# React Components-II

## Props in React

A component can pass information to other components. Information that gets passed from one component to another is known as props short for properties. A component's props is an object which holds information about that component.

Props are passed down from parent to child components as a key and value pair. If we want to pass information that is not string we have to wrap it with curly braces. This information will be stored inside of the props object of the child component.

The most common use of props is to pass data and event handlers down to the child components.

### Rules of Props

There is only one strict rule in regard to props in React. Props are read-only. A component should never try to mutate or change the value of its props.

### Default Props

Default props can be used to define any props that you want to be set for a component. They can be used to ensure that props will have a value if it was not specified by the parent component.

We can set default values for the props by assigning to the special **defaultProps** property on the component class.

### Additional Information: Type Checking in Props

**Note -** Props type checking can be optionally used to ensure that the passed value is of the correct data. This can help prevent errors in rendering and force correct usage of components.

Props type-checking can be used to validate props that are passed down from the parent for missing or incorrect data type values.

Type checking of props can also help document the code to make it easier to understand and debug the component class.

We can add type checking to our props by specifying it on the **propsTypes** static property on the components class after it has been defined.The value of this property is an object that has multiple key and value pairs. Each key corresponds to a prop of that our component expects and the value should be the expected data type for that prop.

We need to import the PropTypes object from ‘prop-types’ and use it to specify the expected data type for each prop.

**Example snippet -**

```
import { Component } from "react";
import PropTypes from "prop-types";

export default class Navbar extends Component {
 render() {
   const { username, avatar } = this.props;
   return (
     <div>
       <span>{username}</span>
       <img alt={username} src={avatar} />
     </div>
   );
 }
}

// setting default props
Navbar.defaultProps = {
 username: "Pranav",
 avatar: "/image.png"
};
// type checking props
Navbar.propTypes = {
 username: PropTypes.string,
 avatar: PropTypes.string
};
```

## State v/s Props

Props and state are both plain JavaScript objects. While both hold information that influences the output of render, one important difference between the two is that Props get passed to the component whereas state is managed within the component.

| State | Props |
|---|---|
| State can be changed (mutable) | Props are read-only and cannot be changed (immutable) |
| State changes can be asynchronous | Props cannot be changed |
| State is managed within the component | Props gets passed to the component |
| State can used to display changes with the component | Props are used to pass information between components |

## Some References:

- Props vs State: link
- Presentational and Container components: link