React JS hands on document for React Components:

1)What is a React Component?

In React, a component is like a reusable block of code that returns HTML (JSX) to be shown on the screen.  
It's like a building block of a React application.

2) difference between React components and JavaScript functions

JavaScript Functions:

* General-purpose code blocks used to perform tasks or calculations.
* Return values like numbers, strings, arrays, or objects.
* Do not manage or display UI directly.
* Do not require JSX or React imports.
* Example: A function that adds two numbers and returns the result.

React Components:

* Special functions used to build user interfaces in React.
* Return JSX (HTML-like code) which describes how the UI should look.
* Must start with a capital letter when named.
* Require importing React (in older versions).
* Can use state, props, and lifecycle methods (in class components).

Example: A component that displays a welcome message on a webpage

| Concept | JavaScript Function | React Component |
| --- | --- | --- |
| Purpose | Perform logic or calculation | Display part of a webpage |
| Returns | Data (number, string, etc.) | JSX (HTML-like structure) |
| Usage | Called in JS code | Used inside React apps |
| UI Involved | No | Yes |

3) Identify the types of components

1. Functional Components

* These are simple JavaScript functions that return JSX.
* They are the most commonly used type in modern React.
* Easy to write and read.
* Can use React Hooks like useState, useEffect.

Example:

function Hello() {

return <h1>Hello World</h1>;

}

2. Class Components (older way)

* These are written using ES6 classes.
* They have a render() method to return JSX.
* Used before React Hooks were introduced.

Example:

import React, { Component } from 'react';

class Hello extends Component {

render() {

return <h1>Hello World</h1>;

}

}

3. Presentational Components

* Also called "Dumb components".
* Focus only on UI display.
* Receive data via props and display it.
* Do not manage any internal logic or state.

4. Container Components

* Also called "Smart components".
* Handle logic, state, and data fetching.
* Pass data to presentational components.

5. Pure Components (Only in Class Components)

* Automatically prevent unnecessary re-renders.
* Used for performance optimization.
* Uses shallow comparison of props and state.

6. Higher-Order Components (HOC)

* A function that takes a component and returns a new enhanced component.
* Used for code reuse and logic sharing.

4)Explain class component?

A Class Component is a traditional way to define a component in React using ES6 JavaScript classes. It can hold state and use lifecycle methods, which makes it more powerful than functional components (before Hooks were introduced).

Syntax of a Class Component:

import React, { Component } from 'react';

class Welcome extends Component {

constructor(props) {

super(props);

this.state = {

name: 'Guest'

};

}

render() {

return (

<div>

<h1>Hello, {this.state.name}!</h1>

</div>

);

}

}

export default Welcome;

Key Features of Class Components:

* Can use state (with this.state)
* Can handle lifecycle methods like componentDidMount, componentDidUpdate
* Uses this.props to access passed data
* Requires render() method to return JSX

Use:

Use class components when:

* You need state or lifecycle methods (in pre-Hooks code)
* You’re working with older React codebases

5) Explain function component?

A functional component is a JavaScript function that returns JSX (HTML-like code).  
It is the simplest and most modern way to create components in React.

Features of Functional Components:

* Just functions that return JSX.
* Can use React Hooks like useState, useEffect.
* Easier to write, understand, and test.
* Don’t require this keyword like class components.

Simple example:

import React from 'react';

function Welcome() {

return <h1>Hello, welcome to React!</h1>;

}

export default Welcome;

6)Define component constructor

In React, the constructor is a special method used inside class components. It is called before the component is mounted and is used to:

* Initialize state
* Bind event handlers
* Access props via super(props)

7)Define render() function

* The render() function is a mandatory method in every React class component.
* It returns JSX (HTML-like code) that tells React what to display on the screen.
* React calls the render() method whenever the component is first loaded or re-rendered due to state/props change.

Syntax:

render() {

return (

<div>

<h1>Hello, React!</h1>

</div>

);

}