React Basics

React is a JavaScript library for building user interfaces.

React is used to build single-page applications.

React allows us to create reusable UI components.

## JSX

JSX stands for JavaScript XML.

JSX allows us to write HTML in React.

JSX makes it easier to write and add HTML in React.

JSX allows us to write HTML elements in JavaScript and place them in the DOM

export default is **used to export a single class, function or primitive from a script file**. The export can also be written as export default class HelloWorld extends React.Component { render() { return <p>Hello, world!</p>

# React Components

Components are like functions that return HTML elements.

Components are independent and reusable bits of code. They serve the same purpose as JavaScript functions, but work in isolation and return HTML.

Components come in two types, Class components and Function components

# React Props

Props are arguments passed into React components.

Props are passed to components via HTML attributes.

props stands for properties.

React Props are like function arguments in JavaScript and attributes in HTML.

React Lifecycle

## Lifecycle of Components

Each component in React has a lifecycle which you can monitor and manipulation

### constructor

The constructor() method is called before anything else, when the component is initiated, and it is the natural place to set up the initial state and other initial values.

The constructor() method is called with the props, as arguments, and you should always start by calling the super(props) before anything else, this will initiate the parent's constructor method and allows the component to inherit methods from its parent (React.Component).

**This**

this keyword refers to an **object**.

**Which** object depends on how this is being invoked (used or called).

The this keyword refers to different objects depending on how it is used:

The preventDefault

The preventDefault() method cancels the event if it is cancelable, meaning that the default action that belongs to the event will not occur.

For example, this can be useful when:

* Clicking on a "Submit" button, prevent it from submitting a form
* Clicking on a link, prevent the link from following the URL

**Note:** Not all events are cancelable. Use the [cancelable](https://www.w3schools.com/jsref/event_cancelable.asp) property to find out if an event is cancelable.

Handle Submit

Handling forms is about how you handle the data when it changes value or gets submitted. In HTML, form data is usually handled by the DOM. In React, **form data is usually handled by the components**. When the data is handled by the components, all the data is stored in the component state.

getElementById('root') MEANS: **Render the whole React App into the element with id=root** .

### render

The render() method is required and will always be called, the others are optional and will be called if you define them.

# React Events

Just like HTML DOM events, React can perform actions based on user events.

React has the same events as HTML: click

# React Forms

Just like in HTML, React uses forms to allow users to interact with the web page.