

Conditional components

Have you ever visited a website that required a user account? To log in you click on a **Log in** button and once you've logged in, the **Log in** button changes to a **Log out** button.

This is often done using something called conditional rendering.

In a previous course, you've already learned about simple conditions using if and switch statements.

Using these statements allows you to change the behaviour of code based on certain conditions being met.

For example, you can set a variable to a different value based on the result of a condition check.

```
1  let name;
2  if (Math.random() > 0.5) {
3    name = "Mike"
4  } else {
5    name = "Susan"
6  }
```

```
1  let name;
2  let newUser = true;
3  if (Math.random() > 0.5 && newUser) {
4    name = "Mike"
5  } else {
6    name = "Susan"
7  }
```

Conditional rendering is built on the same principle. By using conditions, you can return different child components. This is often done using the props that are passed into the parent component, but can also be done based on component state.

Let's take a look at a simple example.

Let's say you have two child components called **LoginButton** and **LogoutButton**; each displaying their corresponding button.

In the parent component, named **LogInOutButton**, you can check the props passed into the parent component and return a different child component based on the value of the props.

In this example, the props contains a property named **isLoggedIn**. When this is set to **true**, the **LogoutButton** component is returned. Otherwise, the **LoginButton** component is returned.

```
1  function LogInOutButton(props) {
2    const isLoggedIn = props.isLoggedIn;
3    if (isLoggedIn) {
4      return <LogoutButton />;
```

```
5     } else {  
6       return <LoginButton />;  
7     }
```

Then when the **LogInOutButton** parent component is used, the prop can be passed in.

```
1   <LogInOutButton isLoggedIn={false} />
```

This is a simple example showing how you can change what is displayed based on a condition check. You will use this often when developing React applications.