**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).
  + Entity Framework Code First Approach
  + One-to-Many Relationship
  + CRUD Operations

**Scenario : UPI Wallet**

Countries are going towards a cashless economy. To increase cashless transactions, the governments wanted a startup to create UPI platforms with features.

Using **C# and Data Annotation Attribute** approach, establish a **One-to-many relationship** between an UPI and Transactions.

**Functionalities**

* Add the UPI details to database.
* Add Transactions done by each UPIs to the database.

1. Create a Model Class called **UPIModel** with the below public properties and with rules [The constraints and entity mapping to be implemented under model class using Data Annotation attributes]

|  |  |  |
| --- | --- | --- |
| **Property Name** | **Data Type** | **Rules** |
| UPI | String | This field should not accept null values. This field should be the KEY. |
| Name | String | This field should not allow null values. Should contain only alphabets. |
| Mobile | Long | This field should not allow null values. |
| Email | String | This field should not allow null values. |
| JoinedOn | DateTime | This field should not allow null values. |
| TransactionModel | ICollection<TransactionModel> | Collection navigation property |

The **UPIModel** entity should be mapped to **“tblUPI”** table using Data Annotation attribute.

1. Create a Model Class called **TransactionModel** with the below public properties and with rules [The constraints and entity mapping to be implemented under model class using Data Annotation Attributes]

|  |  |  |
| --- | --- | --- |
| **Property Name** | **Data Type** | **Rules** |
| TransactionId | Int | This field should be an auto incremented value and no null value. |
| Type | String | This field should not allow null values. |
| Amount | Double | This field should not allow null values. Value should be greater than 0. |
| TransactionDate | DateTime | The field should not allow null value. Should be current date and time. |
| UPI | UPIModel | Virtual field. This field should not allow null values. |

The **TransactionModel** entity should be mapped to **“tblTransactions”** table using Data Annotation attribute.

1. Create a class called **UPIWallet** with the following method:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Method Name** | **Argument** | **Return Type** | **Access Specifier** | **Responsibilities** |
| FavouriteUPI |  | string | Public | This method returns a UPI for whom highest amount is debited |
| GetPerDateTransactions |  | Dictionary<DateTime,int> | Public | The method is used for displaying number transactions per date.  **Note :** Use only the date to find transactions per date. |

1. Create a DbContext class called **UPIWalletContext** to establish connection with SQL Server database. Also implement property for ‘UPIs’ and ‘Transactions’ with required DbSet declaration.

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

1. Create a class called **UPIRepository** with the below methods:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Method Name** | **Argument** | **Return Type** | **Access Specifier** | **Responsibilities** |
| AddUPI | UPIModel upi | Int | public | This method will allow to add UPI details. |
| GetUPIs |  | List<UPIModel> | public | This method will return all the UPI details. |
| AddTransactions | TransactionModel transactions | Int | public | This method has to add the transaction details in the database.  **Note :**  Don’t add transaction details if  UPI is not present. Return -2 and display UPI is not matched. |

**Note : Declare all mentioned class as public.**

1. Class **Program**

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











