LINQ to XML Lab

Setup

Download Songs.xml File

Download the **Songs.xml** file from https://github.com/PaulDSheriff/VSLive-Trainings

Lab 1: Build New Project & Get All Songs

Open a new instance of Visual Studio and create a new Console Application named **XMLLinqLab**.

Right mouse-click on the XMLLinqLab project and create a new folder named **EntityClasses**.

Right mouse-click on the **EntityClasses** folder and add a new class named **Song**.

Add the following code into this new file.

```
#nullable disable

using System.ComponentModel.DataAnnotations.Schema;

namespace Samples;

[Table("Songs", Schema = "dbo")]
public partial class Song
{
  public int SongId { get; set; }
  public string SongName { get; set; }
  public string Artist { get; set; }
  public string Album { get; set; }
  public int? GenreId { get; set; }
  public int? KindId { get; set; }
  public string TrackNumber { get; set; }
  public int? Rating { get; set; }
  public int? Year { get; set; }
  public DateTime? ReleaseDate { get; set; }
  public string Size { get; set; }
  public int? Plays { get; set; }
  public DateTime? DateAdded { get; set; }

  #region ToString Override
  public override string ToString() {
    return $"{SongName} ({SongId})";
  }
  #endregion
}
```

Right mouse-click on the XMLLinqLab project and create a new folder named **Xml**.

Right mouse-click on the **Xml** folder and copy the **Songs.xml** file into this folder.

Right mouse-click on the XMLLinqLab project and create a new folder named **HelperClasses**.

Right mouse-click on the **HelperClasses** folder and create a new class named **FileNameHelper.cs**

Add the following code into this new file

```
namespace Samples;

/// <summary>
/// Creates path to XML file and sets the SongsFile property
/// </summary>
public static class FileNameHelper {
  static FileNameHelper() {
    //string path = @"..\..\..\..\..\Xml\";
    // Get path to current directory and remove everything from
\bin\ forward
    // Then add on the folder where the database is located
    string path = Directory.GetCurrentDirectory();
    if (path.Contains("\\bin")) {
      path = path.Substring(0, path.LastIndexOf("bin") - 1);
    }
    path += @"\Xml\";

    SongsFile = $"{path}Songs.xml";
  }

  public static string SongsFile { get; set; }
}
```

Right mouse-click on the **HelperClasses** folder and create a new class named **XmlHelper.cs**

Add the following code into this new file

```
#nullable disable

using System.Globalization;
using System.Xml.Linq;

namespace Samples;

/// <summary>
/// Extension methods to help process XML files easily
/// </summary>
public static class XmlHelper
{
  #region GetAs<T> Method
  public static T GetAs<T>(this XElement elem, string name, T
defaultValue = default)
  {
    T ret = defaultValue;

    if (elem != null && elem.Element(name) != null &&
!string.IsNullOrEmpty(elem.Element(name).Value))
    {
      // Cast to Return Data Type
      // NOTE: ChangeType can not cast to a Nullable type
      ret = (T)Convert.ChangeType(elem.Element(name).Value,
typeof(T), CultureInfo.InvariantCulture);
    }

    return ret;
  }
  #endregion

  #region GetAttrAs<T> Method
  public static T GetAttrAs<T>(this XElement elem, string name, T
defaultValue = default)
  {
    T ret = defaultValue;

    if (elem != null && elem.Attribute(name) != null &&
!string.IsNullOrEmpty(elem.Attribute(name).Value))
    {
      // Cast to Return Data Type
      // NOTE: ChangeType can not cast to a Nullable type
      ret = (T)Convert.ChangeType(elem.Attribute(name).Value,
typeof(T), CultureInfo.InvariantCulture);
    }

    return ret;
  }
  #endregion

  #region GetMaxValue Method
  public static T GetMaxValue<T>(this XElement elem, string
elemName, string parentNodeName)
  {
    // Get the largest node value
    T maxValue =
      (from node in elem.Elements(parentNodeName)
       select node.GetAs<T>(elemName, default)).Max();

    return maxValue;
  }
  #endregion

  #region GetCount Method
  public static int GetCount(this XElement elem, string
parentNodeName)
  {
    // Get total rows in XML file
    int count = (from node in elem.Elements(parentNodeName)
                 select node).Count();

    return count;
  }
  #endregion
}
```

