**Instructions**

* The duration of this challenge is **120 Minutes**.
* Programming questions have a **Compile and Run** option where you can run your solution against sample test cases before submitting it.
* Click **Evaluate**button only if your code compiles successfully.
* This challenge covers the following topic(s).
  + Generic Collections
  + ADO.Net

**Problem: Movie Screening Application**

Vivera Multi Complex which has multiple movie screens wants to create any application through which customers can know the movie details such as title, duration and viewers rating of the movies that are running in the multi complex. You are given the task to create the application. Help Vivera Multi Complex by creating the application.

Functionalities:

* Add movies information to the database
* Display movies information from the database with highest viewers rating.
* Display movies information from the database searched through first letter of movie name.

Data Design

|  |  |  |
| --- | --- | --- |
| **Table name: tblMovie** | |  |
| **Column Name** | **Data type** | **Constraints** |
| Id | int | PK, Identity, Not Null |
| Name | varchar(100) | Unique |
| Duration | float | Not Null |
| Rating | int | Not Null |

The database connection information is specified in the “App.config” file, which is also provided as part of code skeleton. **THIS IS GIVEN ONLY FOR YOUR REFERENCE**. You need **NOT** change this. The code skeleton will be available in the Tekstac platform.

Component Specification

* Create a model class called **Movie** with the below public auto-implemented properties:

|  |  |  |
| --- | --- | --- |
| **Type (Class)** | **Data Types** | **Properties** |
| Movie | int | Id |
| string | Title |
| float | Duration |
| int | Rating |

* Create a Class called **MovieScreening** with the below methods:

|  |  |  |
| --- | --- | --- |
| **Type(Class)** | **Methods** | **Responsibilities** |
| MovieScreening | public bool AddMovie(Movie movie) | This method should accept a **Movie** object and execute a sql queryto insert a movie record into the database. It returns true if the insertion is successful and false if it fails to insert. |
| public IList<Movie> DisplayMoviesByHighestRating () | This method will retrieve all movies records from the database whose viewer rating is highest. Store each record in Movie object. Add these objects to a ‘List’. ***[Hint: More than one movie can have the highest rating; so use a sub query for this method].*** |
| public IList<Movie> SearchMoviesByStartLetter(string searchedFirstLetter) | This method will retrieve all movies records from the database based on searched first letter of movie name. Store each record in Movie object. Add the objects to a ‘List’. |

* Create a class called **Program** with main method. Call the MovieScreening methods under the Program classand test your application.

Business Rules

* In the **Program** class, if the **AddMovies** method returns true then display the message **“Movie Details Successfully Added to the Database”**. Otherwise display the message **“Failed to Add Movie Details”.**
* In the **Program** class check whether the List returned by **DisplayMoviesByHigestRating** method contains data or not. If no data is present in list display the messag**e “No Records Found”.** If the list contains data display the records with 20 spaces between each data as shown in the *section 7*
* In the **Program** class check whether the List returned by **SearchMoviesByStartLetter** method contains data. If no data is present in list display the messag**e “No Movie(s) Found With The First Letter '<searched\_letter>' ”.** If the list contains data display the records with 20 spaces between each data as shown in the *section 7*

**[Section 7]**

***[Hint: For 20 spaces between data use Console.WriteLine("{0,-20}{1,-20}{2,-30}{3}”,”value1”,”value2”,”value3”,”value4”) format]***

Sample Input / Output:

