# React Handson 2

## Explain React Components

React components are the building blocks of a React application. They allow developers to break down complex UIs into smaller, manageable, and reusable pieces. A component can be thought of as a JavaScript function or class that returns a portion of the UI in the form of JSX (JavaScript XML). Components can be either class-based or function-based.

## Identify the Differences Between Components and JavaScript Functions

React Component vs JavaScript Function:  
  
Feature | React Component | JavaScript Function  
--- | --- | ---  
Purpose | Used to render UI elements | Used to perform logic or return values  
Returns | JSX (UI elements) | Any JavaScript value  
State Management | Can manage state using hooks or class state | Cannot manage UI state directly  
Lifecycle Methods | Has lifecycle methods (in class components) | No lifecycle methods  
React Integration | Can be integrated into React apps | Not directly usable as React components

## Identify the Types of Components

1. Class Components: ES6 classes that extend React.Component and include a render() method.  
2. Function Components: JavaScript functions that return JSX. With React Hooks, function components can now manage state and side effects.

## Explain Class Component

A class component is a JavaScript class that extends React.Component and must include a render() method. Class components have access to lifecycle methods like componentDidMount() and can maintain internal state.  
  
Example:  
class Welcome extends React.Component {  
 render() {  
 return <h1>Hello, {this.props.name}</h1>;  
 }  
}

## Explain Function Component

A function component is a plain JavaScript function that accepts props and returns JSX. It is a simpler way to write components and is commonly used with React Hooks to handle state and side effects.  
  
Example:  
function Welcome(props) {  
 return <h1>Hello, {props.name}</h1>;  
}

## Define Component Constructor

The constructor is a special function used in class components to initialize state and bind methods. It is called before the component is mounted.  
  
Example:  
class MyComponent extends React.Component {  
 constructor(props) {  
 super(props);  
 this.state = { count: 0 };  
 }  
 render() {  
 return <p>Count: {this.state.count}</p>;  
 }  
}

## Define render() Function

The render() function is a required method in class components that returns the JSX (UI) to be rendered on the screen. It should be a pure function and must return a single root element.  
  
Example:  
render() {  
 return (  
 <div>  
 <h1>Welcome to React</h1>  
 </div>  
 );  
}