Get All Songs

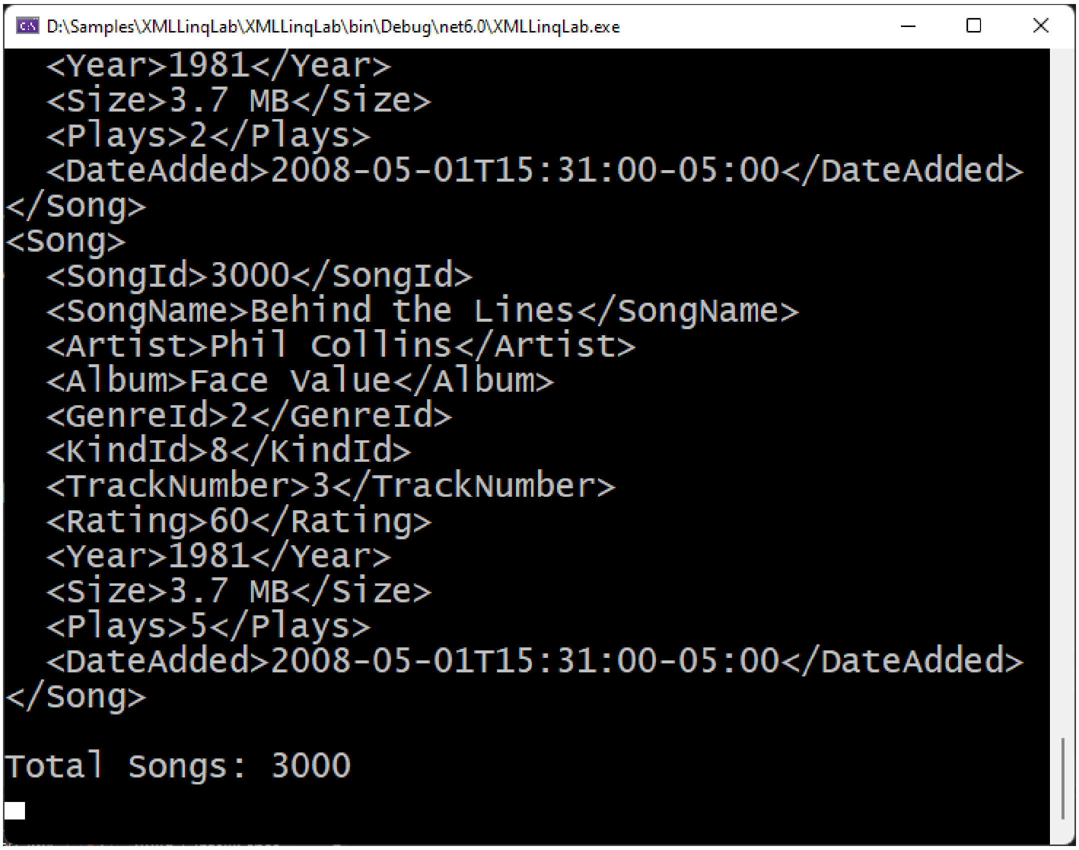
Open the **Program.cs** file and replace all the code in there with the following code.

```
#nullable disable

using System.Xml.Linq;
using Samples;

XElement elem = XElement.Load(FileNameHelper.SongsFile);
List<XElement> list;

// TODO: Write Query Here


// Display Songs
foreach (XElement row in list) {
  Console.WriteLine(row);
}

// Display Total Count
Console.WriteLine();
Console.WriteLine($"Total Songs: {list.Count}");

// Pause for Results
Console.ReadKey();
```

Write the code to retrieve all Song nodes from the XElement document

Try it Out

Run the application and your console window should look like the following:



