In this video we will discuss **views**. Please watch [Part 4](http://csharp-video-tutorials.blogspot.com/2013/04/controllers-in-mvc-application-part-4.html) of [MVC tutorial](http://www.youtube.com/user/kudvenkat/videos?view=1&flow=grid) before proceeding. Let's change the **Index**() method in HomeController to return a list of country names. Since country names are strings, return List<string>. Get rid of id and name parameters.  
public List<string> Index()  
{  
    return new List<string>()  
    {  
        "India",  
        "US",  
        "UK",  
        "Canada"  
    };  
}

**Run the application and notice that the output is not as expected.**  
System.Collections.Generic.List`1[System.String]  
  
**To correct this problem, let's add a view to our project.**  
**1.** Right click on the **Index**() function  
**2.** Click on **"Add View"**  
**3.** Notice that, the view name in **"Add View"** dialog box matches the name of the controller action method  
**4.** Select **"Razor"** as the view engine  
**5.** Leave the rest of the defaults as is, and click **"Add"** button  
  
Make the following modifications to the Index() function of the HomeController class, so that, the HomeController returns a view instead of List<string>.  
// Change the return type from List<string> to ActionResult  
public ActionResult Index()  
{  
    // Store the list of Countries in ViewBag.    
    ViewBag.Countries = new List<string>()  
    {  
        "India",  
        "US",  
        "UK",  
        "Canada"  
    };  
  
    // Finally return a view  
    return View();  
}  
  
We will discuss **ViewBag**& **ViewData**, and the differences between them in our next video session. For now, understand that, ViewBag & ViewData is a mechanism to pass data from the controller to the view.   
  
**Please Note:**To pass data from controller to a view, It's always a good practice to use strongly typed view models instead of using ViewBag & ViewData. We will discuss view models in a later video session.

**Now, copy and paste the following code in "Index.cshtml" view**  
@{  
    ViewBag.Title = "Countries List";  
}  
  
<h2>Countries List</h2>  
  
<ul>  
  
@foreach (string strCountry in ViewBag.Countries)  
{  
    <li>@strCountry</li>  
}  
  
</ul>  
  
**Please Note:** We use **"@"** symbol to switch between html and c# code. We will discuss razor views and their syntax in a later video session.