ASP .NET

Canva

Topic : View Components

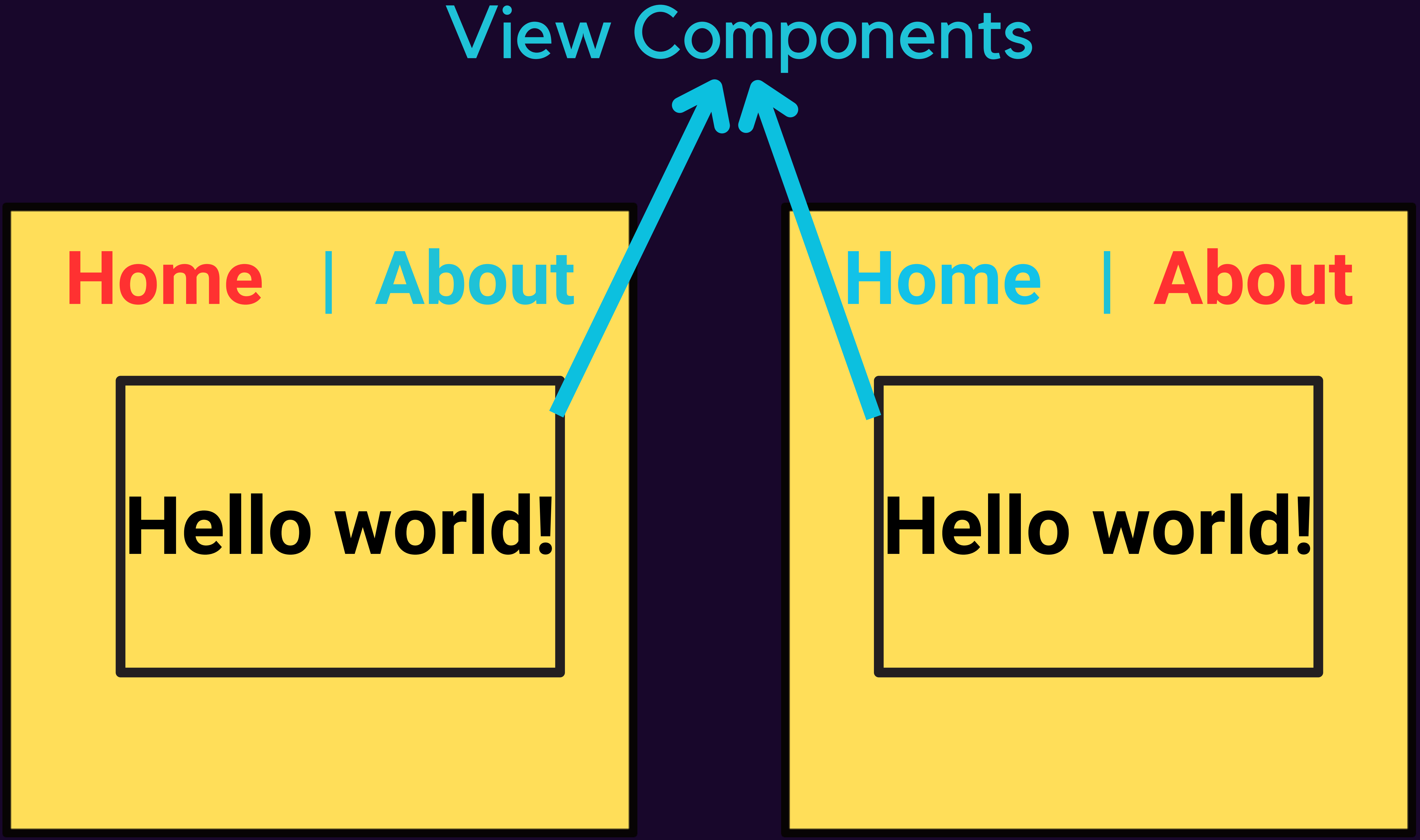
Speaker : Usama Javed

Instructor : Shuja ur Rehman

Table of Contents

What is A View Component?

"View components are similar to partial views, but they're much more powerful. View components don't use model binding, they depend on the data passed when calling the view component."


View Components
Home | About
Hello world!
Home | About
Hello world!

HOW VIEW COMPONENTS DIFFER FROM PARTIAL VIEWS:

Partial Views are similar to View Components in that they allow developers to create reusable UI elements. However, Partial Views are not self-contained and rely on the parent view's model and controller. View Components, on the other hand, are self-contained and have their own models and controllers

Benefits of View Components

- **Reusability:**
  View Components are self-contained and can be reused across different views.
- **Separation of Concerns:**
  View Components provide a way to separate UI concerns from controller logic.
- **Testability:**
  View Components can be easily tested in isolation.
- **Flexibility:**
  View Components can be used to create complex UI elements that can be reused across different projects.
- **Clean Code:**
  It provides built-in methods/properties from ViewComponent.
  It makes the code more clean.

STEP-I

Create a View Component class that inherits from the Microsoft.AspNetCore.Mvc.ViewComponent class

```
public class WeatherSummary : ViewComponent
{
    public string Invoke()
      {
          string data = "this is some weather information";
          return data;
      }
}
```

STEP-II

Invoke the View Component in the parent view.

```
<vc:weather-summary/>
```

You can also pass parameters

IMPLEMENTATION!