D:\Samples\XMLLinqLab\XMLLinqLab\bin\Debug\net6.0\XMLLinqLab.exe

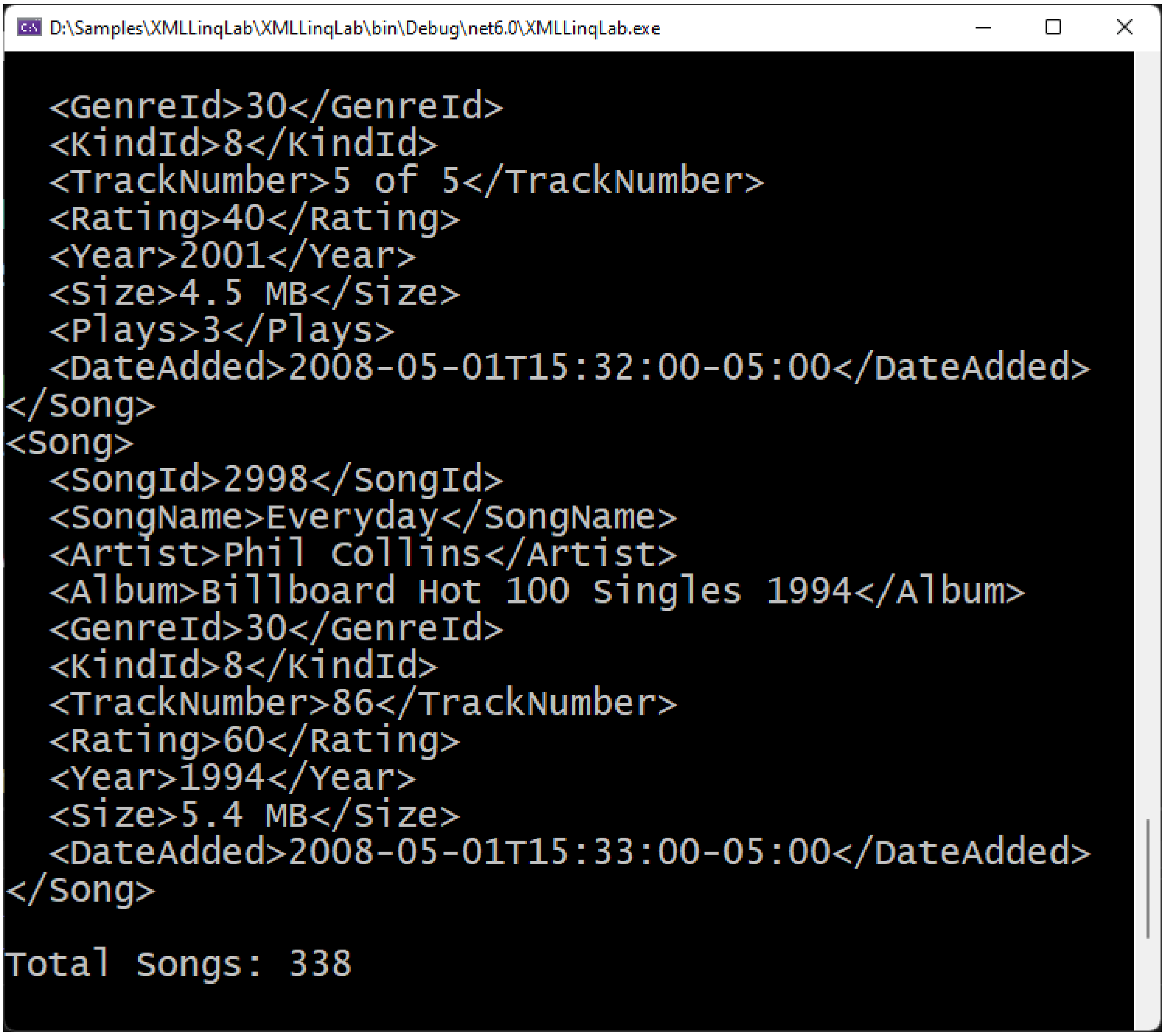
```
  <Year>1981</Year>
  <Size>3.7 MB</Size>
  <Plays>2</Plays>
  <DateAdded>2008-05-01T15:31:00-05:00</DateAdded>
</Song>
<Song>
  <SongId>3000</SongId>
  <SongName>Behind the Lines</SongName>
  <Artist>Phil Collins</Artist>
  <Album>Face Value</Album>
  <GenreId>2</GenreId>
  <KindId>8</KindId>
  <TrackNumber>3</TrackNumber>
  <Rating>60</Rating>
  <Year>1981</Year>
  <Size>3.7 MB</Size>
  <Plays>5</Plays>
  <DateAdded>2008-05-01T15:31:00-05:00</DateAdded>
</Song>

Total Songs: 3000
```

