|  |  |
| --- | --- |
| <asp:Label ID="lblVin" runat="server" Width="100px" Text="VIN: "></asp:Label>  <asp:TextBox ID="txtVin" runat="server"></asp:TextBox>  <br />  <asp:Label ID="lblMake" runat="server" Width="100px" Text="Make: "></asp:Label>  <asp:TextBox ID="txtMake" runat="server"></asp:TextBox>  <br />  <asp:Label ID="lblModel" runat="server" Width="100px" Text="Model: "></asp:Label>  <asp:TextBox ID="txtModel" runat="server"></asp:TextBox>  <br />  <asp:Label ID="lblYear" runat="server" Width="100px" Text="Year: "></asp:Label>  <asp:DropDownList ID="ddlYear" runat="server">  <asp:ListItem Text="All Years" Value="0" />  <asp:ListItem Text="> 1995" Value="1995" />  <asp:ListItem Text="> 2000" Value="2000" />  <asp:ListItem Text="> 2005" Value="2005" />  </asp:DropDownList>  <br />  <asp:Label ID="lblCost" runat="server" Width="100px" Text="Cost: "></asp:Label>  <asp:DropDownList ID="ddlCost" runat="server">  <asp:ListItem Text="Any Cost" Value="0" />  <asp:ListItem Text="> 5000" Value="5000" />  <asp:ListItem Text="> 20000" Value="20000" />  </asp:DropDownList>  <br />  <asp:Label ID="lblSort" runat="server" Width="100px" Text="Sort Order: "></asp:Label>  <asp:DropDownList ID="ddlSort" runat="server">  <asp:ListItem Text="" />  <asp:ListItem Text="VIN" />  <asp:ListItem Text="Make" />  <asp:ListItem Text="Model" />  <asp:ListItem Text="Year" />  <asp:ListItem Text="Cost" />  </asp:DropDownList>  <br />  <br />  <asp:Button ID="btnExecute" runat="server" Text="Execute" />  <br />  <asp:GridView ID="gvVehicles" runat="server">  </asp:GridView> |  |

4 files for class, Vehicle.cs, Car.cs, Boat.cs, Truck.cs

|  |  |  |  |
| --- | --- | --- | --- |
| Vehicle (master class | Car.cs | Truck | Boat |
| namespace VehicleProject  {  public class Vehicle  {  public string VIN { get; set; }  public string Make { get; set; }  public string Model { get; set; }  public int Year { get; set; }  public decimal Cost { get; set; }  }  } | namespace VehicleProject  {  public class Car : Vehicle  {  }  } | namespace VehicleProject  {  public class Truck : Vehicle  {  }  } | namespace VehicleProject  {  public class Boat : Vehicle  {  }  } |

namespace VehicleProject

{

public partial class \_Default : System.Web.UI.Page

{

private List<Vehicle> vehicles = new List<Vehicle>();

protected void Page\_Load(object sender, EventArgs e)

{

if (vehicles.Count == 0)

{

vehicles.Add(new Truck {VIN = "AAA123", Make = "Ford", Model = "F-250", Cost = 2000, Year = 1998});

vehicles.Add(new Truck {VIN = "ZZZ123", Make = "Ford", Model = "F-150", Cost = 10000, Year = 2005});

vehicles.Add(new Car {VIN = "FFF123", Make = "VW", Model = "Bug", Cost = 2500, Year = 1997});

vehicles.Add(new Boat {VIN = "LLL123", Make = "SeaRay",Model = "Signature", Cost = 12000, Year = 1995});

vehicles.Add(new Car {VIN = "CCC123", Make = "BMW", Model = "Z-3", Cost = 21000, Year = 2005});

vehicles.Add(new Car {VIN = "EEE123", Make = "Ford", Model = "Focus", Cost = 15000, Year = 2008});

vehicles.Add(new Boat {VIN = "QQQ123", Make = "Craft", Model = "BowRider", Cost = 102000, Year = 1945});

vehicles.Add(new Truck {VIN = "PPP123", Make = "Ford", Model = "F-250", Cost = 1000, Year = 1980});

vehicles.Add(new Car {VIN = "TTT123", Make = "Dodge", Model = "Viper", Cost = 95000, Year = 2007});

vehicles.Add(new Car {VIN = "DDD123", Make = "Mazda", Model = "Miata", Cost = 20000, Year = 2005});

}

var result = from v in vehicles

.Where(v => v.VIN.StartsWith(txtVin.Text))

.Where(v => v.Make.StartsWith(txtMake.Text))

.Where(v => v.Model.StartsWith(txtModel.Text))

.Where(v => v.Cost > Decimal.Parse(ddlCost.SelectedValue))

.Where(v => v.Year > int.Parse(ddlYear.SelectedValue));

result = **SetOrder**(ddlSort.SelectedValue, result);

gvVehicles.DataSource = **result.Select((v, i)=> new {Index = i, v.VIN, v.Make, v.Model, v.Year, v.Cost});**

gvVehicles.DataBind();

}

private IEnumerable<Vehicle> **SetOrder**( string order, IEnumerable<Vehicle> query)

{

switch (order)

{

case "VIN": return query.OrderBy(v => v.VIN);

case "Make": return query.OrderBy(v => v.Make);

case "Model": return query.OrderBy(v => v.Model);

case "Year": return query.OrderBy(v => v.Year);

case "Cost": return query.OrderBy(v => v.Cost);

default: return query;

}

}

}

}