# Qustions Answers

What is JavaScript?

￼A high-level, interpreted programming language called JavaScript makes it possible to create interactive web pages and online apps with dynamic functionality. Commonly referred to as the universal language, Javascript is primarily used by developers for front-end and back-end work.

What are the different data types in JavaScript?

Number, String, Boolean, Null, Undefined, Symbol

It also has two compound data types: Object

What is the difference between null and undefined?

null is an assignment value that represents no value or an empty value, while undefined is a variable that has been declared but not assigned a value.

What is the purpose of the “this” keyword in JavaScript?

￼The this keyword refers to the object that is executing the current function or method. It allows access to object properties and methods within the context of that object.

What is the difference between == and === operators in JavaScript The equality == operator is a comparison operator that compares two values and returns true if they are equal. The strict equality === operator is also a comparison operator, but it compares two values and returns true only if they are equal and of the same type.

The var and let keywords are both used to declare variables in JavaScript. However, there are some key differences between the two keywords.

What is the difference between “var” and “let” keywords in JavaScri The var keyword declares a global variable, which means that the variable can be accessed from anywhere in the code. The let keyword declares a local variable, which means that the variable can only be accessed within the block of code where it is declared.

10. What is event delegation in JavaScript? Event delegation is a technique where you attach a single event listener to a parent element, and that event listener handles events occurring on its child elements. It helps optimize performance and reduce memory consumption. What is the difference between “let”, “const”, and “var”? let and const were introduced in ES6 and have block scope. let is reassignable, and const is non-reassignable. var is function-scoped and can be redeclared and reassigned throughout the function.

Implicit type coercion is a JavaScript concept that refers to the process of converting a value from one type to another. If you try to add a string to a number, JavaScript will implicitly coerce the string to a number before performing the addition operation.

What is implicit type coercion in JavaScript?

For example, the following code will add the string "10" to the number 5. This is because JavaScript will implicitly coerce the string "10" to a number before performing the addition operation.

How can you clone an object in JavaScript? There are multiple ways to clone an object in JavaScript. One common method is using the Object.assign() method or the spread operator (...).

What is the difference between function declarations and function e Function declarations are defined using the function keyword, while function expressions are defined by assigning a function to a variable. Function declarations are hoisted, while function expressions are not.

Syntax errors: These errors occur when the JavaScript code is not syntactically correct.

Runtime errors: These errors occur when the JavaScript code is executed and there is a problem.

What are the different types of errors in JavaScript?

Logical errors: These errors occur when the JavaScript code does not do what it is supposed to do.

What is recursion in JavaScript? Recursion is a programming technique that allows a function to call itself. Recursion can be used to solve a variety of problems, such as finding the factorial of a number or calculating the Fibonacci sequence A callback function is a function passed as an argument to another function, which is then invoked inside the outer function. It allows asynchronous or event-driven programming.

What is a callback function in JavaScript?

What are promises in JavaScript? Promises are objects used for asynchronous operations. They represent the eventual completion or failure of an asynchronous operation and allow chaining and handling of success or error cases.

What is the difference between synchronous and asynchronous pro In synchronous programming, the program execution occurs sequentially, and each statement blocks the execution until it is completed. In asynchronous programming, multiple tasks can be executed concurrently, and the program doesn’t wait for a task to finish before moving to the next one. How do you handle errors in JavaScript? Errors in JavaScript can be handled using try-catch blocks. The try block contains the code that may throw an error, and the catch block handles the error and provides an alternative execution path.

What is the difference between querySelector and getElementById querySelector is a more versatile method that allows you to select elements using CSS-like selectors, while getElementById specifically selects an element with the specified ID. What is the purpose of the setTimeout() function in JavaScript? The setTimeout() function is used to delay the execution of a function or the evaluation of an expression after a specified amount of time in milliseconds.

What is event delegation and why is it useful? Event delegation is a technique where you attach a single event listener to a parent element to handle events occurring on its child elements. It’s useful for dynamically created elements or when you have a large number of elements How can you prevent the default behavior of an event in JavaScript You can use the preventDefault() method on the event object within an event handler to prevent the default behavior associated with that event.

Both localStorage and sessionStorage are web storage objects in JavaScript, but they have different scopes and lifetimes.

localStorage persists data even after the browser window is closed and is accessible across different browser tabs/windows of the same origin.

What is the difference between localStorage and sessionStorage in sessionStorage stores data for a single browser session and is accessible only within the same tab or window How can you convert a string to lowercase in JavaScript? You can use the toLowerCase() method to convert a string to lowercase in JavaScript.

What is the purpose of the map() function in JavaScript? The map() function is used to iterate over an array and apply a transformation or computation on each element. It returns a new array with the results of the transformation. splice() is used to modify an array by adding, removing, or replacing elements at a specific position.

What is the difference between splice() and slice()?

slice() is used to create a new array that contains a portion of an existing array, specified by the starting and ending indices.

An array is a data structure that can store a collection of values. An object is a data structure that can store a collection of properties.