Lab 2: Add a Where Clause

Add a where clause to the last query to filter the songs where the ***GenreId*** property is equal to 30.

Try it Out

Run the application and your console window should look like this.


D:\Samples\XMLLinqLab\XMLLinqLab\bin\Debug\net6.0\XMLLinqLab.exe

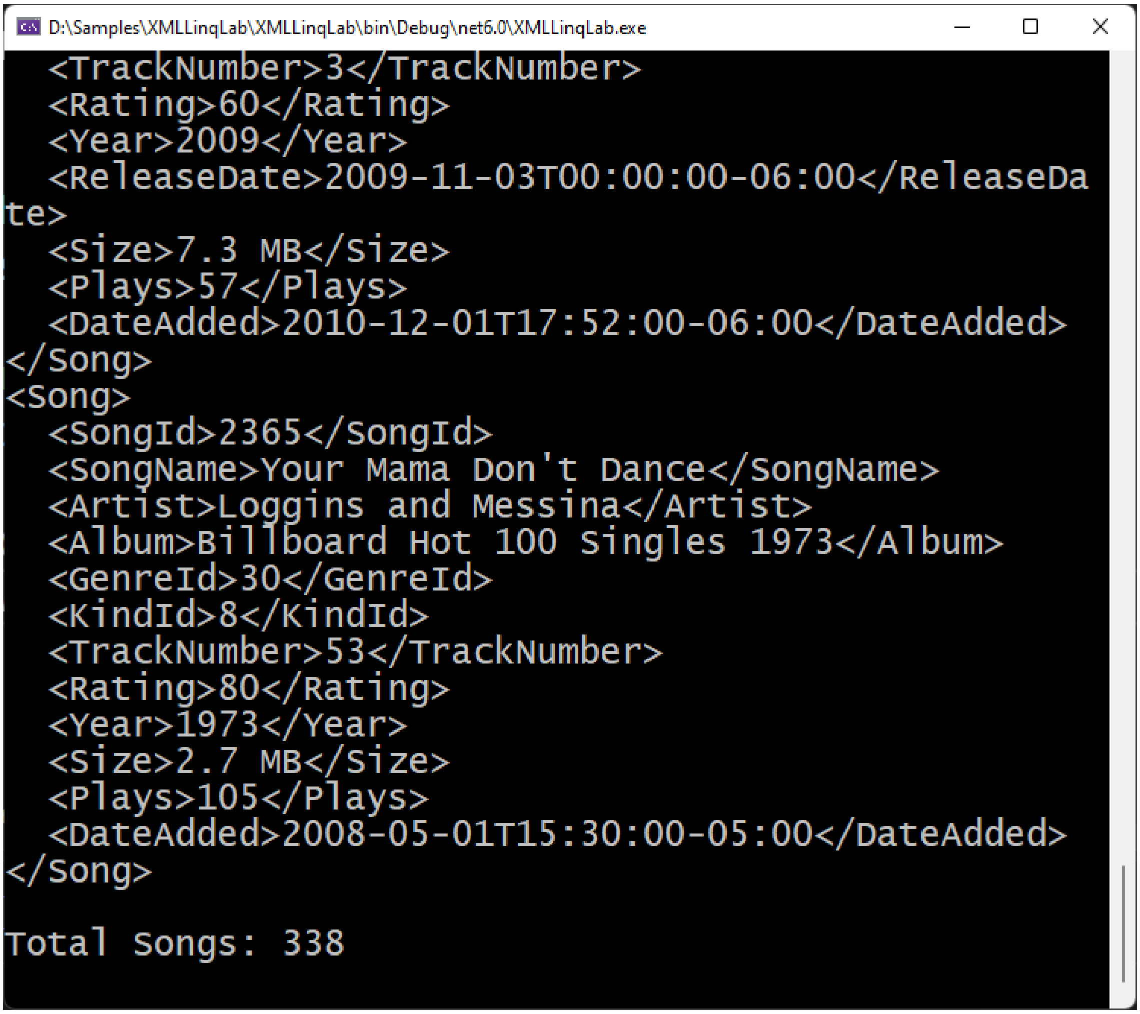
```
  <GenreId>30</GenreId>
  <KindId>8</KindId>
  <TrackNumber>5 of 5</TrackNumber>
  <Rating>40</Rating>
  <Year>2001</Year>
  <Size>4.5 MB</Size>
  <Plays>3</Plays>
  <DateAdded>2008-05-01T15:32:00-05:00</DateAdded>
</Song>
<Song>
  <SongId>2998</SongId>
  <SongName>Everyday</SongName>
  <Artist>Phil Collins</Artist>
  <Album>Billboard Hot 100 Singles 1994</Album>
  <GenreId>30</GenreId>
  <KindId>8</KindId>
  <TrackNumber>86</TrackNumber>
  <Rating>60</Rating>
  <Year>1994</Year>
  <Size>5.4 MB</Size>
  <DateAdded>2008-05-01T15:33:00-05:00</DateAdded>
</Song>

Total Songs: 338
```

