ReactJS Hands-On Lab Report - Score Calculator App

# Objectives

- Explain React components  
- Identify the differences between components and JavaScript functions  
- Identify the types of components  
- Explain class component  
- Explain function component  
- Define component constructor  
- Define render() function

# Concepts and Explanations

## React Components

React components are the core building blocks of a React application. They represent parts of the UI and can be reused, composed, and nested.

## Components vs JavaScript Functions

While JavaScript functions are used for performing logic, React components return JSX that defines UI structure. React components are either function-based or class-based.

## Types of Components

1. Functional Components – created using JavaScript functions and may use hooks.  
2. Class Components – use ES6 classes and include lifecycle methods.

## Class Component

A class component is an ES6 class that extends from React.Component. It must include a render() method.

## Function Component

A function component is a plain JavaScript function that returns JSX. It’s the preferred modern approach for building components.

## Component Constructor

In class components, the constructor is used to initialize state and bind methods. It's not used in function components.

## render() Function

The render() function returns the JSX that defines the UI of a class component.

# Hands-On Lab: Score Calculator App

Create a React App named 'scorecalculatorapp' with a function component called 'CalculateScore' that calculates and displays the average score of a student.

## Step-by-Step Instructions

1. 1. Open Command Prompt and run: npx create-react-app scorecalculatorapp
2. 2. Navigate to the project folder: cd scorecalculatorapp
3. 3. Open the project in Visual Studio Code: code .
4. 4. In the 'src' folder, create a new folder named 'Components'.
5. 5. Inside 'Components', create a file named 'CalculateScore.js'.
6. 6. Add the provided function component code to 'CalculateScore.js'.
7. 7. In the 'src' folder, create a folder named 'Stylesheets'.
8. 8. Inside 'Stylesheets', create a CSS file named 'mystyle.css'.
9. 9. Add the provided styles to 'mystyle.css'.
10. 10. Modify 'App.js' to use the 'CalculateScore' component.
11. 11. Save all files and run the app using: npm start
12. 12. Visit http://localhost:3000 to view the result.

## Component Code: CalculateScore.js

import React from 'react';  
import '../Stylesheets/mystyle.css';  
  
function CalculateScore() {  
 const name = "Aditi";  
 const school = "KIIT";  
 const total = 450;  
 const goal = 500;  
 const average = (total / goal) \* 100;  
  
 return (  
 <div className="scoreBox">  
 <h2>Student Score Summary</h2>  
 <p><strong>Name:</strong> {name}</p>  
 <p><strong>School:</strong> {school}</p>  
 <p><strong>Total Marks:</strong> {total}</p>  
 <p><strong>Goal:</strong> {goal}</p>  
 <p><strong>Average Score:</strong> {average.toFixed(2)}%</p>  
 </div>  
 );  
}  
  
export default CalculateScore;

## CSS Code: mystyle.css

.scoreBox {  
 border: 2px solid #333;  
 padding: 20px;  
 margin: 20px;  
 width: 50%;  
 background-color: #f0f8ff;  
 font-family: Arial, sans-serif;  
 box-shadow: 2px 2px 12px rgba(0,0,0,0.1);  
}  
.scoreBox h2 {  
 color: #003366;  
}

## Modified App.js Code

import React from 'react';  
import CalculateScore from './Components/CalculateScore';  
  
function App() {  
 return (  
 <div className="App">  
 <CalculateScore />  
 </div>  
 );  
}  
  
export default App;





