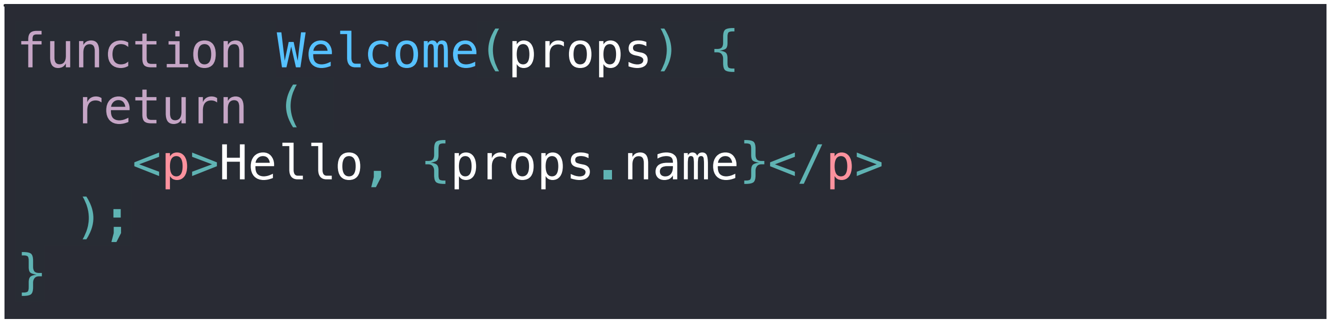
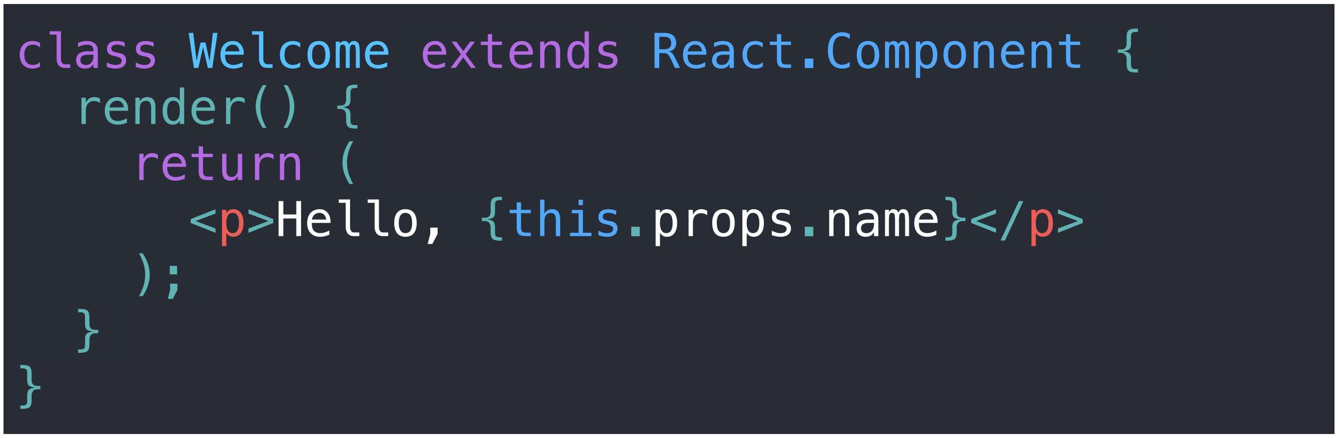
REACT

FUNCTIONAL COMPONENTS



```
function Welcome(props) {
  return (
    <p>Hello, {props.name}</p>
  );
}
```

CLASS COMPONENTS



```
class Welcome extends React.Component {
  render() {
    return (
      <p>Hello, {this.props.name}</p>
    );
  }
}
```

RENDERING A COMPONENT

```
ReactDOM.render(element, target);
```

JSX

JavaScript expressions	Wrap in curly braces `Hello, {formatName(user)}!`
Attribute names	Camel case, using JS standards `<div className="hidden"></div>;`

Attribute values	String literals in quotes; expressions without quote
	`<div tabIndex="0"></div>;`
	`<img src={user.avatarUrl}></img>;`

Empty tags	Close with space and slash `<img src={user.avatarUrl} />;`

BUILDING A LIST

Usage:

```
const name = array.map((element) =>
  <li>{element}</li>
);
return (
  <ul>{name}</ul>
);
```

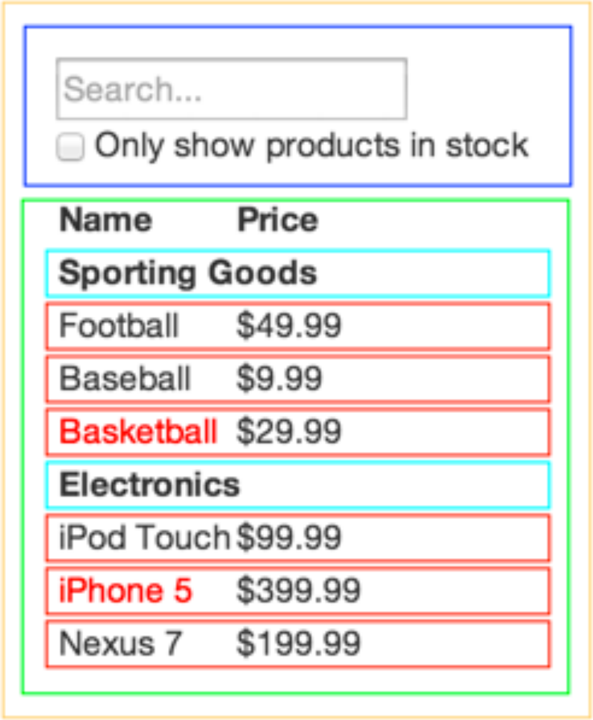
Example:

```
const listItems = numbers.map((number) =>
  <li>{number}</li>
);
return (
  <ul>{listItems}</ul>
);
```

CREATING A COMPONENT HIERARCHY

1. Start with a mock
2. Draw a box around each item that does one thing (including groups)
3. Name the boxes (semantically)
4. Make a hierarchy

- FilterableProductTable
  - SearchBar
  - ProductTable
    - ProductCategoryRow
    - ProductRow



Search...

□ Only show products in stock

Name	Price
Sporting Goods	
Football	$49.99
Baseball	$9.99
Basketball	$29.99
Electronics	
iPod Touch	$99.99
iPhone 5	$399.99
Nexus 7	$199.99