

HelloWorld!

Web application

What's in this session?

1. Question and Data
2. Create ASP.NET project
3. Add NuGet packages
4. Add using name space
5. Create data set input/output scheme
6. Set data set path
7. Load data

8. Add algorithm
9. Train the model
10. Predict single item

Question and Data

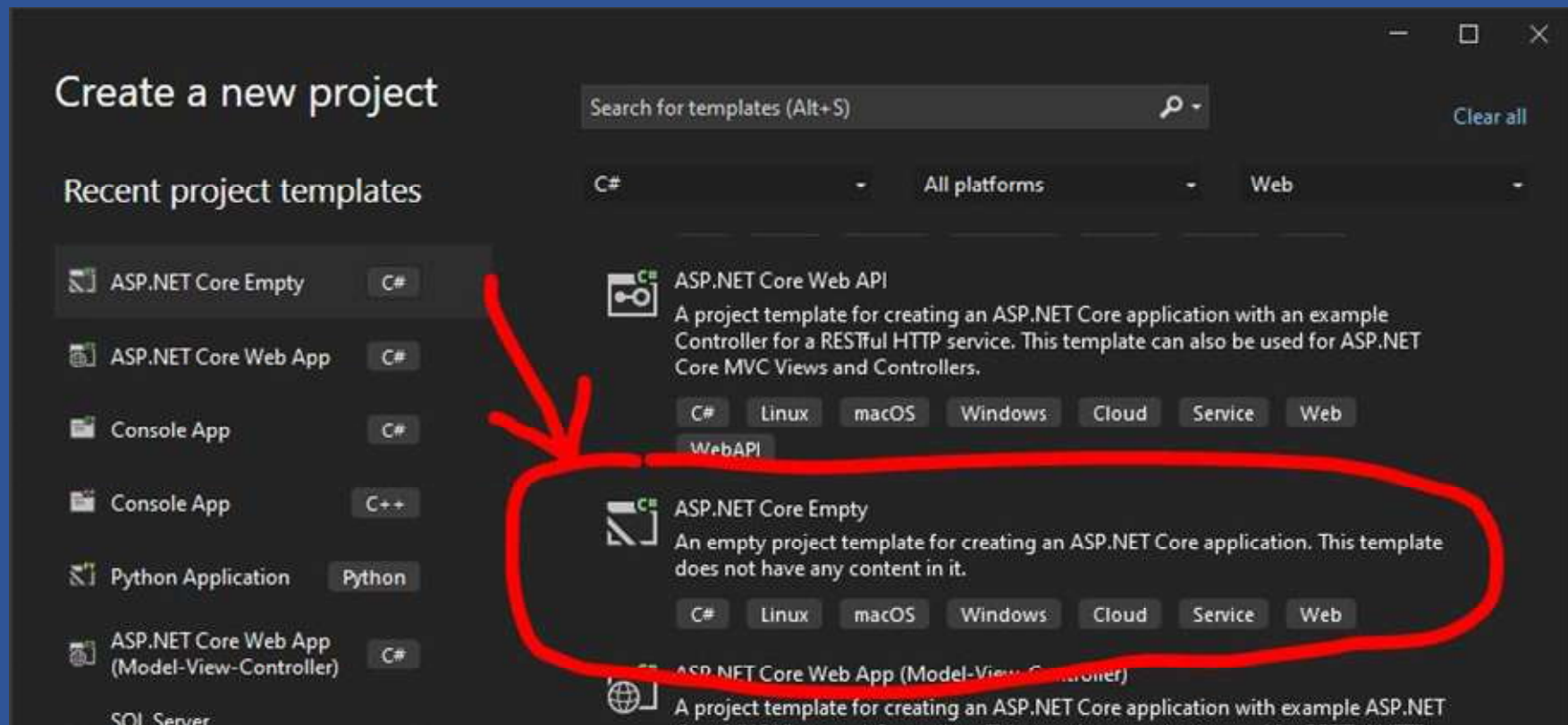
Question: How much is my diamond costed?

