

# Presetting List Entries with CascadingDropDown

Christian Wenz

## Overview

The CascadingDropDown control in the AJAX Control Toolkit extends a DropDownList control so that changes in one DropDownList loads associated values in another DropDownList. (For instance, one list provides a list of US states, and the next list is then filled with major cities in that state.) With a little bit of code it is possible that a list element is preselected once the data has been dynamically loaded.

## Steps

In order to activate the functionality of ASP.NET AJAX and the Control Toolkit, the **ScriptManager** control must be put anywhere on the page (but within the **<form>** element):

```
<asp:ScriptManager ID="asm" runat="server" />
```

Then, a DropDownList control is required:

```
<div>
  Vendor: <asp:DropDownList ID="VendorsList" runat="server" />
</div>
```

For this list, a CascadingDropDown extender is added, providing web service URL and method information:

```
<ajaxToolkit:CascadingDropDown ID="ccd1" runat="server"
  ServicePath="CascadingDropDown2.cs.aspx"
  ServiceMethod="GetVendors"
  TargetControlID="VendorsList" Category="Vendor" />
```

The CascadingDropDown extender then asynchronously calls a web service with the following method signature:

```
public CascadingDropDownNameValue[] MethodNameHere(string
  knownCategoryValues, string category)
```

The method returns an array of type CascadingDropDown value. The type's constructor expects first the list entry's caption and then the value (HTML **value** attribute). If the third argument is set to true, the list element is automatically selected in the browser.

```
<%@ WebService Language="C#" Class="CascadingDropDown2" %>

using System.Web.Script.Services;
using AjaxControlToolkit;
using System;
using System.Web;
```

```

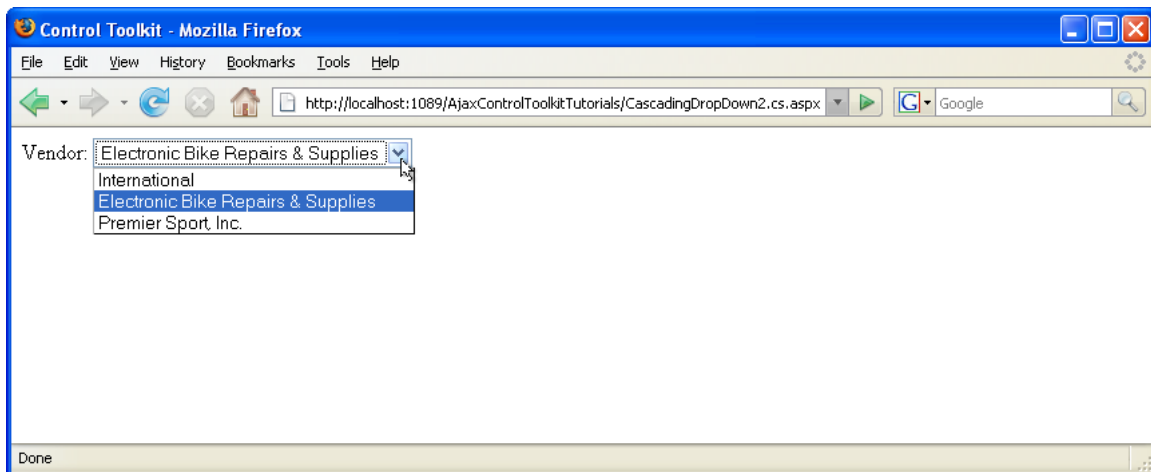
using System.Web.Services;
using System.Web.Services.Protocols;
using System.Collections.Generic;

[ScriptService]
public class CascadingDropDown2 : System.Web.Services.WebService
{
    [WebMethod]
    public CascadingDropDownNameValue[] GetVendors(string
        knownCategoryValues, string category)
    {
        List<CascadingDropDownNameValue> l = new
        List<CascadingDropDownNameValue>();

        l.Add(new CascadingDropDownNameValue(
            "International", "1"));
        l.Add(new CascadingDropDownNameValue(
            "Electronic Bike Repairs & Supplies", "2", true));
        l.Add(new CascadingDropDownNameValue(
            "Premier Sport, Inc.", "3"));
        return l.ToArray();
    }
}

```

Loading the page in the browser will fill the dropdown list with three vendors, the second one being preselected.



**The list is filled and preselected automatically**