**Instructions**

* You need to solve **one programming problem** in this challenge.
* 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).
  + Entity Framework Code First Approach
  + Data Annotation Attributes
  + CRUD Operations

**Scenario:**

Arsenal Company’s HRM wants to have an offline application where the Employees can apply for leaves. You are given the task to create the Employee Leave Tracker Application. Help the HRM department by developing the application.

Using **C#** and **Entity Framework Code First Approach**, create the HRM Leave Tracker application.

**Functionalities:**

* An employee can apply leave
* An employee can see his/her leave details by entering the employee id
* An admin can see leave application made by all employees

1. Create a Model class called **LeaveModel** with the below public properties and rules **[The constraints will be applied using Data Annotation Attributes on Model Class]:**

|  |  |  |
| --- | --- | --- |
| **Property Name** | **Datatype** | **Constraint** |
| LeaveNumber | integer | This is a required field. Make this field Primary Key and auto-generate its value in database. |
| EmployeeId | string | This is a required field |
| EmployeeName | string | This is a required field |
| FromDate | DateTime | This is a required field |
| ToDate | DateTime | This is a required field |
| LeaveType | string | This is a required field |

LeaveModel entity should be mapped to **tblLeave** Table using Data Annotation attribute in Model class.

2. Create a DbContext class called **EmployeeLeaveTracker** to establish connection with SQLSERVER database. Also implement property for **‘Leaves'** with required **DbSet** declaration.

**Note: Use public accessor and virtual keyword while declaring DbSet.**

Use Code first Approach to create a database called **EmployeeLeaveTrackerDb** and store the **tblLeave** table in this database.

**App.Config snipet:**

<connectionStrings>

<add name="DbConnect" connectionString="xxxxxx xxxxxx xxxxx xxxxxx xxxxxx " providerName="System.Data.SqlClient"/>

</connectionStrings>

4. Create a class **EmployeeLeaveRepository** with the following methods:

|  |  |
| --- | --- |
| **Method Name** | **Description** |
| public LeaveModel ApplyForLeave(LeaveModel leave) | This method will be used by an employee to insert data to Leave Table. |
| public List<LeaveModel> DisplayEmployeeLeaveInfo(string employeeId) | This method will be used by an employee to display the leave details searched through his/her employee id. Display Employee Id, Employee Name, Start Date, End Date and Leave Type as shown in sample input output. If the List returns no records display the message ‘No Leave information found for your id’. |
| public List<LeaveModel> DisplayEmployeeLeavesForAdmin() | This method will display leave details of all the employees for the admin. Display Leave Id, Employee Id, Employee Name, Start Date, End Date and Leave Type as shown in sample input output If the List returns no records display the message ‘No employee has applied for leave’. |

5. Class **Program**

Implement the Main method based on the Sample Input/Output given below.