Dataset:


<https://github.com/laploy/ML.NET/blob/master/HelloWorld/diamondsmall.csv>

Create New Project

ASP.NET Core Empty

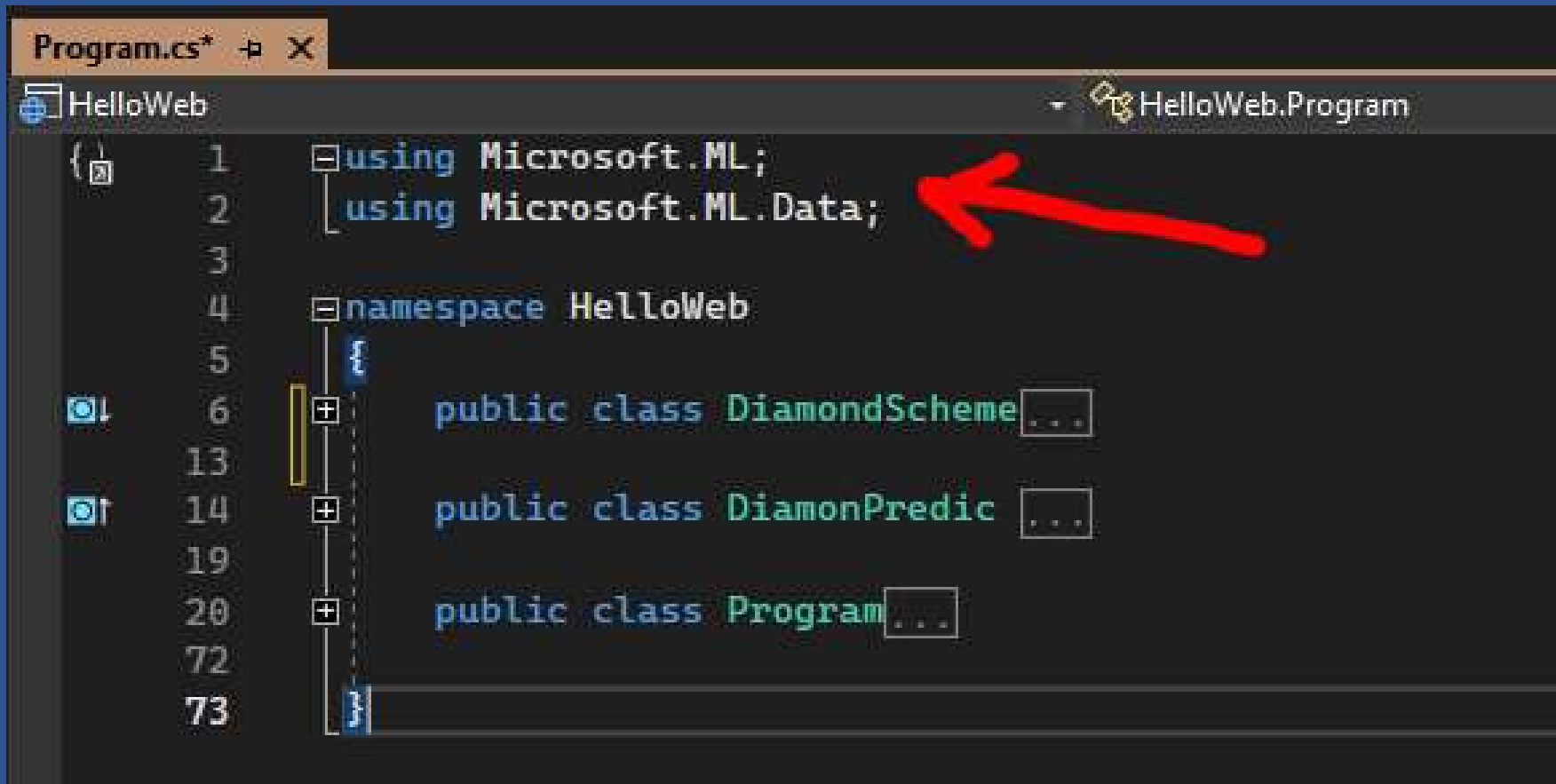


Add NuGet Packages

 **Microsoft.ML** by Microsoft ↓ v1.1.0

ML.NET is a cross-platform open-source machine learning framework which makes ma...

