content. DOM can be modified with a scripting language like JavaScript.

Q15:Is JavaScript a static type or a dynamic type?

A: JavaScript is a dynamic type.

Q16:What is memoization?

A: Memoization is an optimization technique that speeds up applications by storing the results of expensive function calls and returning the cached result when the same inputs occur again.

Q17:What is the use of is NaN function?

A: Is Nan function returns true if the argument is not a number; otherwise, it is false.

Q18:What is negative Infinity?

A: Negative Infinity is a number in JavaScript that can be derived by dividing negative numbers by zero.

Q19:What is a prompt box?

A: A prompt box is a box that allows the user to enter input by providing a text box. A label and box will be provided to enter the text or number.

Q20:What is the working of timers in JavaScript?

A: Timers are used to execute a piece of code at a set time or repeat the code in a given interval. This is done by using the functions setTimeout, setInterval, and clearInterval.

Q21: Does JavaScript support automatic type conversion?

A: Yes, JavaScript does support automatic type conversion. It is the common way of type conversion used by JavaScript developers.

Q22:What do you mean by NULL in Javascript?

A: The NULL value is used to represent no value or no object. It implies no object or null string, no valid boolean value, no number, and no array object.

Q23: What is the use of Void (0)?

A: Void(0) is used to prevent the page from refreshing, and parameter “zero” is passed while calling. Void(0) is used to call another method without refreshing the page.

Q24: What are JavaScript Cookies?

A: Cookies are the small text files stored in a computer, and they get created when the user visits the websites to store information that they need. Examples could be User Name details and shopping cart information from previous visits.

Q25: What is break and continue statements?

A: Break statement exits from the current loop. Continue statement continues with the next statement of the loop.

JavaScript Part 2

Q1: What ES6 features did you use?

A: let and const Keywords, Arrow Functions, Multi-line Strings, Default Parameters, Template Literals, Destructuring Assignment, Classes, Modules

Q2: What are the differences between var, let, and const?

A: Var declarations are globally scoped or function scoped while let and const are block-scoped. Var variables can be updated and re-declared within their scope. Let variables can be updated but not re-declared, Const variables can neither be updated nor re-declared. They are all hoisted to the top of their scope. But while var variables are initialized with undefined, let and const variables are not initialized. While var and let can be declared without being initialized, const must be initialized during declaration.

Q3: Why will you use default parameters?

A: Default parameters also came along with ES6. It allows you to set default values for your function parameters if no value is passed or if undefined is passed. First, what happens when no parameter is passed to a function that requires parameters. We are going to define a simple function and call it without assigning variables to it.

Q4: How does the Spread operator work?

A: The spread operator is commonly used to make shallow copies of JS objects. Using this operator makes the code concise and enhances its readability.

Q5: Difference between class and object

A: Object is an instance of a class, while Class is a blueprint or template from which objects are created. The object is a real-world entity such as pen, laptop, mobile, bed, keyboard, mouse, chair, etc and Class is a group of similar objects. The object is a physical entity but the Class is a logical entity. There are many ways to create an object in java such as new keyword,

newInstance() method, clone() method, factory method, and deserialization while There is only one way to define a class in java using the class keyword.

Q6: What is a Prototype chain?

A: Each object has a private property that holds a link to another object called its prototype. That prototype object has a prototype of its own, and so on until an object is reached with null as its prototype. By definition, null has no prototype and acts as the final link in this prototype chain.

Q7: Explain Call by value vs call by reference

A: In the Call by value method original value is not modified whereas, in Call by reference method, the original value is modified. In Call by value, a copy of the variable is passed whereas in Call by reference, a variable itself is passed. In Call by value, actual and formal arguments will be created in different memory locations whereas in Call by reference, actual and formal arguments will be created in the same memory location.

Q8:What is the scope of JavaScript? Or Explain JavaScript scope, Block scope, and global scope?

