**ADVANCED WEB APPLICATION DEVELOPMENT**

**ASSIGNMENT WEEK 5**

**1.Explain Model, View and Controller in brief?**

**Answer1:** Model, View, and Controller (MVC) is a software architectural pattern that is commonly used for developing user interfaces that divide the related program logic into three interconnected elements. [The purpose of using MVC is to separate internal representations of information from the ways information is presented to and accepted from the user, as well as to facilitate code reuse and maintenance1](https://en.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93controller).

The three elements of MVC are:

* **Model**: The model is responsible for managing the data of the application. It defines the structure, logic, and rules of the data, as well as how to interact with the database or other data sources. It receives user input from the controller and updates itself accordingly[1](https://en.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93controller)[2](https://www.codecademy.com/article/mvc).
* **View**: The view is responsible for generating the user interface, such as tables, forms, charts, etc. It displays the data from the model in a particular format, and allows the user to interact with the application. It sends user requests to the controller and updates itself based on the changes in the model[1](https://en.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93controller)[2](https://www.codecademy.com/article/mvc).
* **Controller**: The controller is responsible for handling user actions and performing interactions on the data model objects. It acts as a mediator between the model and the view, receiving user input, processing it, and sending it to the model or the view. It also implements the application logic and controls the flow of the application[1](https://en.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93controller)[2](https://www.codecademy.com/article/mvc).

An example of MVC in action is a web application that allows users to create and manage tasks. The model would define what a task is, how to store and retrieve tasks from the database, and how to perform operations on tasks, such as adding, deleting, or updating. The view would define how to display tasks to the user, such as in a list or a table, and how to provide input fields for creating or editing tasks. The controller would define how to handle user requests, such as clicking on a button or a link, and how to communicate with the model or the view accordingly.

2.What is the purpose of render method?

Answer 2: The render method is a function that is used to define and display the user interface of a component in React. The render method can be used in two ways:

* In class-based components, the render method is a required and essential method that must be defined in every component class. The render method returns a React element, which is a JavaScript object that describes what the component should look like. The render method can use the props and state of the component to dynamically generate the user interface. [The render method is also responsible for updating the virtual DOM whenever there is a change in the component’s props or state1](https://www.geeksforgeeks.org/react-js-render-method/)[2](https://upmostly.com/tutorials/understand-the-render-method-and-rendering-in-react).
* In functional components, the render method is not explicitly defined, but rather implied by the function body. Functional components are simple functions that return a React element, just like the render method in class-based components. Functional components can also use props and hooks to dynamically generate the user interface[2](https://upmostly.com/tutorials/understand-the-render-method-and-rendering-in-react)[3](https://codingstatus.com/react-render-method/).

The purpose of the render method in React is to make the applications reactive and declarative. It defines what the component should look like, how it should function, and what its dynamic features are. It does not directly update the actual DOM, but rather updates the virtual DOM, which is a representation of the actual DOM in memory. [React then compares the virtual DOM with the actual DOM and updates only the parts that have changed1](https://www.geeksforgeeks.org/react-js-render-method/)

3.How Do I Render Plain Html?

Answer 3: To render plain HTML elements like <div>s, you need to use different methods depending on the context and the technology you are using. Here are some possible ways to render plain HTML elements:

* Using CSS: You can use the display property to show or hide HTML elements based on a condition. For example, you can use display: none to hide an element, or display: block to show it. [You can also use media queries, pseudo-classes, or other CSS selectors to apply different display values based on the screen size, user interaction, or other factors1](https://www.w3schools.com/react/react_render.asp).
* Using Angular: You can use the ngIf directive to show or hide HTML elements based on a condition. This directive allows you to add or remove elements from the DOM depending on the truthiness of an expression. [For example, you can use \*ngIf=“condition” to show an element only if the condition is true2](https://www.geeksforgeeks.org/how-to-render-plain-text-of-html-in-node-js/).
* Using React: You can use conditional rendering to show or hide HTML elements based on a condition. This means that you can use JavaScript operators like &&, ||, or ternary operators to render different elements depending on the state of your component. For example, you can use {condition && <div>…</div>} to show a div only if the condition is true[3](https://stacktuts.com/how-to-conditionally-render-plain-html-elements-like-div-s).
* Using JavaScript: You can use conditional statements like if and else to show or hide HTML elements based on a condition. This means that you can use the document.getElementById() method to access an element by its id, and then use the style.display property to change its display value. For example, you can use if (condition) {document.getElementById(“myDiv”).style.display = “block”;} else {document.getElementById(“myDiv”).style.display = “none”;} to show or hide a div with id=“myDiv” depending on the condition.

4.[**Develop Node.js**](https://nodejs.org/)**is a platform for building fast and scalable server applications using JavaScript.**