Add Namespace



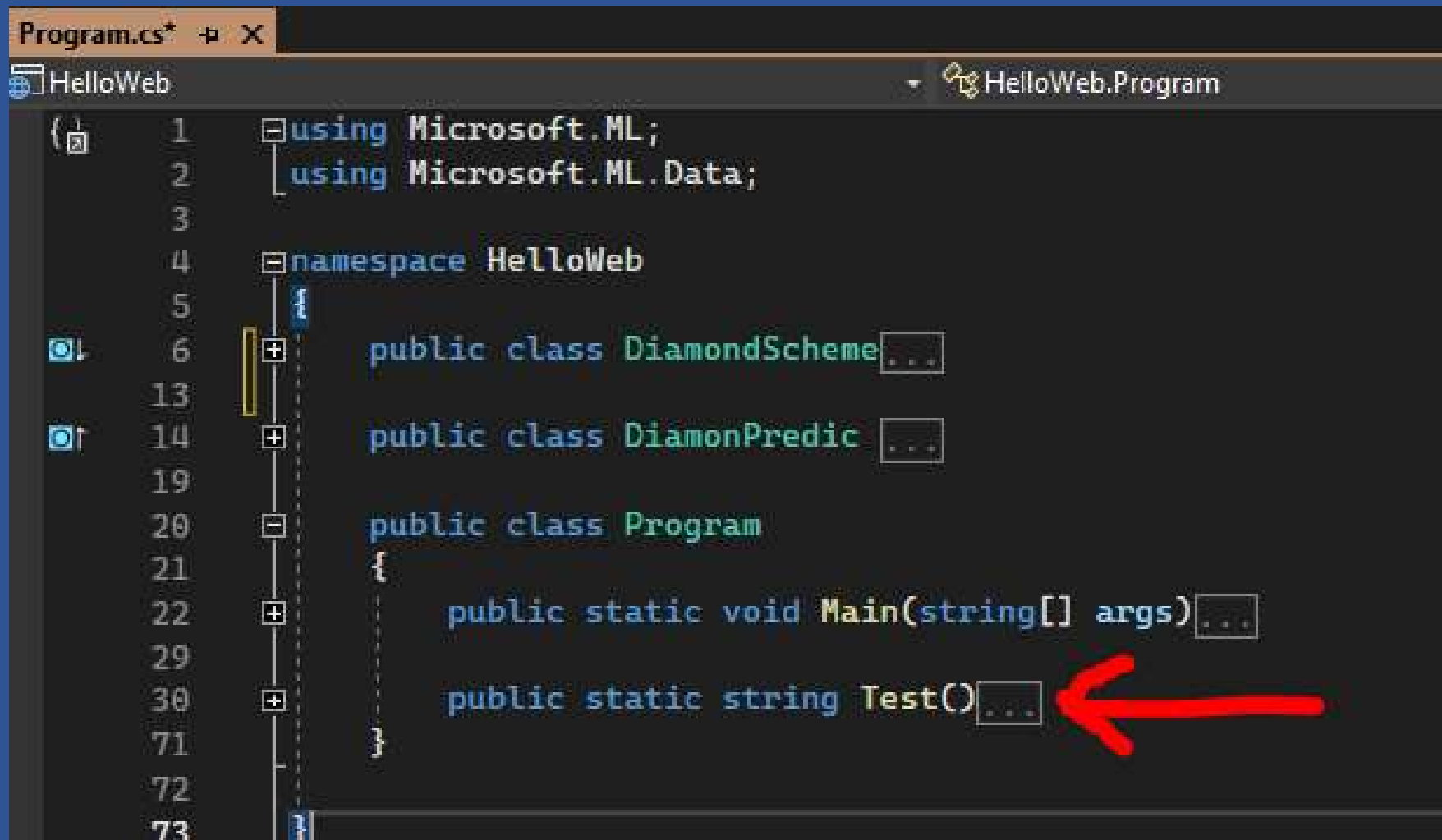
```
Program.cs* [X]
HelloWeb [HelloWeb.Program]
{
1  using Microsoft.ML;
2  using Microsoft.ML.Data;
3
4  namespace HelloWeb
5  {
6      public class DiamondScheme ...
13
14      public class DiamonPredic ...
19
20      public class Program ...
72
73  }
```

Create data scheme

```
public class DiamondScheme
{
    [LoadColumn(0)]
    public float Size { get; set; }
    [LoadColumn(1)]
    public float Price { get; set; }
}

public class DiamonPredic: DiamondScheme
{
    [ColumnName("Score")]
    public float PricePredict { get; set; }
}
```


Add method Test()



```
Program.cs* [X]
HelloWeb
HelloWeb.Program

1  using Microsoft.ML;
2  using Microsoft.ML.Data;
3
4  namespace HelloWeb
5  {
6      public class DiamondScheme ...
13
14      public class DiamonPredic ...
19
20      public class Program
21      {
22          public static void Main(string[] args) ...
29
30          public static string Test() ...
71      }
72
73  }
```

In FormLoad

Add data path, context, IDataView, and Debug

```
string trainDataPath = @"E:\ml\diamondSmall.csv";

// create context
MLContext mlContext = new MLContext();

// Load train data
IDataView trainData = mlContext.Data.LoadFromTextFile<DiamondScheme>
    (trainDataPath, hasHeader: true, separatorChar: ',');

// debug watch
var myPreview = trainData.Preview();
```