A: Scope is the accessibility of variables, functions, and objects in some particular part of your code during runtime. In other words, scope determines the visibility of variables and other resources in areas of your code. Two types of scopes: Global Scope, Local Scope.

Q9: What is a Higher-order Function?

A: Higher-order functions are functions that take other functions as arguments or return functions as their results.

Q10: What is API? Difference between Get vs Post?

A: API stands for Application Programming Interface. A Web API is an application programming interface for the Web. A Browser API can extend the functionality of a web browser. A Server API can extend the functionality of a web server.

GET is used to request data from a specified resource. POST is used to send data to a server to create/update a resource. GET requests to remain in the browser history. POST requests do not remain in the browser history.

Q11: Difference between local storage and Session storage

A: Local storage stores a larger amount of data on the client's computer in a key-value pair format and has no expiration date. Data is never transferred to the server and is accessible via JavaScript and HTML5. Session storage stores a larger amount of data on the client's computer only for the current session, expiring the data on tab close. Data is never transferred to the server and is accessible client-side from the same tab.

Q12:What are cookies? And why will you use it?

A: Cookies are small text files stored in a web user's browser directory or data folder. Secure websites use cookies to validate a user's identity as they browse from page to page; without cookies, login credentials would have to be entered before every product is added to the cart or wish list.

Q13:What is object-oriented programming?

A: Object-oriented Programming treats data as a crucial element in program development and doesn't allow it to flow freely around the system. It ties data more securely to the function that operates on it and protects it from accidental modification from an outside function. OOP breaks down a problem into several entities called objects and builds data and functions around these objects.

Q14: Difference between Array vs LinkedList.

A: An array is a collection of elements of a similar data type. A linked list is a collection of objects known as a node where a node consists of two parts, i.e., data and address. Array elements stored in a contiguous memory location. Linked list elements can be stored anywhere in the memory or randomly stored.

Q15:How will you debug a JavaScript application?

A: All modern browsers have a built-in JavaScript debugger. Built-in debuggers can be turned on and off, forcing errors to be reported to the user. With a debugger, you can also set breakpoints, and examine variables while the code is executing.

Q16:What is Object Destructuring?

A: It is similar to array destructuring except that instead of values being pulled out of an array, the properties (or keys) and their corresponding values can be pulled out from an object. In object destructuring, the values are extracted by the keys instead of position.

Q17:What are generator functions?

A: A generator function is defined like a normal function, but whenever it needs to generate a value, it does so with the yield keyword rather than return. The yield statement suspends the function's execution and sends a value back to the caller, but retains enough state to enable the function to resume where it is left off.

Q18:What is the use of promises in javascript?

A: Promises are used to handle asynchronous operations in JavaScript. They are easy to manage when dealing with multiple asynchronous operations where callbacks can create callback hell leading to unmanageable code.

Q19: Is JavaScript a case-sensitive language?

A: Yes, JavaScript is a case-sensitive language. The language keywords, variables, function names, and any other identifiers must always be typed with a consistent capitalization of letters.

Q20:What is the difference between innerHTML & innerText?

A: innerHTML – It will process an HTML tag if found in a string. innerText – It will not process an HTML tag if found in a string

Q21: What are the syntax rules of JSON?

A: Data is in name/value pairs. Data is separated by commas. Curly braces hold objects. Square brackets hold arrays.

Q22:What is a rest parameter?

A: Rest parameter is an improved way to handle function parameter, allowing us to more easily handle various inputs as parameters in a function. The rest parameter syntax allows us to represent an indefinite number of arguments as an array.

Q23:What is Node.js?

A: Node.js is an open-source server-side runtime environment built on Chrome's V8 JavaScript engine. It provides an event-driven, non-blocking (asynchronous) I/O and cross-platform runtime environment for building highly scalable server-side applications using JavaScript.

Q24:What is TypeScript

A: TypeScript is a superset of JavaScript. TypeScript builds on top of JavaScript. First, you write the TypeScript code. Then, you compile the TypeScript code into plain JavaScript code using a TypeScript compiler.