Lab 3: Add an Order By Clause

Add an order by clause to the previous query to sort the data on the ***SongName*** property.

Try it Out

Run the application and your console window should look like this.



D:\Samples\XMLLinqLab\XMLLinqLab\bin\Debug\net6.0\XMLLinqLab.exe

```
  <TrackNumber>3</TrackNumber>
  <Rating>60</Rating>
  <Year>2009</Year>
  <ReleaseDate>2009-11-03T00:00:00-06:00</ReleaseDa
te>
  <Size>7.3 MB</Size>
  <Plays>57</Plays>
  <DateAdded>2010-12-01T17:52:00-06:00</DateAdded>
</Song>
<Song>
  <SongId>2365</SongId>
  <SongName>Your Mama Don't Dance</SongName>
  <Artist>Loggins and Messina</Artist>
  <Album>Billboard Hot 100 Singles 1973</Album>
  <GenreId>30</GenreId>
  <KindId>8</KindId>
  <TrackNumber>53</TrackNumber>
  <Rating>80</Rating>
  <Year>1973</Year>
  <Size>2.7 MB</Size>
  <Plays>105</Plays>
  <DateAdded>2008-05-01T15:30:00-05:00</DateAdded>
</Song>

Total Songs: 338
```

Lab 4: Build Song Collection

Open the **Program.cs** file and modify the ***list*** variable to build a collection of Song objects.

```
List<Song> list;
```

Write a query to select all Song elements

Do NOT use a where clause

Order the song elements by the **SongName** property.

