

# Filling a List Using 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.) The first challenge to solve is to actually fill a dropdown list using this control.

## 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. It will send an asynchronous request to a web service which will then return a list of entries to be displayed in the list. For this to work, the following CascadingDropDown attributes need to be set:

- **ServicePath**: URL of a web service delivering the list entries
- **ServiceMethod**: Web method delivering the list entries
- **TargetControlID**: ID of the dropdown list
- **Category**: Category information that is submitted to the web method when called
- **PromptText**: Text displayed when asynchronously loading list data from the server

Here is the markup for the **CascadingDropDown** element. The only difference between C# and VB is the name of the associated web service:

```
<ajaxToolkit:CascadingDropDown ID="ccd1" runat="server"
  ServicePath="CascadingDropdown0.cs.aspx"
  ServiceMethod="GetVendors"
  TargetControlID="VendorsList" Category="Vendor" />
```

The JavaScript code coming from the **CascadingDropDown** extender calls a web service method with the following signature:

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

So the important aspect is that the method needs to return an array of type **CascadingDropDownNameValue** (defined by the ASP.NET AJAX Control Toolkit). In the **CascadingDropDownNameValue** constructor, first the list entry's text and then its value must be provided, just as **<option value="VALUE">NAME</option>** would do in HTML. Here is some sample data:

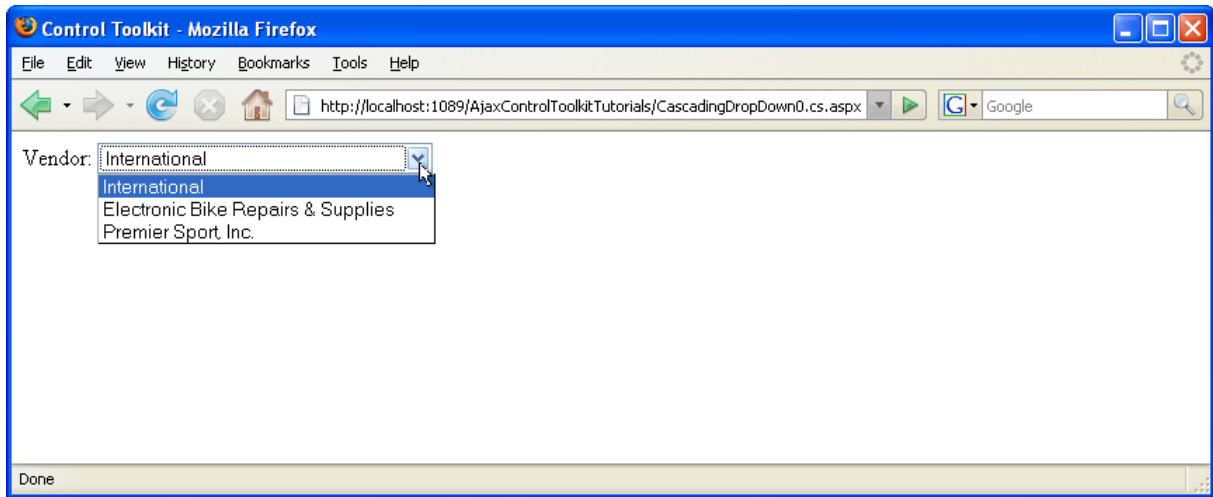
```
<%@ WebService Language="C#" Class="CascadingDropDown0" %>

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 CascadingDropDown0 : 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"));
        l.Add(new CascadingDropDownNameValue(
            "Premier Sport, Inc.", "3"));
        return l.ToArray();
    }
}
```

Loading the page in the browser will trigger the list to be filled with three vendors.



**The list is filled automatically**