IIT MANDI CS-207 APPLIED DATABASE PRACTICUM FALL 2018

ASSIGNMENT 5: Understanding Cursors and Triggers

(In groups of 4-5 students containing B. Tech. 2nd years)

Dr. Varun Dutt **Due: Before 12:00 NOON on September 24th, 2018**

Readings:

- Class notes and slides from week 6
- Activity 6 on Moodle

Objectives:

- Understand how Cursors and Triggers works

Submission:

Each group will make one single submission.
Please submit as a zipped file, the following content:

1. A doc file containing the code and images of the **result table** (Please use the **limit** clause to only show the first 10 entries of the table)

Assignment:

Q1. In this question load the database from the dump file (join.sql) given in moodle

The Database has the following tables:

Customers: stores customer's data.

Products: stores a list of scale model cars.

ProductLines: stores a list of product line categories.

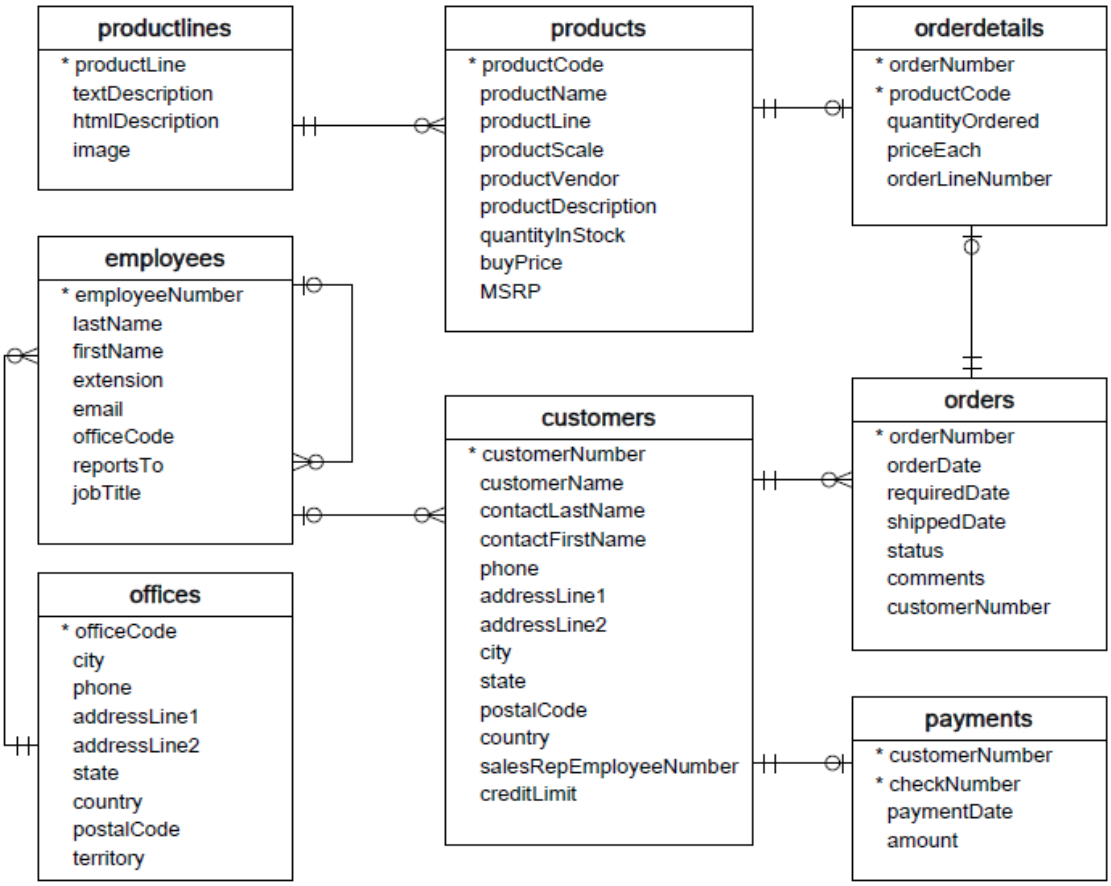
Orders: stores sales orders placed by customers.