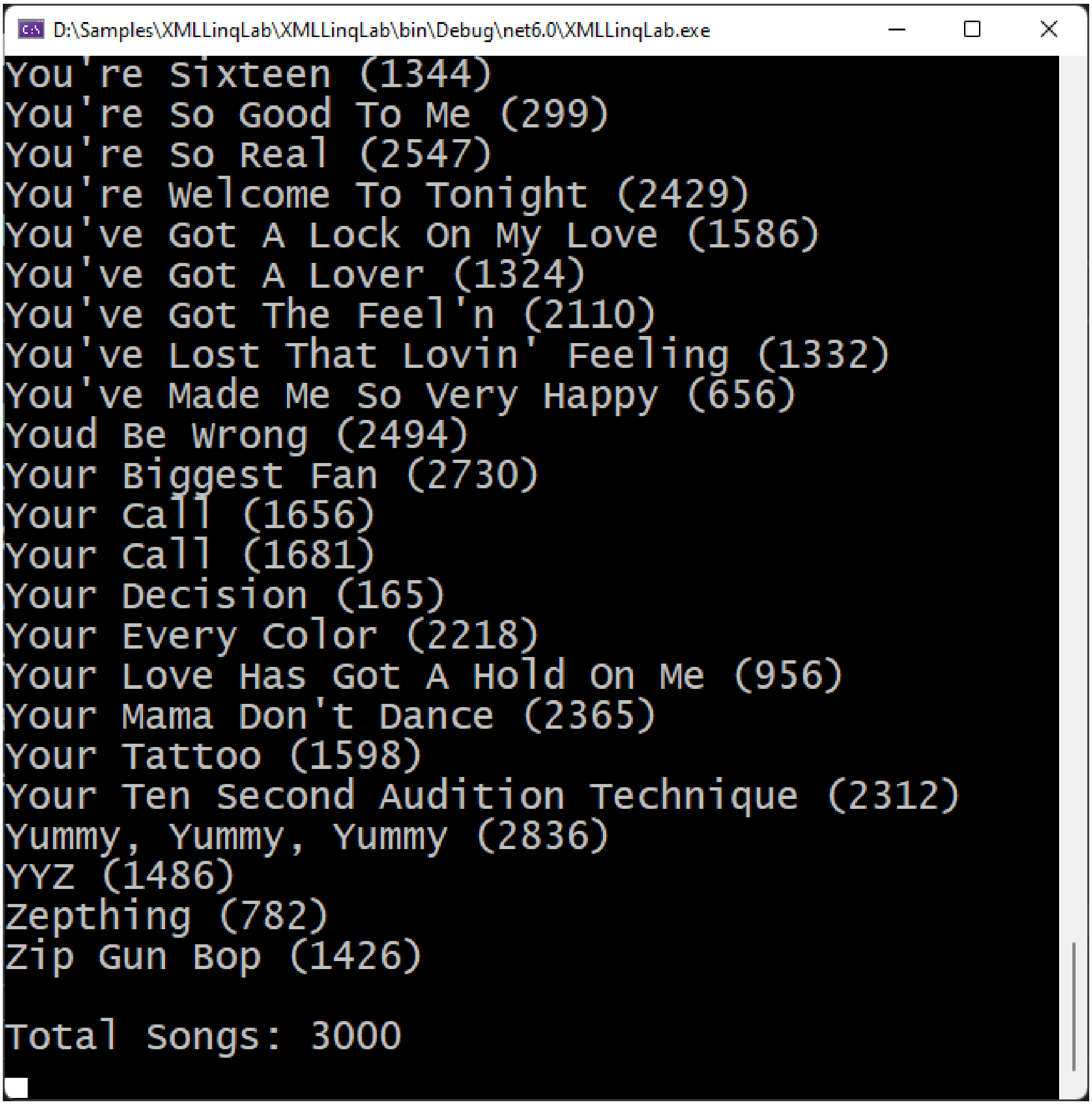
Build a new Song object and put each song element from the XML file into the appropriate properties of the Song object

Rewrite the loop to use a Song object

```
// Display Songs
foreach (Song row in list) {
  Console.WriteLine(row);
}
```

Try it Out

Run the application and your console window should look like this.



D:\Samples\XMLLinqLab\XMLLinqLab\bin\Debug\net6.0\XMLLinqLab.exe

```
You're Sixteen (1344)
You're So Good To Me (299)
You're So Real (2547)
You're Welcome To Tonight (2429)
You've Got A Lock On My Love (1586)
You've Got A Lover (1324)
You've Got The Feel'n (2110)
You've Lost That Lovin' Feeling (1332)
You've Made Me So Very Happy (656)
Youd Be Wrong (2494)
Your Biggest Fan (2730)
Your Call (1656)
Your Call (1681)
Your Decision (165)
Your Every Color (2218)
Your Love Has Got A Hold On Me (956)
Your Mama Don't Dance (2365)
Your Tattoo (1598)
Your Ten Second Audition Technique (2312)
Yummy, Yummy, Yummy (2836)
YYZ (1486)
Zepthing (782)
Zip Gun Bop (1426)

Total Songs: 3000
```