Create pipe line / set algorithm / train

```
// create pipe line
var pipeline = mlContext.Transforms.Concatenate
    ("Features", new[] { "Size" })

// set training algorithm
    .Append(mlContext.Regression.Trainers.Sdca
        (labelColumnName: "Price",
         maximumNumberOfIterations:100));

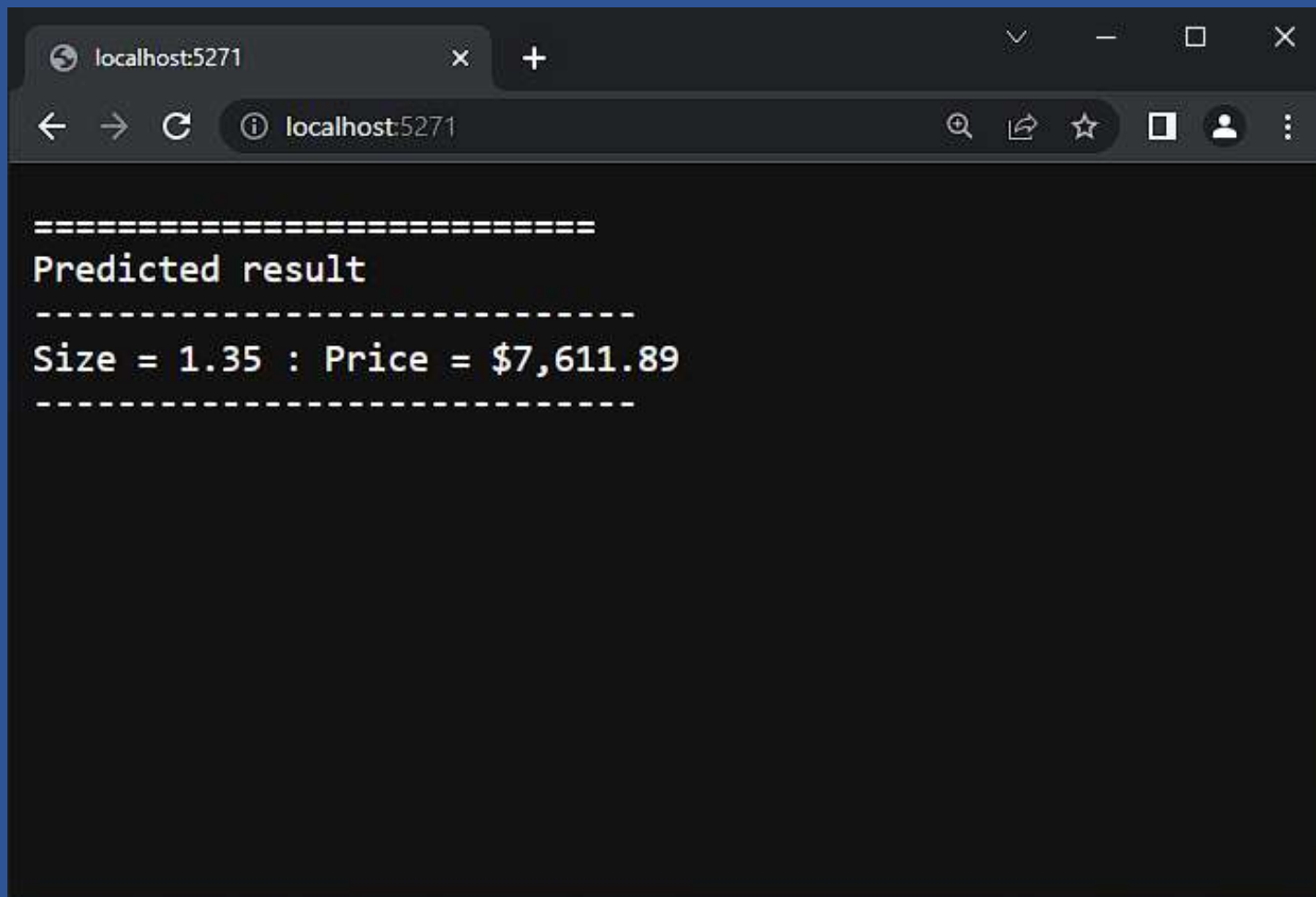
// Train model
var myModel = pipeline.Fit(trainData);
```

Make predict / Show result

```
// Make a prediction
var mySize = new DiamondScheme() { Size = 1.35F };
var myPrice = mlContext.Model.CreatePredictionEngine
    <DiamondScheme, DiamonPredic>
    (myModel).Predict(mySize);

// Show result
string s = "=====\n";
s+= "Predicted result\n";
s += "-----\n";
s += $"Size = {mySize.Size} " +
    $"Price = {myPrice.PricePredict:C}\n";
s += "-----";
return s;
```

Run to examine the result



A screenshot of a web browser window with a dark theme. The address bar shows 'localhost:5271'. The page content is displayed in a monospaced font and is framed by dashed lines. It shows the predicted result of a machine learning model, which is a house size and price.

```
=====
Predicted result
-----
Size = 1.35 : Price = $7,611.89
-----
```

What's next?

