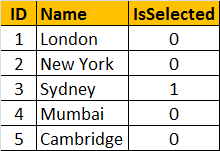
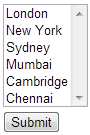
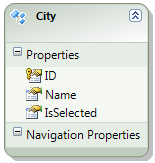
In this video we will discuss **implementing ListBox in asp.net mvc**. We will be using table **"tblCity"** for this demo.   


Please refer to [Part 38](http://csharp-video-tutorials.blogspot.com/2013/06/part-38-checkboxlist-in-aspnet-mvc.html), if you need the sql script to create and populate this table. Please watch [Part 38](http://csharp-video-tutorials.blogspot.com/2013/06/part-38-checkboxlist-in-aspnet-mvc.html), before proceeding.  
  
The generated listbox should be as shown below. Notice that for each city in table "tblCity", there is an entry in the listbox.   


For this demo, we will be using the **"City"** entity that we generated using **ADO.NET entity framework**, in [Part 38](http://csharp-video-tutorials.blogspot.com/2013/06/part-38-checkboxlist-in-aspnet-mvc.html).   
   
  
The first step is to create a **ViewModel** class. In asp.net mvc, view model's are used as techniques to shuttle data between the controller and the view. Right click on the "Models" folder, and add a class file with **name=CitiesViewModel.cs**. Copy and paste the following code. This class is going to be the Model for the view.  
public class CitiesViewModel  
{  
    public IEnumerable<string> SelectedCities { get; set; }  
    public IEnumerable<SelectListItem> Cities { get; set; }  
}  
  
Right click on the **"Controllers"** folder, and add a **"HomeController"**. Include the following 2 namespaces in **"HomeController"**  
using MVCDemo.Models;  
using System.Text;  
  
**Copy and paste the following code.**  
[HttpGet]  
public ActionResult Index()  
{  
    SampleDBContext db = new SampleDBContext();  
    List<SelectListItem> listSelectListItems = new List<SelectListItem>();  
              
    foreach (City city in db.Cities)  
    {  
        SelectListItem selectList = new SelectListItem()  
        {  
            Text = city.Name,   
            Value = city.ID.ToString(),   
            Selected = city.IsSelected  
        };  
        listSelectListItems.Add(selectList);  
    }  
  
    CitiesViewModel citiesViewModel = new CitiesViewModel()  
    {  
        Cities = listSelectListItems  
    };  
  
    return View(citiesViewModel);  
}  
  
[HttpPost]  
public string Index(IEnumerable<string> selectedCities)  
{  
    if (selectedCities == null)  
    {  
        return "No cities selected";  
    }  
    else  
    {  
        StringBuilder sb = new StringBuilder();  
        sb.Append("You selected - " + string.Join(",", selectedCities));  
        return sb.ToString();  
    }  
}  
  
Right click on the **"Index"** action method in **"HomeController"** and select **"Add View"**from the context menu. Set  
View Name = Index  
View Engine = Razor  
and click "Add".  
  
**Copy and paste the following code in "Index.cshtml"**  
@model MVCDemo.Models.CitiesViewModel  
  
@{  
    ViewBag.Title = "Index";  
}  
  
<div style="font-family:Arial">  
<h2>Index</h2>  
@using (Html.BeginForm())  
{  
    @Html.ListBoxFor(m => m.SelectedCities, Model.Cities, new { size = 4 })  
    <br />  
    <input type="submit" value="Submit" />  
}  
</div>  
  
**Note:** To select multiple items from the listbox, hold down the CTRL Key.