OrderDetails: stores sales order line items for each sales order.

Payments: stores payments made by customers based on their accounts.

Employees: stores all employee information as well as the organization structure such as who reports to whom.

Offices: stores sales office data.



Q1) Create a table named **customer_quantity_copy** with the following schema:

```
mysql> describe customer_quantity_copy$$
+--------------+--------------+------+-----+---------+-------+
| Field        | Type         | Null | Key | Default | Extra |
+--------------+--------------+------+-----+---------+-------+
| customerName | varchar(255) | YES  |     | NULL    |       |
| QUANTITY     | int(11)      | YES  |     | NULL    |       |
+--------------+--------------+------+-----+---------+-------+
2 rows in set (0.00 sec)
```

ii) Create a stored procedure that used a cursor that fetches the total orders (**orderdetails** table) placed by each customer (**customers** table) and store it in the **customer_quantity_copy** table.

Q2. Create a table named **employee_audit** that keeps track of any updation made to the **employees** table. The **employee_audit table** should have the following schema:

```
+----------------+-------------+------+-----+---------+----------------+
| Field          | Type        | Null | Key | Default | Extra          |
+----------------+-------------+------+-----+---------+----------------+
| id             | int(11)     | NO   | PRI | NULL    | auto_increment |
| employeeNumber | int(11)     | NO   |     | NULL    |                |
| lastname       | varchar(50) | NO   |     | NULL    |                |
| changedat      | datetime    | YES  |     | NULL    |                |
| action         | varchar(50) | YES  |     | NULL    |                |
+----------------+-------------+------+-----+---------+----------------+
```

Create a BEFORE UPDATE trigger that is invoked before a change is made to the **employees** table. This trigger when invoked should insert into **employee_audit** the employeeNumber, Lastname of the employee and the time at which the record was updated into the **employees** table. Also SET action value to update

e.g if the below given query

UPDATE employees SET lastName = 'Phan' WHERE employeeNumber = 1056;

is executed. Then the employee_audit should contain the following values:

```
mysql> SELECT
    ->     *
    -> FROM
    ->     employees_audit;
+----+----------------+-----------+---------------------+--------+
| id | employeeNumber | lastname  | changedat           | action |
+----+----------------+-----------+---------------------+--------+
|  1 |           1056 | Patterson | 2018-09-17 00:15:27 | update |
+----+----------------+-----------+---------------------+--------+
```

Q3. Load the dump file (dump.sql) from moodle.

The database is named **d1** and contains the following tables:

```
mysql> show tables;
+--------------+
| Tables_in_d1 |
+--------------+
| Customers    |
| Orders       |
+--------------+
2 rows in set (0.00 sec)
```

Create a table named **customer_audit** that keeps track of any row deletion made to the **Customers** table. The **customer_audit** table should have the following schema:

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
customerID	int(11)	NO		NULL	
user_name	varchar(50)	NO		NULL	
ContactName	varchar(100)	NO		NULL	
changedat	datetime	YES		NULL	
action	varchar(50)	YES		NULL	

Create a AFTER DELETE trigger that is invoked after a record is deleted from the **Customers** table. This trigger when invoked should insert into **customer_audit** the CustomerID, Contact_name of the customer and the time at which the record was deleted from the Customers table. The user who deleted the record should also be inserted into the **customers_audit** table. Also SET action value to **delete.**

if the below given query:

DELETE from Customers where CustomerID=1;

is executed. Then the **customer_audit** should contain the following values:

```
mysql> select * from customer_audit;
+----+------------+----------------+-------------+---------------------+--------+
| id | customerID | user_name      | ContactName | changedat           | action |
+----+------------+----------------+-------------+---------------------+--------+
|  1 |          1 | root@localhost | Maria Anders | 2018-09-17 00:47:01 | DELETE |
+----+------------+----------------+-------------+---------------------+--------+
1 row in set (0.00 sec)
```