What is the difference between an array and an object in JavaScrip Arrays are indexed by numbers. Objects are indexed by strings. Arrays can only store primitive data types and objects. Objects can store primitive data types, objects and arrays. How can you remove duplicates from an array in JavaScript? One way to remove duplicates from an array is by using the Set object or by using the filter() method with the indexOf() method.

What is the purpose of the fetch() function in JavaScript? The fetch() function is used to make asynchronous HTTP requests in JavaScript. It returns a Promise that resolves to the response from the server.

There are many different events in JavaScript, but some of the most common events include:

What are the different events in JavaScript?

Click: The click event occurs when a user clicks on an HTML element.

Mouseover: The mouseover event occurs when a user's mouse pointer moves over an HTML element. Keydown: The keydown event occurs when a user presses a key on the keyboard.

Keyup: The keyup event occurs when a user releases a key on the keyboard.

Change: The change event occurs when a user changes the value of an HTML input element.

Using the getElementById() method: The getElementById() method takes a string as an argument and returns the HTML element with the specified ID.

Using the getElementsByTagName() method: The getElementsByTagName() method takes a string as an argument and returns an array of all the HTML elements with the specified tag name.

What are the different ways to access an HTML element in JavaScr Using the querySelector() method: The querySelector() method takes a CSS selector as an argument and returns the first HTML element that matches the selector.

What is the scope of a variable in JavaScript? The scope of a variable in JavaScript is the part of the code where the variable can be accessed. Variables declared with the var keyword have a local scope, which means that they can only be accessed within the block of code where they are declared. Variables declared with the let keyword have a block scope, which means that they can only be accessed within the block of code where they are declared and any nested blocks. Variables declared with the const keyword have a global scope, which means that they can be accessed from anywhere in the code. What are the different ways to create objects in JavaScript? There are multiple ways to create objects in JavaScript, including object literals, constructor functions, the Object.create() method and the class syntax introduced in ECMAScript 2015 (ES6).

The window object represents the browser window. The window object can be used to access the browser’s features, such as the location bar, the status bar and the bookmarks bar.

What is the purpose of the window object in JavaScript?

What is the purpose of the async and await keywords in JavaScript The async and await keywords are used for handling asynchronous operations in a more synchronous-like manner. The async keyword is used to define an asynchronous function, and the await keyword is used to pause the execution of an async function until a promise is fulfilled or rejected.

React is a front-end and open-source JavaScript library which is useful in developing user interfaces specifically for applications with a single page. It is helpful in building complex and reusable user interface(UI) components of mobile and web applications as it follows the component-based approach. The important features of React are:

What is React?

What are the advantages of using React?

What are the limitations of React?

It supports server-side rendering.

It will make use of the virtual DOM rather than real DOM (Data Object Model) as RealDOM manipulations are expensive. It follows unidirectional data binding or data flow.

It uses reusable or composable UI components for developing the view.

Use of Virtual DOM to improve efficiency: React uses virtual DOM to render the view. As the name suggests, virtual DOM is a virtual representation of the real DOM. Each time the data changes in a react app, a new virtual DOM gets created. Creating a virtual DOM is much faster than rendering the UI inside the browser. Therefore, with the use of virtual DOM, the efficiency of the app improves. Gentle learning curve: React has a gentle learning curve when compared to frameworks like Angular. Anyone with little knowledge of javascript can start building web applications using React.

SEO friendly: React allows developers to develop engaging user interfaces that can be easily navigated in various search engines. It also allows server-side rendering, which boosts the SEO of an app.

Reusable components: React uses component-based architecture for developing applications. Components are independent and reusable bits of code. These components can be shared across various applications having similar functionality. The re-use of components increases the pace of development. Huge ecosystem of libraries to choose from: React provides you with the freedom to choose the tools, libraries, and architecture for developing an application based on your requirement

React is not a full-blown framework as it is only a library.

The components of React are numerous and will take time to fully grasp the benefits of all. It might be difficult for beginner programmers to understand React.

Coding might become complex as it will make use of inline templating and JSX.

What is useState() in React? The useState() is a built-in React Hook that allows you for having state variables in functional components. It should be used when the DOM has something that is dynamically manipulating/controlling

JSX is basically a syntax extension of regular JavaScript and is used to create React elements. These elements are then rendered to the React DOM. All the React components are written in JSX. To embed any JavaScript expression in a piece of code written in JSX we will have to wrap that expression in curly braces {}.

What is JSX?

What are components and their type in React?

Example of JSX: The name written in curly braces { } signifies JSX

A Component is one of the core building blocks of React. In other words, we can say that every application you will develop in React will be made up of pieces called components. Components make the task of building UIs much easier. In React, we mainly have two types of components:

Functional Components: Functional components are simply javascript functions. We can create a functional component in React by writing a javascript function.

Class Components: The class components are a little more complex than the functional components. The functional components are not aware of the other components in your program whereas the class components can work with each other. We can pass data from one class component to another class component.

