SQL Assignment SQL - Database

Your assignment is to build a database from scratch. Add tables to the database, insert data into tables and query the database.

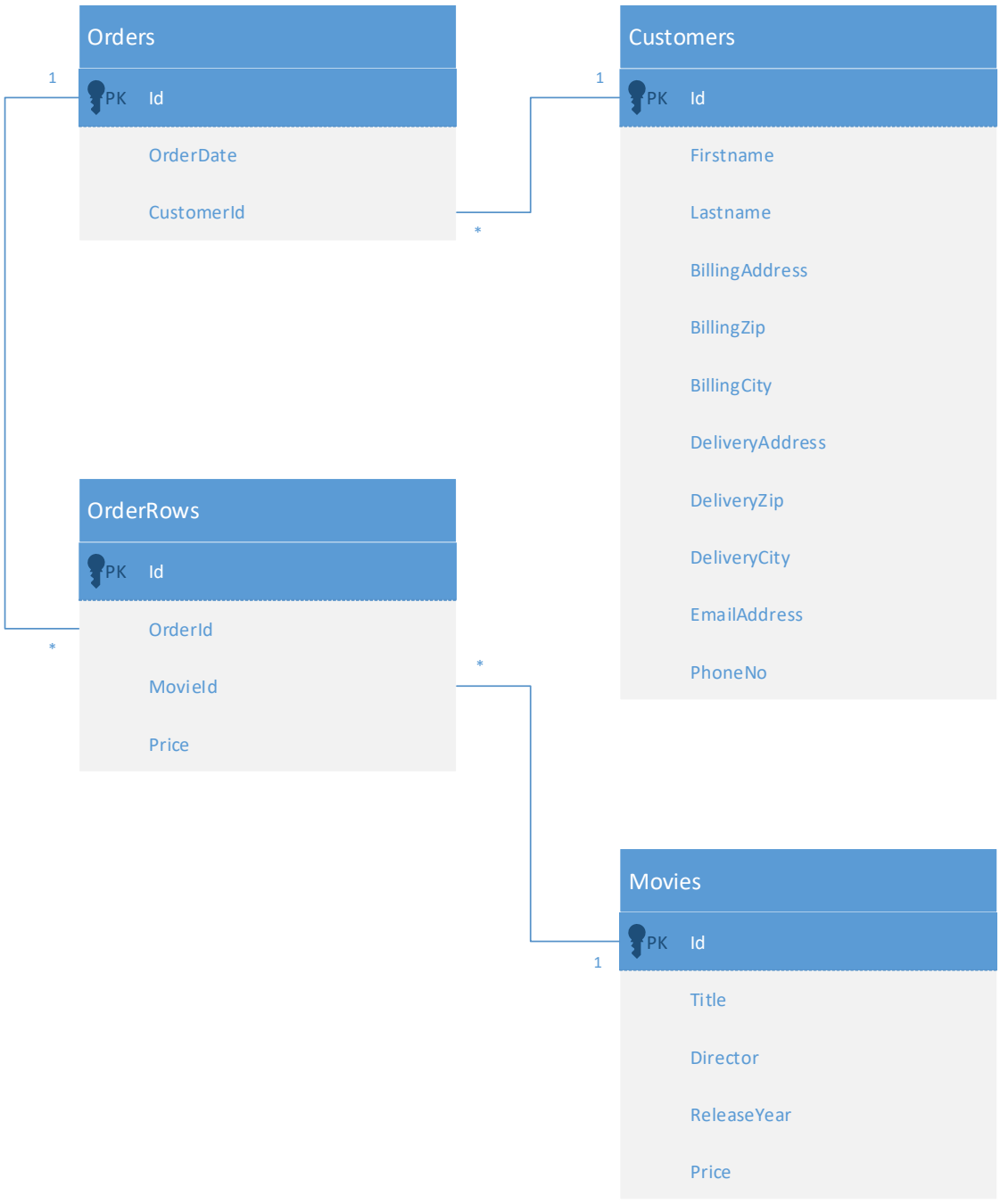
All queries should be contained in one single query-file(.sql). Create a folder named Database on your OneDrive. Submit query file and screenshots of your table design from Visual Studio to that folder.

Check YouTube , TutorialsPoint and Pluralsight for information on how to implement the required features.

Assignment 1

Features

- Create a database for a webshop of DVD movies with following entities and relationship
  - Movie: The moves are the products being sold in the webshop
  - Customer: Makes an order to the webshop and supplies information for billing and delivery of order
  - Order: Order details are saved to the database
  - OrderRows: Maintains the sale details of all orders.
  - Following diagram describes the required columns and their relationship among tables within database


Orders
PK Id
OrderDate
CustomerId
Customers
PK Id
Firstname
Lastname
BillingAddress
BillingZip
BillingCity
DeliveryAddress
DeliveryZip
DeliveryCity
EmailAddress
PhoneNo
OrderRows
PK Id
OrderId
MovieId
Price
Movies
PK Id
Title
Director
ReleaseYear
Price
1
*
1
*
*
1

Explanation

Relation

Each Order is related to one customer and one or more OrderRows, which in turn contains the price of movies being sold in that order .

