**New class**

public class Generico

{

public string ids { get; set; }

public string names { get; set; }

}

**Global Variable**

List<Generico> ListMetrics = new List<Generico>();

**Load Data into List from MultiSelect control**

ListMetrics.Clear();

foreach (ListItem item in listMetric.Items)

{

if (item.Selected)

{

ListMetrics.Add(new Generico() { ids = item.Value.ToString(), names = item.Text });

}

}

**Load information in RowDataBound**

protected void Threshold\_RowDataBound(object sender, GridViewRowEventArgs e)

{

if (e.Row.RowType == DataControlRowType.DataRow)

{

DataRowView rowView = (DataRowView)e.Row.DataItem;

string **metric\_id** = rowView["metric\_id"].ToString();

var result2 = ListMetrics.FirstOrDefault(a => a.ids == **metric\_id**);

if (result != null)

e.Row.Cells[2].Text = result2.**names**.ToString();

}

}

**LOADING INTO GRID**

// load data set, luego inserte lo faltante

ds = Load\_TL\_Metric(ds);

Threshold.DataSource = ds;

Threshold.DataBind();

// guarde information para posterior usarla en sort

Session["Threshold"] = ds.Tables[0];

Threshold.Visible = true;

// cargue datos faltantes en dataset

private DataSet Load\_TL\_Metric(DataSet ds)

{

int rows = ds.Tables[0].Rows.Count;

string supv\_id, metric\_id;

for (int i =0;i<rows;i++)

{

metric\_id = ds.Tables[0].Rows[i]["metric\_id"].ToString();

var result2 = = ListMetrics.FirstOrDefault(a => a.ids == metric\_id);

if (result2 != null) ds.Tables[0].Rows[i]["met\_name"] = result2.names.ToString();

}

return ds;

}