How do browsers read JSX? In general, browsers are not capable of reading JSX and only can read pure JavaScript. The web browsers read JSX with the help of a transpiler. Transpilers are used to convert JSX into JavaScript. The transpiler used is called Babel

Explain the use of render method in React? React renders HTML to the web page by using a function called render(). The purpose of the function is to display the specified HTML code inside the specified HTML element. In the render() method, we can read props and state and return our JSX code to the root component of our app.

What is state in React? The state is an instance of React Component Class that can be defined as an object of a set of observable properties that control the behaviour of the component. In other words, the State of a component is an object that holds some information that may change over the lifetime of the component.

React allows us to pass information to a Component using something called props (which stands for properties). Props are objects which can be used inside a component

Explain props in React?

We can access any props inside from the component’s class to which the props is passed. The props can be accessed as shown below:

this.props.propName;

What is higher-order component in React? Higher-order components or HOC is the advanced method of reusing the component functionality logic. It simply takes the original component and returns the enhanced component. HOC are beneficial as they are easy to code and read. Also, helps to get rid of copying the same logic in every component.

# Functional Components

A functional component is just a plain JavaScript pure function that accepts props as an argument No render method used

Also known as Stateless components

React lifecycle methods (for example, componentDidMount) cannot be used in functional components. Constructors are not used.

# Class Components

A class component requires you to extend from React. Component and create a render function It must have the render() method returning JSX

Also known as Stateful components

React lifecycle methods can be used inside class components (for example, componentDidMount) Explain the difference between functional and class component in RConstructor is used as it needs to store state.

What is conditional rendering in React? When there are multiple components in react and we want to render components according to our preference and some conditions then we use conditional rendering. In conditional rendering, a condition is specified and if the condition is passed then this component is rendered. What is react router? React Router is a standard library for routing in React. It enables the navigation among views of various components in a React Application, allows changing the browser URL, and keeps the UI in sync with the URL.

The main components of a react-router are:

Explain the components of a react-router

Router(usually imported as BrowserRouter): It is the parent component that is used to store all of the other components. Everything within this will be part of the routing functionality Switch: The switch component is used to render only the first route that matches the location rather than rendering all matching routes.

Route: This component checks the current URL and displays the component associated with that exact path. All routes are placed within the switch components. Link: The Link component is used to create links to different routes.

What is the use of ref in React? Refs are a function provided by React to access the DOM element and the React element that you might have created on your own. They are used in cases where we want to change the value of a child component, without making use of props and all. They have wide functionality as we can use callbacks with them. What are hooks in React? Hooks are a new addition in React 16.8. They let developers use state and other React features without writing a class. Hooks doesn’t violate any existing React concepts. Instead, Hooks provide a direct API to react concepts such as props, state, context, refs and life-cycle

What is useState() in React ?

The most used hook in React is the useState() hook. It allows functional components to manipulate DOM elements before each render. Using this hook we can declare a state variable inside a function but only one state variable can be declared using a single useState() hook. Whenever the useState() hook is used, the value of the state variable is changed and the new variable is stored in a new cell in the stack.

Explain the useEffect hook in react? The useEffect hook in React eliminates the side effect of using class based components. It is used as an alternative to componentDidUpdate() method. The useEffect hook accepts two arguments where second argument is optional.

What is React Fragments? when we are trying to render more than one root element we have to put the entire content inside the ‘div’ tag which is not loved by many developers. So since React 16.2 version, Fragments were introduced, and we use them instead of the extraneous ‘div’ tag. The following syntax is used to create fragment in react.

What is prop drilling and its disadvantages? Prop drilling is basically a situation when the same data is being sent at almost every level due to requirements in the final level. The problem with Prop Drilling is that whenever data from the Parent component will be needed, it would have to come from each level, Regardless of the fact that it is not needed there and simply needed in last.

What is custom hooks in React? Custom hooks are normal JavaScript functions whose names start with “use” and they may call other hooks. We use custom hooks to maintain the DRY concept that is Don’t Repeat Yourself. It helps us to write a logic once and use it anywhere in the code

# useRef

It is a hook.

It uses the same ref throughout.

It saves its value between re-renders in a functional component It returns a mutable ref object.

The refs created using the useRef can persist for the entire component lifetime. It is used in functional components.

# createRef

It is a function

It creates a new ref every time.

It creates a new ref for every re-render. It returns a read-only ref object

The refs created using the createRef can be referenced throughout the component.

What is the difference between useref and createRef in React ? It is used in class components.

In ReactJS, Cross-Origin Resource Sharing (CORS) refers to the method that allows you to make requests to the server deployed at a different domain. As a reference, if the frontend and backend are at two different domains, we need CORS there. We can setup CORS evironment in frontend using two methods:

Explain CORS in React?

axios fetch

What is arrow function in React? How is it used? The Arrow function is the new feature of the ES6 standard. If you need to use arrow functions, it is not necessary to bind any event to 'this.' Here, the scope of 'this' is global and not limited to any calling function. So If you are using Arrow Function, there is no need to bind 'this' inside the constructor. It is also called 'fat arrow '(=>) functions. For axios gothrow these website https://interviewprep.org/axios-interview-questions/

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | | | | | |