



**Skills**  
Network

## Introduction to Components

**IBM**

# What you will learn

---



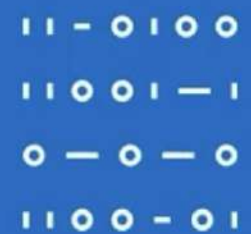
Explain what React components are



Describe the features of React components



Compare and contrast the three types of components

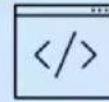


Create a Hello World React component

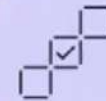
# What is a component?

---

Modular chunks of code



Building blocks



Makes building UI easier

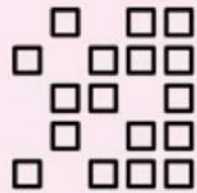


Merge into a parent component



# More about components

---



Enables you to break up the user interface that can then be reused and handled independently



Take an optional input and return a React object, which is rendered on the screen

# Purpose

---



Renders pieces of the web page



Updates its state when the user interacts with it



Manages events such as button selection

# Component features

---



## Properties

Store and pass data



## Events

Manage the DOM



## States

Update the UI

# Functional components

---

Written like JavaScript functions

Hooks introduced in v16.8

Take in arguments

Hooks enable states

Return JSX

Easy to read and test

# Function component example

---

```
const Democomponent = () =>
{
  return<h1>Welcome Message!</h1>;
}
```



# Class components

---

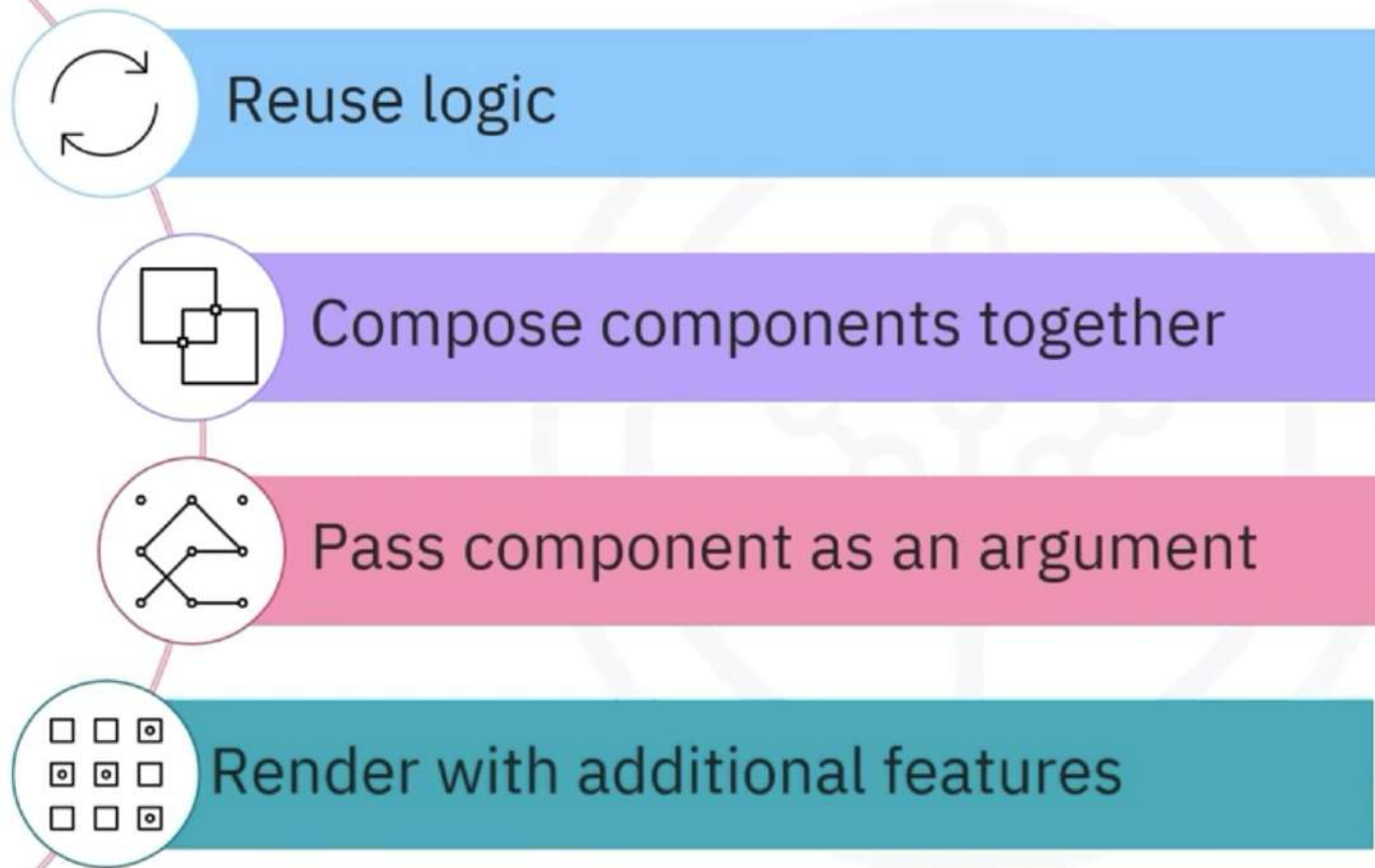
Create using  
JavaScript ES6  
classes

Pass data from  
one to another

Use states,  
props, and  
lifecycle  
methods

More complex  
than functional

# Higher-order component



# Recap

---

In this video, you learned that:

- You create React applications with modular code called components
- Components break up the user interface into separate parts that you can reuse and handle independently
- A React component:
  - Renders the initial user interface
  - Manages and handles events
  - Updates the user interface according to its state
- The three types of React components include functional, class, and higher-order