Why is the Price column included in both the Movie and the OrderRow tables?

Because the Movie - DVD price may change at any time in future. For a record we can trace the older price of the movie.

For example let's say a customer, Joe, buys the movie Se7en on January 12th 2015 for 179 kr. One year later, the price of Se7en is lowered to 79 kr.

If we do not keep the Price column in the OrderRows table, we lose the price of the movie at time it was sold to Joe.

The Price-column in the Movie table represents the current price of the movie. However, the Price-column in the OrderRows table represents the price of the movie at the time it was sold.

Exercise 1 - CREATE DATABASE AND TABLES

- Create all tables, as mentioned in diagram
- Table names and column names should be exactly same as mentioned in diagram, even no misspellings allowed.
- Set primary key and REFERENCES to other tables as mentioned in diagram
- Create the tables in right order, which will help you to reference to correct table.
- Select the appropriate datatype for each column
- Primary key should be automatically increased while adding new record to table
- No column should allow NULL values.

Exercise 2 - INSERT

- Insert following data to Movies and Customers table
- At least one record in each table should be inserted using INSRT query. (You can also enter data directly to table)

Customers

Id	Firstname	Lastname	BillingAddress	BillingZip	BillingCity	DeliveryAddress	DeliveryZip	DeliveryCity	EmailAddress	PhoneNo
1	Jonas	Gray	23 Green Corner Street	56743	Birmingham	23 Green Corner Street	56743	Birmingham	jonas.gray@hotmail.com	0708123456
2	Jane	Harolds	10 West Street	43213	London	10 West Street	43213	London	jane_h77@gmail.com	0701245512
3	Peter	Birro	12 Fox Street	45681	New York	89 Moose Plaza	45321	Seattle	peter_the_great@hotmail.com	0739484322

Movies

Id	Title	Director	ReleaseYear	Price
1	Interstellar	Christoper Nolan	2014	179
2	Hobbit: Battle of the five armies	Peter Jackson	2014	179
3	The Wolf of Wall Street	Martin Scorcese	2013	119
4	Pulp Fiction	Quentin Tarantinc	1994	49

Exercise 3 - INSERT

- Write a queries to create orders and OrderRows for the following scenarios
  a. On 2015-01-01, Jonas Gray purchases Interstellar and Pulp Fiction
  b. On 2015-01-15, Peter Birro purchases 2 copies of The Wolf of Wall Street.
  c. On 2014-12-20, Jonas Gray purchased The Wolf of Wall Street.

(**Tip:** You should create the Order first, otherwise you will not have OrderId , which will be required while creating the OrderRows.)

Query example:

```
INSERT INTO Orders
VALUES(.....)

SELECT TOP 1 Id FROM Orders ORDER BY Id DESC --Get the latest inserted OrderId
INSERT INTO OrderRows
VALUES(....)
```

Exercise 4 – UPDATE

- Write a query that changes the price of all movies made in 2014 to 169 kr.

Exercise 5 – SELECT

- Write queries for the following SELECT operations:
  a. Get Firstname, Lastname, PhoneNo and Email to all Customers.
  b. Get all movies, ordered by Year from newest to oldest.
  c. Get all movie titles, ordered by Price, from cheapest to most expensive.
  d. Get Firstname, Lastname, DeliveryAddress, DeliveryZip, DeliveryCity for all customers who bought The Wolf of Wall Street.
  e. Get Id, Date, Customer (Firstname, Lastname) and total cost of every individual order.
  f. **(Optional)** Get Customer (Firstname, Lastname), total number of movies ordered by this customer, number of orders by this customer and total cost of all orders by this customer.
  g. **(Optional)** Get number of orders and total cost for all orders in the database.

Exercise 6

- Add a new column, CellNo to the Customers table. The column should contain the customer's cellphone number.

  (The old column, PhoneNo currently holds cellphone numbers only.)

- Write a query to copy the information from PhoneNo to CellNo.
- Write a query to empty the PhoneNo column(Sets it to an empty string)

Assignment 2 - Optional

Features

- Design and create a database that emulates the functionality on Facebook. The tables should hold information about the following:
  - Store information about a user
  - Store information about who is friend with whom, and when they became friends
  - Store information about uploaded images
  - Store information about posts (text written by the user)
  - Store information about comments given on a post/image
  - Store information about who is tagged inside an image (who tagged, who got tagged, and tag area)
  - Store details of likes (a user likes a post/image)

Resources:

http://www.sqlcourse.com - SQL training

http://www.tutorialspoint.com/sql/ - TutorialsPoint

Expected Duration: 2 Days.

Subjects Covered:

- SQL - Database
  - Create Table
  - Insert Query
  - Select Query
  - Alter Table
  - Where Clause
  - Order By
  - Group By
  - Constraints
  - using joins