



@CODE.CLASH

React Props :

An overview



NEXT →

Hey Everyone 🙌

In this Post, we'll talk about an important concept in React – props.

we'll see how to use them to keep the flow of data in react app dynamic.

Do Like, save and Share This Post If You Found This Helpful.

React Props

- Props stands for properties.
- Props are arguments passed into React components.
- Props are passed to components via HTML attributes.
- React Props are like function arguments in JavaScript and attributes in HTML.
- React's data flow between components is **Uni-directional**.
[from parents to child only]

Purpose of Props

- They are used to pass data between React Components.
- It is similar to function arguments.
- Props are passed to the component in the same way as arguments passed in a function.
- Props are immutable so we cannot modify the props from inside the component.

Props Example

- To send props into a component, use the same syntax as HTML attributes.
- Add a "brand" attribute to the Laptop element:

```
const myElement = <Laptop brand="Acer" />;
```

- The component receives the argument as a props object:

```
function Laptop(props) {  
  return <h2>I am a { props.brand } Laptop!</h2>;  
}
```

Passing Data

- Props are also how you pass data from one component to another, as parameters.
- Create a variable named `webName` and send it to the `WebApp` component:

```
WebApp.jsx

import React from 'react'

const WebApp = (props) => {
  return (
    <h2>I am a {props.website}</h2>
  )
}

export default WebApp;
```

```
Server.jsx

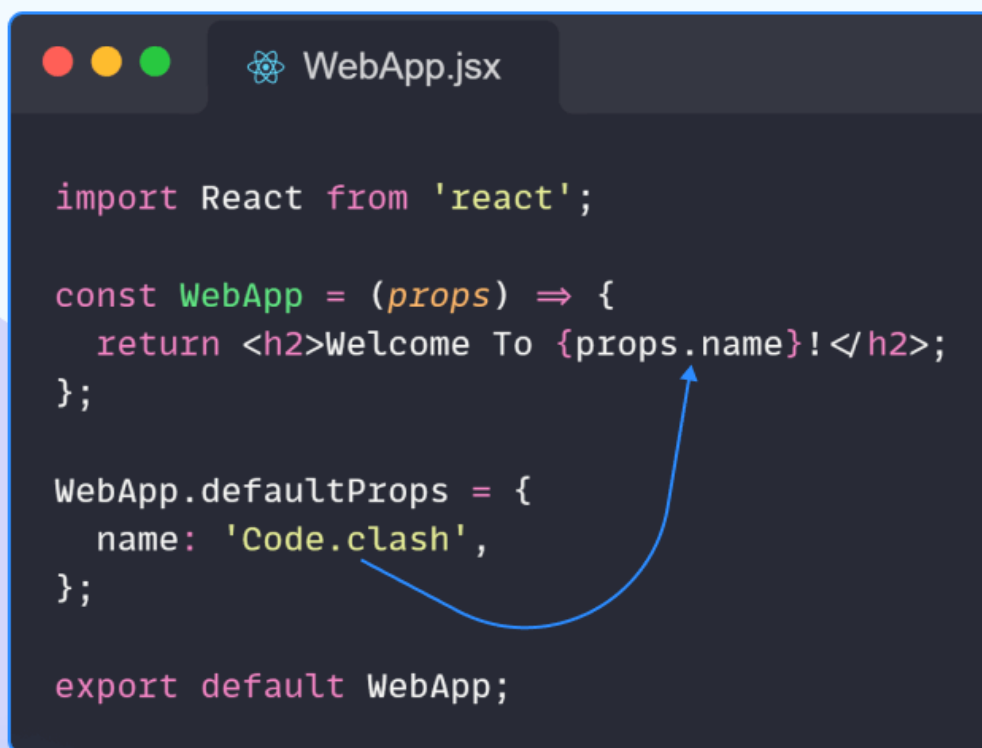
import React from 'react';
import WebApp from './WebApp';

const Server = () => {
  const webName = 'instagram.com';
  return (
    <div>
      <h1>Who live in my Server?</h1>
      <WebApp website={webName} />
    </div>
  );
};

export default Server;
```

Default Props

- It is not necessary to always add props from parent components.
- You can also set default props directly on the component



```
import React from 'react';

const WebApp = (props) => {
  return <h2>Welcome To {props.name}</h2>;
};

WebApp.defaultProps = {
  name: 'Code.clash',
};

export default WebApp;
```

The image shows a code editor window titled 'WebApp.jsx'. It contains a React component definition. A blue arrow points from the 'name' property in the 'defaultProps' object to the '{props.name}' placeholder in the JSX element, illustrating how the default prop is used when no prop is passed from the parent.

- React Props are **read-only!** You will get an **error** if you try to change their value.

Best Of Luck :)

@CODE.CLASH

THANKS FOR YOUR ATTENTION



CODE.CLASH



IMTIYAZ NANDASANIYA

**LIKE AND SAVE IT FOR
LATER**