

ECMAScript

ECMAScript is a scripting-language specification standardized by Ecma International. It was created to standardize JavaScript to **help advance multiple independent implementations**.

Variables

JavaScript variables are containers for storing data values.

Operators

Assign values to variables and add them together:

The assignment **operator (=)** assigns a value to a variable.

Operands

The numbers (in an arithmetic operation) are called **operands**.

Assignment

Assignment operators assign values to JavaScript variables.

=, +=, *=

Data Types

JavaScript variables can hold many **data types**: numbers, strings, objects and more

JavaScript Objects

JavaScript objects are written with curly braces **{}**.

```
var person = {firstName:"John", lastName:"Doe", age:50,  
eyeColor:"blue"};
```

Difference Between Undefined and Null

Null means an empty or non-existent value. Null is assigned, and explicitly means nothing. Undefined means a variable has been declared, but the value of that variable has not yet been defined.

Primitive Data

A primitive data value is a single simple data value with no additional properties and methods. The `typeof` operator can return one of these primitive types:

String number Boolean undefined

Events

Onchange, onclick, onmouseover, onmouseout, onkeydown, onload

Looping Array Elements

```
<script>
```

```
var fruits, text, fLen, i;
```

```
fruits = ["Banana", "Orange", "Apple", "Mango"];
```

```
fLen = fruits.length;
```

```
text = "<ul>";
```

```
for (i = 0; i < fLen; i++) {  
    text += "<li>" + fruits[i] + "</li>";  
}  
  
text += "</ul>";  
  
document.getElementById("demo").innerHTML = text; </script>
```

Popping

The `pop()` method removes the last element from an array:

```
var fruits = ["Banana", "Orange", "Apple", "Mango"];  
fruits.pop();
```

Pushing

The `push()` method adds a new element to an array (at the end):

```
var fruits = ["Banana", "Orange", "Apple", "Mango"];  
fruits.push("Kiwi");
```

Sorting an Array

The `sort()` method sorts an array alphabetically:

```
var fruits = ["Banana", "Orange", "Apple", "Mango"];  
fruits.sort();
```

Reversing an Array

The `reverse()` method reverses the elements in an array.

```
var fruits = ["Banana", "Orange", "Apple", "Mango"];  
fruits.sort();    // First sort the elements of fruits  
fruits.reverse();
```

JavaScript Array Iteration Methods

`Array.forEach()`

`Array.map()`

`Array.filter()`

`Array.reduce()`

`Array.indexOf()`

`Array.lastIndexOf()`

`Array.find()`

`Array.findIndex()`

Arrow Function

Arrow functions were introduced in ES6.

Arrow functions allow us to write shorter function syntax:

Closures

JavaScript variables can belong to the **local** or **global** scope.

Global variables can be made local (private) with **closures**.

JSON

JavaScript Object Notation

Lexical Scoping

lexical scope is the author-time scope created by a closure. It is the 'outer' scope of a function which is defined inside a closure.

Bubbling

When an event happens on an element, it first runs the handlers on it, then on its parent, then all the way up on other ancestors.

Let's say we have 3 nested elements `FORM > DIV > P` with a handler on each of them:

REACT Questions

What is Redux?

Redux is a state management tool.

Reducers

Reducers specify how the application's state changes in response to actions sent to the store.

Actions

Actions are payloads of information that send data from your application to your store. They are the only source of information for the store. You send them to the store using `store.dispatch()` .

What is props in react?

Props are basically kind of global variable or object. React allows us to pass information to a Component using something called props (stands for properties).

What is state in react?

The State of a component is an object that holds some information that may change over the lifetime of the component.

What is difference between props and state in react?

In a React **component**, props are variables passed to it by its parent **component**. State on the other hand is still variables, but directly initialized and managed by the component.

What is axios in react?

Axios is promise-based and thus we can take advantage of `async` and `await` for more readable asynchronous code. **Axios** is a lightweight HTTP client.

What is HOC in react?

A higher-order component (**HOC**) is an advanced technique in **React** for reusing component logic.

What is the difference between `componentDidMount` and `componentWillMount`?

`componentDidMount()` is only called once, on the client, compared to `componentWillMount()` which is called twice, once to the server and once on the client. It is called after the initial render when the client received data from the server and before the data is displayed **in the** browser.

What is render in react?

Render refers to show output in browser and @**JavaScript** (programming language) utilize the document object to manipulate DOM elements. You can get and set element through document.getElementById() to manipulate the rendered elements.

What is setState in react?

we're passing an object to **setState()** . The object contains the part of the state we want to update which, in this case, is the value of searchTerm . **React** takes this value and merges it into the object that needs it.

What is virtual dom in react?

Like the actual **DOM**, the **Virtual DOM** is a node tree that lists elements and their attributes and content as objects and properties. **React's** render() method creates a node tree from **React** components and updates this tree in response to mutations in the data model, caused by actions.

What are Lifecycle method in react?

When developing in React, every Component follows a cycle from when it's created and mounted on the DOM to when it is unmounted and destroyed.

What is dispatch method in react?

`dispatch()` is the method **used to** dispatch **actions and trigger state changes to the store**. **react-redux is simply trying to give you convenient access to it.**

What is middleware in react?

Redux Middleware. Middleware provides a way to interact with actions that have been dispatched to the store before they reach the store's reducer.

What is context?

Context provides a way to pass data through the component tree without having to pass props down manually at every level. For example, authenticated user, locale preference, UI theme need to be accessed in the application by many components.

What is stateful component?

That means the **stateful components** are keeping track of changing data, while stateless **components** print out what is given to them via props, or they always render the same thing. **Stateful/Container/Smart component:** class Main extends **Component** { constructor() { super() this.

What are stateless components?

STATELESS COMPONENT declared as a function that has no state and returns the same markup given the same props. A quote from the React documentation: These **components** must not retain internal state, do not have backing instances, and do not have the **component** lifecycle methods.