Q25:What is Babel?

A: Babel is a toolchain that is mainly used to convert ECMAScript 2015+ code into a backward-compatible version of JavaScript in current and older browsers or environments.

REACT JS

Q1:What is react js? Tell us about the advantages and disadvantages of using react js.

A: React is a JavaScript library created for building fast and interactive user interfaces for web and mobile applications. It is an open-source, component-based, front-end library responsible only for the application's view layer.

Advantages of ReactJS: 1. Easy to Learn and Use.2. Creating Dynamic Web Applications Becomes Easier.3. Reusable Components.4. Performance Enhancement.5. The Support of Handy Tools.

The disadvantage of ReactJS:1. The high pace of development.2. Poor Documentation.3. View Part.4. JSX as a barrier

Q2:What is JSX? How does it work

A: It is a syntax extension to JavaScript. We recommend using it with React to describe what the UI should look like. JSX may remind you of a template language, but it comes with the full power of JavaScript.

Q3:What is Virtual dom? What are the differences between virtual and real dom?

A: DOM is a language-neutral interface allowing programs and scripts to dynamically access and update multiple objects like content, structure, and style of a document. Is a collection of modules designed to provide a declarative way to represent the DOM for an application. The DOM represents the document as nodes and objects. A virtual DOM object is a representation of a DOM object, like a lightweight copy.

Q4:Differences between props and state?

A: The Data is passed from one component to another. The Data is passed within the component only. It is Immutable. It is Mutable. Props can be used with state and functional components. State can be used only with the state components/class component. Props are read-only. State is both read and write.

Q5:What is the purpose of useState? When and why will you use it?

A: The useState function is a built-in hook that can be imported from the react package. It allows you to add state to your functional components. Using the useState hook inside a function component, you can create a piece of state without switching to class components.

In simple terms, doesn't matter if it's a class component or a functional component with a hook, you'll only need to use state if it requires dynamically manipulating/controlling something within the DOM.

Q6: What is a context API? How does it work?

A: The React Context API is a way for a React app to effectively produce global variables that can be passed around. This is the alternative to "prop drilling" or moving props from grandparent to child to parent, and so on. `react.createContext()` is all you need. It returns a consumer and a provider. The provider is a component that as its name suggests provides the state to its children. It will hold the "store" and be the parent of all the components that might need that store. Consumer as it so happens is a component that consumes and uses the state.

Q7:Difference between useEffect and useState?

A: `useState()` is a hook function which enable developers to use State in functional component. `useEffect()` is a hook function which enable developers to use group of lifecycle method like `componentDidMount ()` , `componentDidUpdate()` , `componentWillUnmount()` in functional component.

Q8:What other hooks have you used other than useState and useEffect

A: `useRef`, `useContext`.

Q9:Tell us about React Component lifecycle

A: Each component in React has a lifecycle that you can monitor and manipulate during its three main phases. The three phases are Mounting, Updating, and Unmounting.

Q10:What is the purpose of a custom hook? How will you create a custom hook? Give us an example.

A: Custom hooks hold a certain logic that makes use of React's hooks like `useState`, `useEffect`, etc. Usually create custom hooks when a certain part in your project is reusable and makes use of React's hooks. So, create a custom hook that can use throughout a project just like using React's hooks.

Q11: What is the most challenging task you have accomplished using react?

- A:**
1. Not Starting a Component Name with a Capital Letter
 2. Using Single Quotes Instead of Back-ticks
 3. Using `React.PropTypes`
 4. Not Using the Right Versions of What a Tutorial is Using

5. Confusing Functions with Classes

6. Passing Numbers as Strings

Q12: What is Redux and its uses?

A: Redux is a predictable state container for JavaScript apps. It helps you write applications that behave consistently, run in different environments (client, server, and native), and are easy to test. Redux allows you to manage your app's state in a single place and keep changes in your app more predictable and traceable. It makes it easier to reason about changes occurring in your app. But all of these benefits come with tradeoffs and constraints. One might feel it adds up boilerplate code, making simple things a little overwhelming; but that depends upon the architecture decisions.

Q13: Do you know about React native?

A: Little bit.

Q14: What is a Higher-order component? Give us an example.

A: A higher-order component is a function that takes a component and returns a new component. A higher-order component (HOC) is the advanced technique in React.js for reusing component logic. Higher-Order Components are not part of the React API. They are the pattern that emerges from React's compositional nature. The component transforms props into UI, and a higher-order component converts a component into another component.

Q15: How would you optimize a react js application?

- A:**
1. Using Immutable Data Structures
 2. Function/Stateless Components and React
 3. Multiple Chunk Files
 4. Use React
 5. Avoid Inline Function Definition in the Render Function.
 6. Throttling and Debouncing Event Action in JavaScript.
 7. Avoid using Index as Key for map.

Q16: What is the virtual DOM?

A: virtual DOM is a concept where a virtual representation of the real DOM is kept inside the memory and is synced with the real DOM by a library such as ReactDOM.

Q17: What is React Router?

A: React Router refers to the standard library used for routing in React. It permits us for building a single-page web application in React with navigation without even refreshing the page when the user navigates. It also allows to change the browser URL and will keep the user interface in sync with the URL. React Router will make use of the component structure for calling the components, using which appropriate information can be shown.

Q18: What is children prop?

A: Children is a prop (`this.props.children`) that allows you to pass components as data to other components, just like any other prop you use. Component tree put between component's opening and closing tag will be passed to that component as children prop.

Q19: What are fragments?

A: It's a common pattern in React which is used for a component to return multiple elements. Fragments let you group a list of children without adding extra nodes to the DOM.

Q20:What are the advantages of React?

A: Below are the list of main advantages of React,
Increases the application's performance with Virtual DOM.
JSX makes code easy to read and write.
It renders both on the client and server-side (SSR).
Easy to integrate with frameworks (Angular, Backbone) since it is only a view library.
Easy to write unit and integration tests with tools such as Jest.

Q21:How to use innerHTML in React?

A:The dangerouslySetInnerHTML attribute is React's replacement for using innerHTML in the browser DOM. Just like innerHTML, it is risky to use this attribute considering cross-site scripting (XSS) attacks. You just need to pass a __html object as key and HTML text as value.

Q22:How events are different in React?

A: Handling events in React elements has some syntactic differences:
React event handlers are named using camelCase, rather than lowercase.
With JSX you pass a function as the event handler, rather than a string.

Q23:What is Jest?

A: Jest is a JavaScript unit testing framework created by Facebook based on Jasmine and provides automated mock creation and a js dom environment. It's often used for testing components.

Q24:What are the features of React?

A: JSX, Components, One-way Data Binding, Virtual DOM, Simplicity, Performance

Q25:Explain the purpose of render() in React.

A: It is mandatory for each React component to have a render() function. Render function is used to return the HTML which you want to display in a component. If you need to rendered more than one HTML element, you need to grouped together inside single enclosing tag (parent tag) such as <div>, <form>, <group> etc. This function returns the same result each time it is invoked.

Node & MongoDB

Q1:What is Nodejs? Difference between Nodejs and JavaScript?

A: Node.js is an open-source, cross-platform runtime environment for developing server-side and networking applications. Node.js applications are written in JavaScript and can be run within the Node.js runtime on OS X, Microsoft Windows, and Linux. Node.js also provides a rich library of various JavaScript modules which simplifies the development of web applications using Node.js to a great extent.

Different between Node Js and JavaScript:

Javascript is a programming language that is used for writing scripts on a website. NodeJS is a Javascript runtime environment. Javascript can only be run in browsers. It is basically used on the client-side. We can run Javascript outside the browser with the help of NodeJS. It is mostly used on the server-side. Javascript is capable enough to add HTML and play with the DOM. Nodejs does not have the capability to add HTML tags. Javascript is used in frontend development. Nodejs is used in server-side development.

