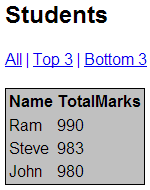
**ASP.NET AJAX enables a Web application to retrieve data from the server asynchronously and to update portions of the existing page.** So these, partial page updates make web application more responsive and hence improves user experience.   
  
   
  
In this video, let's discus using **Ajax.ActionLink**helper. By the end of this video, we should be able to load either  
**1.** All students or  
**2.** Top 3 students or  
**3.** Bottom 3 students   
depending on the link we have clicked.  
   
  
**Here are the steps to achieve this**  
**Step 1:**Sql script to create and populate table tblStudents  
Create table tblStudents  
(  
 Id int identity primary key,  
 Name nvarchar(50),  
 TotalMarks int  
)  
  
Insert into tblStudents values ('Mark', 900)  
Insert into tblStudents values ('Pam', 760)  
Insert into tblStudents values ('John', 980)  
Insert into tblStudents values ('Ram', 990)  
Insert into tblStudents values ('Ron', 440)  
Insert into tblStudents values ('Able', 320)  
Insert into tblStudents values ('Steve', 983)  
Insert into tblStudents values ('James', 720)  
Insert into tblStudents values ('Mary', 870)  
Insert into tblStudents values ('Nick', 680)  
  
**Step 2:** Create an ado.net entity **data model**using table **tblStudents.**Upon creating the entity model, change the name of the entity to Student. Save changes and build the solution, so that the Student entity class gets compiled.  
  
**Step 3:**Add **"Shared"**folder (if it doesn't already exists) in **"Views"**folder. Right click on **"Shared"**folder and add a partial view, with name = **\_Student.cshtml**.   
@model IEnumerable<MVCDemo.Models.Student>  
  
<table style="border:1px solid black; background-color:Silver">  
<tr>  
    <th>  
        @Html.DisplayNameFor(model => model.Name)  
    </th>  
    <th>  
        @Html.DisplayNameFor(model => model.TotalMarks)  
    </th>  
</tr>  
  
@foreach (var item in Model)   
{  
    <tr>  
        <td>  
            @Html.DisplayFor(modelItem => item.Name)  
        </td>  
        <td>  
            @Html.DisplayFor(modelItem => item.TotalMarks)  
        </td>  
    </tr>  
}  
</table>  
  
**Step 4:** Right click on the **"Controllers"**folder and add a controller with the following settings  
**1.** Controller Name = HomeController  
**2.** Template = Empty MVC controller  
  
Copy and paste the following code. Please make sure to include MVCDemo.Models namespace.  
public class HomeController : Controller  
{  
    // Create an instance of DatabaseContext class  
    DatabaseContext db = new DatabaseContext();  
  
    public ActionResult Index()  
    {  
        return View();  
    }  
  
    // Return all students  
    public PartialViewResult All()  
    {  
        List<Student> model = db.Students.ToList();  
        return PartialView("\_Student", model);  
    }  
  
    // Return Top3 students  
    public PartialViewResult Top3()  
    {  
        List<Student> model = db.Students.OrderByDescending(x => x.TotalMarks).Take(3).ToList();  
        return PartialView("\_Student", model);  
    }  
  
    // Return Bottom3 students  
    public PartialViewResult Bottom3()  
    {  
        List<Student> model = db.Students.OrderBy(x => x.TotalMarks).Take(3).ToList();  
        return PartialView("\_Student", model);  
    }  
}  
  
**Step 5:** Right click on the **"Views"**folder and add a folder with **name = "Home"**. Right click on the **"Home"**folder and add a view with Name = **"Index".**   
**Points to remember:**  
**a)** For AJAX to work, jquery and jquery.unobtrusive-ajax javascript files need to be referenced. Reference to jquery.unobtrusive-ajax.js file should be after jquery.js reference.  
**b)** Ajax.ActionLink() html helper has several overloaded versions. We have used the following version

ActionLink(string linkText, string actionName, AjaxOptions ajaxOptions);  
**First parameter :** is the link text that we see on the user interface  
**Second parameter :**is the name of the action method that needs to be invoked, when the link is cicked.  
**Third parameter :**is the ajaxOptions object.   
  
<div style="font-family:Arial">  
  
<script src="~/Scripts/jquery-1.7.1.min.js" type="text/javascript"></script>  
<script src="~/Scripts/jquery.unobtrusive-ajax.min.js" type="text/javascript"></script>  
  
<h2>Students</h2>  
  
@Ajax.ActionLink("All", "All",   
    new AjaxOptions   
    {  
      HttpMethod = "GET", // HttpMethod to use, GET or POST  
      UpdateTargetId = "divStudents", // ID of the HTML element to update  
      InsertionMode = InsertionMode.Replace // Replace the existing contents  
    })  
  
<span style="color:Blue">|</span>  
  
@Ajax.ActionLink("Top 3", "Top3",   
    new AjaxOptions   
    {  
      HttpMethod = "GET", // HttpMethod to use, GET or POST  
      UpdateTargetId = "divStudents", // ID of the HTML element to update  
      InsertionMode = InsertionMode.Replace // Replace the existing contents  
    })  
  
<span style="color:Blue">|</span>  
  
@Ajax.ActionLink("Bottom 3", "Bottom3",   
    new AjaxOptions   
    {  
      HttpMethod = "GET", // HttpMethod to use, GET or POST  
      UpdateTargetId = "divStudents", // ID of the HTML element to update  
      InsertionMode = InsertionMode.Replace // Replace the existing contents  
    })  
  
<br /><br />  
<div id="divStudents">  
</div>  
</div>