2 filtros, y un hidden field

Client:

<asp:DropDownList ID="ddlClient" runat="server" ></asp:DropDownList>

Sites:

<asp:DropDownList ID="ddlSites" runat="server" ></asp:DropDownList>

<input type="hidden" id="HDClient" value="">

Cargue al inicio clientes

Lo carga de DataTable que puede ser leida de cache, o de DAO

La lista tiene repetido los clientes y los agrupa, luego agrega “Select One” “-1” al inicio

private void LoadClients(DataTable dt)

{

var newSort2 = from row in dt.AsEnumerable()

where row.Field<object>("client\_id") != null

orderby row.Field<object>("client\_descr")

group row by new

{

ID = Convert.ToString(row.Field<object>("client\_id")),

nombre = row.Field<string>("client\_descr")

} into grp

select new

{

ids = Convert.ToString(grp.Key.ID),

nombres = grp.Key.nombre

};

ddlClient.DataSource = newSort2;

ddlClient.DataValueField = "ids";

ddlClient.DataTextField = "nombres";

ddlClient.DataBind();

ListItem li = new ListItem("Select One", "-1");

ddlClient.Items.Insert(0, li);

}

Al momento de seleccionar Client, cargue los Sitios

$(function() {

$("#ddlClient").focus();

$("#ddlSites").hide();

$('#ddlClient').bind("change keyup", function () {

loadSites($(this).val());

});

});

// Perform AJAX call to function FetchEmployeeCars [Web Method]

function loadSites(selectedItem) {

if (selectedItem == "0") {

$("#ddlSites").fadeOut("slow"); // fadeOut

return;

}

$("#ddlSites").fadeIn("slow"); // fadeIn

// HIDDEN FIELD STORE CLIENT

document.getElementById("HDClient").value = selectedItem;

$.ajax({

type: "POST",

url: "Default3.aspx/FetchSites",

data: "{id: " + selectedItem + "}",

contentType: "application/json; charset=utf-8",

dataType: "json",

async: true,

success: function (data) {

LoadComboSites(data.d);

}

});

}

// Once it has the data then load the combo

function LoadComboSites (data) {

// Clean Cars Combo Box

var Drop1 = document.getElementById("ddlSites");

Drop1.options.length = 0;

for (var i = 0; i < data.length; i++) {

var opt = document.createElement("option");

opt.text = data[i].Descr;

opt.value = data[i].Id;

// Add an Option object to Drop Down/List Box

Drop1.options.add(opt);

}

}

Code Behind

[WebMethod]

public **static** List<SiteEntity> FetchSites(int id)

{

var item1 = new SiteEntity();

return item1.DBLoadSites(id);

}

Una vez cargado filtros, para traer datos, cargue los HiddenFields

var x = document.getElementById("HDClient").value;

**= = = = = = = = = = = = = = = = = = = = = = = =**

Otra opcion de cargar sites

// LOAD SITES

function Load\_Sites() {

var myDDL = document.getElementById('cplMianDisplay\_ddlClient');

var cliente = myDDL.value;

if (cliente == "-1") return;

// Si escogio cliente, borre la primera file que dice “Select One”, -1

if (myDDL[0].value == -1) {

myDDL.remove(0);

}

// HIDDEN FIELD STORE CLIENT

document.getElementById("HDClient").value = cliente;

PageMethods.Load\_Sitios(cliente, OnSuccess, OnFail);

}

function OnSuccess(response) {

var myDDL = document.getElementById('cplMianDisplay\_ddlSites');

myDDL.options.length = 0;

for (var i in response) {

var option = document.createElement('option');

option.text = response[i].names;

option.value = response[i].ids;

myDDL.add(option);

}

$("#cplMianDisplay\_listTeam").empty();

$("#cplMianDisplay\_listTeam").multiselect('refresh');

}

function OnFail(err) {

alert(err.get\_message());

}

**CODE BEHIND**

[System.Web.Services.WebMethod]

public static List<SupvListEntity> Load\_Sitios(string cliente)

{

System.Web.Caching.Cache cache;

cache = HttpContext.Current.Cache;

DataTable dt = (DataTable)cache["TeamLeaders"];

var newSort2 = from row in dt.AsEnumerable()

where row.Field<object>("agent\_site\_id") != null

&& row.Field<string>("client\_id") == cliente

orderby row.Field<object>("agent\_site\_descr")

group row by new

{

ID = Convert.ToString(row.Field<object>("agent\_site\_id")),

nombre = row.Field<string>("agent\_site\_descr")

} into grp

select new

{

ids = Convert.ToString(grp.Key.ID),

names = grp.Key.nombre

};

List<SupvListEntity> listado = new List<SupvListEntity>();

listado.Add(new SupvListEntity() { ids = "-1", names = "Select One" });

foreach (var sitios in newSort2)

{

listado.Add(new SupvListEntity() { ids = sitios.ids, names = sitios.names });

}

return listado;

// ddlSites.Items.Clear();

// ddlSites.DataSource = newSort2;

// ddlSites.DataValueField = "ids";

// ddlSites.DataTextField = "nombres";

// ddlSites.DataBind();

//ListItem li = new ListItem("Select One", "-1");

//ddlSites.Items.Insert(0, li);

//listTeam.Items.Clear();

}