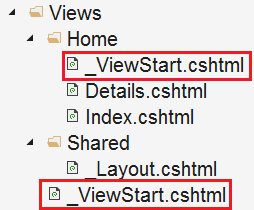
In this video, we will discuss **what is \_ViewStart.cshtml file** and it's use in ASP.NET Core MVC   
  
  
**Setting the Layout property**   
  
We use the Layout property to associate a view with a layout view. Without the \_ViewStart.cshtml file, we set the Layout property in each and every view. This violates DRY (Don't Repeat Yourself) principle, and has the following disadvantages

1. Redundant code.
2. Maintenance overhead.

To use a different layout file, all the individual views need to be updated. This is not only tedious and time-consuming process, but also error prone.   
  
**What is \_ViewStart.cshtml file in ASP.NET Core MVC**   
  
It's a special file in ASP.NET Core MVC. The code in this file is executed before the code in an individual view is executed. So this means, instead of setting the Layout property in each individual view, we can move that code into the \_ViewStart.cshtml file.

@{  
    Layout = "\_Layout";  
}

By setting the Layout property in \_ViewStart.cshtml file maintaining our application becomes much easier. In the future, if we want to use a different layout file we just need to change the code in one place in \_ViewStart.cshtml.   
  
**\_ViewStart.cshtml file is hierarchical**   
  
We usually place the **ViewStart**file in the **Views**folder. Since this file is hierarchical we can also place it in any sub-folder in the Views folder.   
  
   
  
In the folder hierarchy above, we have placed one **ViewStart**file in the **Views**folder and another **ViewStart**file in the **Home**sub-folder. The layout page specified in the ViewStart file in the Home sub-folder overwrites the layout page specified in the ViewStart file in the Views folder.   
  
This means, all the views in the **Views**folder will use the the layout page specified in the ViewStart file in the Views, but then the views in the Home folder will use the layout page specified in the ViewStart file in the Home folder.   
  
**Please note :**If you want to use a layout file that is different from what is specified in \_ViewStart.cshtml, you can do so by setting the Layout property in an individual view. You can also set the Layout property to null, if you want your view to be rendered without a layout view.   
  
**Logic to select layout view**   
  
In an ASP.NET Core MVC application, we can have multiple layout views. Let's say for example, we have the following 2 layout views in our application.

1. \_AdminLayout.cshtml
2. \_NonAdminLayout.cshtml

The following logic in \_ViewStart.cshtml will select the layout view spending on the logged-in user role.

@{  
    if (User.IsInRole("Admin"))  
    {  
        Layout = "\_AdminLayout";  
    }  
    else  
    {  
        Layout = "\_NonAdminLayout";  
    }  
}