# React.js Week 1 – Day 1: Introduction to Components

## 1. What are Components?

- In React, **components** are like reusable building blocks.
- Each component is a **function (or class)** that returns UI (HTML-like code using JSX).
- Components let you break your UI into **small, independent, reusable pieces**.

Example analogy: - Think of a **website** as a house. - **Components** are like bricks, windows, and doors. - You build the full house by combining these smaller parts.

---

## 2. Types of Components

1. **Functional Components (modern standard)**

2. Just a JavaScript function that returns JSX.

```
function Welcome() {
  return <h1>Hello, I am a Functional Component!</h1>;
}
export default Welcome;
```

3. **Class Components (older style)**

4. A class that extends `React.Component`.
5. Still works, but nowadays **functional components + hooks** are preferred.

```
import React, { Component } from 'react';

class Welcome extends Component {
  render() {
    return <h1>Hello, I am a Class Component!</h1>;
  }
}
export default Welcome;
```

---

## 3. How Components Work

- Each component can:
- Return **JSX** (HTML inside JavaScript).
- Be **reused** multiple times.
- Be **nested** inside other components.

Example:

```
function Header() {
  return <h1>Welcome to My Website</h1>;
}

function Footer() {
  return <p>© 2025 My Website</p>;
}

function App() {
  return (
    <div>
      <Header />
      <p>This is the main content.</p>
      <Footer />
    </div>
  );
}

export default App;
```

Here: - `Header` and `Footer` are components. - `App` combines them to make the full page.

---

## 4. Rules for Components

- Component names **must start with a capital letter** (e.g., `Header`, not `header`).
- Must return **one root element** (usually a `<div>` or `<>...</>` fragment).
- Components can be imported and reused in other files.

---

## 5. Your Task (Practice)

👉Now it's your turn: 1. Inside your `src/` folder, create a file called `Header.js`. 2. Define a **functional component** `Header` that shows: `Hello, I am learning React Components 😥`. 3. Import

`Header` into `App.js` and display it **above** your Hello React.js text. 4. Run the app and confirm your custom component is visible.

---

👉🏼By completing this, you’ll understand how to: - Create a custom component. - Import and use it inside another component. - Structure your React app with reusable pieces.