Contains Lab

Lab 1: All() Method

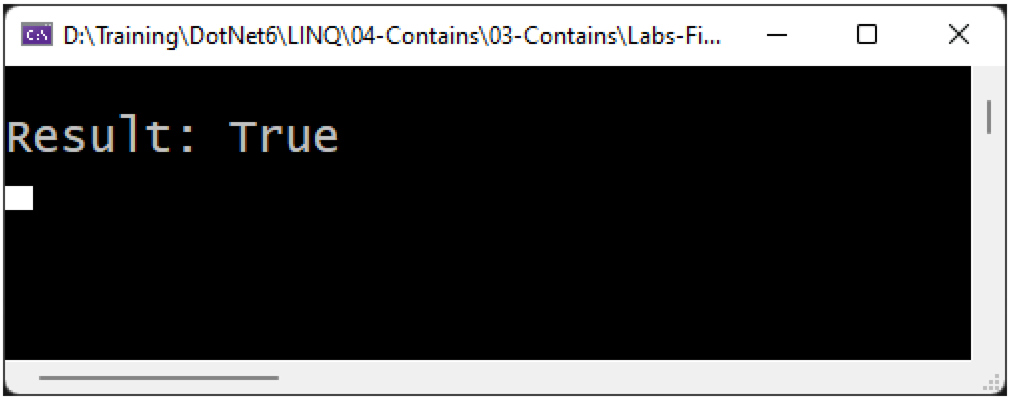
Open the **Program.cs** file and make it look like the following:

```
using LINQLab.EntityClasses;
using LINQLab.RepositoryClasses;

// Declare variables and fill data
List<Song> songs = SongRepository.GetAll();
bool value = false;

// TODO: Write Your Query Here

// Display Result
Console.WriteLine();
Console.WriteLine($"Result: {value}");

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

Write a query to check if All songs have a ***Rating*** that is greater than zero.

Try it Out

Run the application and the results should look like the following:



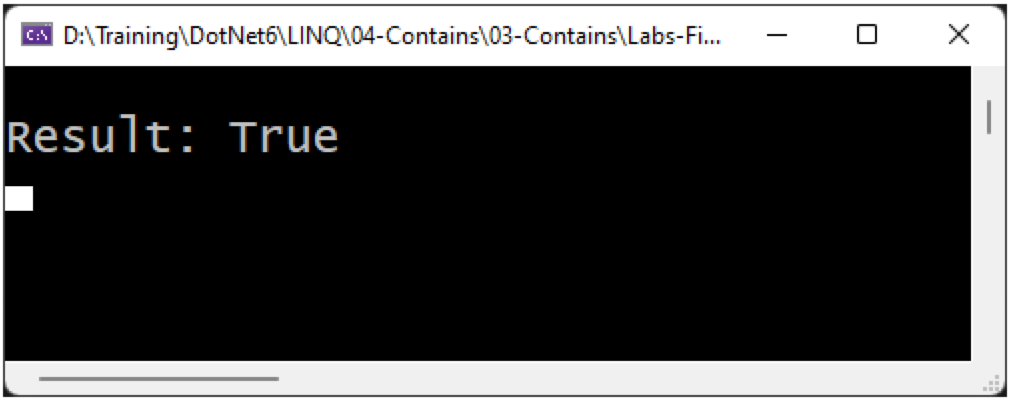
```
Result: True
```

Lab 2: Any() Method

Write a query to check if Any songs have a ***Rating*** that is greater than 90.

Try it Out

Run the application and the results should look like the following:



```
Result: True
```

Lab 3: Contains() Method using Comparer

Create a new class named **SongComparer** that looks like the following:

```
#nullable disable

namespace LINQLab.EntityClasses;

public class SongComparer : EqualityComparer<Song> {
  public override bool Equals(Song x, Song y) {
    return (x.SongId == y.SongId);
  }

  public override int GetHashCode(Song obj) {
    return obj.SongId.GetHashCode();
  }
}
```

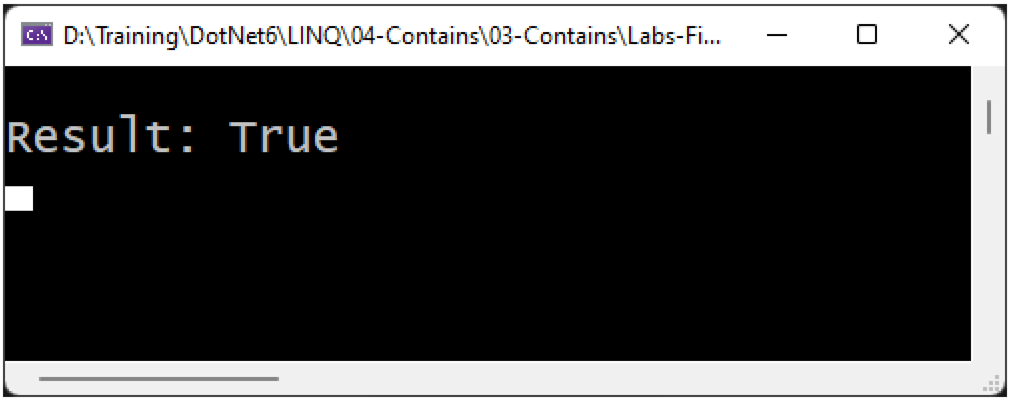
Open the **Program.cs** file and declare a new instance of the SongComparer class you just created.

```
SongComparer sc = new();
```

Write a query to check if the ***SongId*** of 115 exists in the ***songs*** collection.

Try it Out

Run the application and the results should look like the following:



```
Result: True
```