Or is Node js blocking or non-blocking?

A: non-blocking

Q2:Why did you use Node and Mongo with your React project?

A: It's because of two main reasons: Using an NPM (Node Package Manager), Node works alongside the NPM registry to easily install any package through the NPM CLI. Node bundles a React application into a single file for easy compilation using a web pack and several other Node modules. MongoDB is built on a scale-out architecture that has become popular with developers of all kinds for developing scalable applications with evolving data schemas. MongoDB was built for people building internet and business applications who need to evolve quickly and scale elegantly. Companies and development teams of all sizes use MongoDB for a wide variety of reasons.

Q3:What are the differences between SQL and NoSQL database

A: **SQL** databases are primarily called Relational Databases (RDBMS); whereas **NoSQL** databases are primarily called non-relational or distributed databases. **SQL** databases define and manipulate data-based structured query language (SQL). Seeing from a side this language is extremely powerful. **SQL** is one of the most versatile and widely-used options available which makes it a safe choice, especially for great complex queries. **A NoSQL** database has a dynamic schema for unstructured data. Data is stored in many ways which means it can be document-oriented, column-oriented, graph-based, or organized as a KeyValue store.

Q4:What have you done with MongoDB?

A: I have done with MongoDB for storage my data

Q5: Have you worked on website hosting?

A: Yes, I use it for hoisting Netlify and Firebase.

5 Common Q&A in Interview

1. About myself:

I am Nishat Tasnim Nahid. I have just over a half year of experience as a junior web

Developer. In this short time, I have completed some projects. I used HTML, CSS, Bootstrap, Material UI, React.js, JavaScript, Node.js, JSON, Rest API to finish these projects. And I've worked with MongoDB, Firebase, Figma, GitHub too. I like HTML, CSS, Tailwind, Material UI, Netlify, and I feel very comfortable working with those. I want to be a good web developer.

It's my pleasure to introduce myself in front of you. This is Nishat Tasnim Nahid. I am studying Honours in Statistics. Talking about my strengths & skills, I am innovative, hardworking and a quick learner. I can work under pressure. I am good at communication skills which enable me to convince people in my favor through logical arguments. I am a positive thinker. My hobbies are learning, reading.

As I have just started my career as a web developer, I have done some projects. Where I use JavaScript, React, Node Js, HTML, CSS, Bootstrap, Material UI, and for hosting I use Netlify, Firebase. And for datastore use MongoDB.

But I want to practically explore and experience my potential by delivering the best services to your company. & insha'Allah I will prove myself once I have the opportunity.

That's all about myself, sir and madam. If you/anyone want to know anything else about me, I would be glad to answer you. Agin Thanks for hearing me.

2. My Best Project:

My best project is BABY TOY. In this project, I have arranged for a user to register and log in system. And I have also created the admin system in this project to be able to create new admins and manage orders. If the user wants, he can delete or edit his product this feature is also here. All the information on this project is stored in MongoDB. I had to face some challenges in creating admins here and allowing users to delete or edit their own data and add this project information to MongoDB.

3. Why should you hire me:

I think I have almost all the skills you need to work in your company. And I will try to give my 100% effort in case of work. Even though I am new to this, I will try my best to work and I will do my best. I want the company to benefit from me. I have the skills to work tirelessly and good communication ability which I can use while working in your company. And that's why I think you will hire me in your company.

4. Salary Expectation:

I want to learn better and to work on better projects at the beginning of my career. And I hope you will give me a competitive salary.

5. My question for them:

- If I were hired for this role, what would you want me to achieve in my first month of the job?
- How do you ensure that your workers are constantly learning new things?

- How do you see the candidates here and what is your working environment like?
- Do you think that there are some shortcomings in my qualifications according to your needs?
- What communication methods are most commonly used in the workplace?