In this video, we will discuss opening URL's in a new window. Along the way, we also discuss using **UIHint**attribute. Please watch [Part 41](http://csharp-video-tutorials.blogspot.com/2013/06/part-41-using-datatype-and.html), before proceeding. We will be using the same example that we started in [Part 40](http://csharp-video-tutorials.blogspot.com/2013/06/part-40-using-displayname-displayformat.html) of this video series.

**Changes to Employee.cs class file in Models folder**  
[MetadataType(typeof(EmployeeMetaData))]  
public partial class Employee  
{  
}  
  
public class EmployeeMetaData  
{  
    [DataType(DataType.Url)]  
    public string PersonalWebSite { get; set; }  
}  
  
**Details action method in HomeController**  
public ActionResult Details(int id)  
{  
    SampleDBContext db = new SampleDBContext();  
    Employee employee = db.Employees.Single(x => x.Id == id);  
    return View(employee);  
}

**Code in Details.cshtml view**  
@model MVCDemo.Models.Employee  
  
@{  
    ViewBag.Title = "Details";  
}  
  
@Html.DisplayForModel()  
  
At this point, buid the application and navigate to **localhost/MVCDemo/Home/Details/1**. When you click on the personal website link, the target page will open in the same window.  
  
**If you want the page to open in a new window,**  
**1.** Right click on **"Views"** folder, and add **"Shared"** folder.   
**2.** Right click on **"Shared"** folder, and add **"DisplayTemplates"** folder.   
**3.** Right click on **DisplayTemplates** folder, and add a view. Set **"Url"** as the name and use Razor view engine.  
**4.** Copy and paste the following code in Url.cshtml view  
<a href="@ViewData.Model" target="\_blank">@ViewData.Model</a>  
  
That's it. Build the application and click on the link. Notice that, the page, now opens in a new window. The downside of this approach is that, from now on all the links, will open in a new window. To overcome this, follow these steps.  
**1.** Rename **Url.cshtml** to **OpenInNewWindow.cshtml**  
**2.** Decorate **"PersonalWebSite"** property in **EmployeeMetaData** class with **UIHint**attribute and specify the name of the template to use. In our case, the name of the template is **"OpenInNewWindow"**.  
public class EmployeeMetaData  
{  
    [DataType(DataType.Url)]  
    [UIHint("OpenInNewWindow")]  
    public string PersonalWebSite { get; set; }  
}  
  
So, **UIHint**attribute is used to specify the name of the template to use to display the data field.