MSDS 7330
File Organization and Database Management

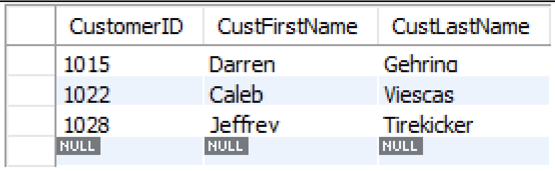
Mini Project 4

Evangelos Giakoumakis

Question 1 : Use the Sales Order Database created during previous week InClass lab and answer following queries using MySQL Workbench. I am also attaching script to create and populate the database in case you don't have it. Submit screen shots of queries along with screen shots of results. If results are longer than one page then simply provide number of rows returned from the query. Answers for the following queries:

1) Display the customers who have never ordered bikes or tires.

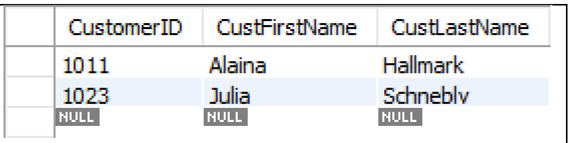
```
        SELECT Customers.CustomerID,
        Customers.CustFirstName,
        Customers.CustLastName
        FROM Customers
        WHERE NOT EXISTS
        (SELECT *
        FROM (Orders
        INNER JOIN Order_Details
        ON Orders.OrderNumber =
        Order_Details.OrderNumber)
        INNER JOIN Products
        ON Products.ProductNumber =
        Order_Details.ProductNumber
        WHERE Products.ProductName LIKE '%Bike'
        AND Orders.CustomerID =
        Customers.CustomerID)
        AND NOT EXISTS
        (SELECT *
        FROM (Orders
        INNER JOIN Order_Details
        ON Orders.OrderNumber =
        Order_Details.OrderNumber)
        INNER JOIN Products
        ON Products.ProductNumber =
        Order_Details.ProductNumber
        WHERE Products.ProductName LIKE '%Tire'
        AND Orders.CustomerID =
Customers.CustomerID)
```



CustomerID	CustFirstName	CustLastName
1015	Darren	Gehring
1022	Caleb	Viescas
1028	Jeffrey	Tirekicker
NULL	NULL	NULL

2) List the customers who have purchased a bike but not a helmet.

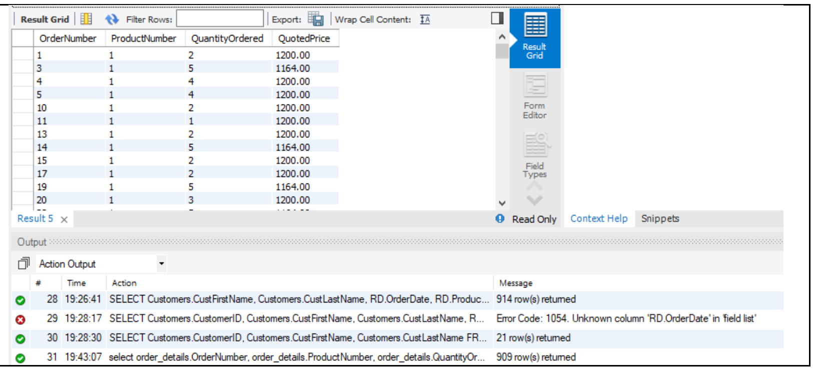
```
        SELECT Customers.CustomerID,
        Customers.CustFirstName,
        Customers.CustLastName
        FROM Customers
        WHERE EXISTS
        (SELECT *
        FROM (Orders
        INNER JOIN Order_Details
        ON Orders.OrderNumber =
        Order_Details.OrderNumber)
        INNER JOIN Products
        ON Products.ProductNumber =
        Order_Details.ProductNumber
        WHERE Products.ProductName LIKE '%Bike'
        AND Orders.CustomerID =
        Customers.CustomerID)
        AND NOT EXISTS
        (SELECT *
        FROM (Orders
        INNER JOIN Order_Details
        ON Orders.OrderNumber =
        Order_Details.OrderNumber)
        INNER JOIN Products
        ON Products.ProductNumber =
        Order_Details.ProductNumber
        WHERE Products.ProductName LIKE '%Helmet'
        AND Orders.CustomerID =
Customers.CustomerID)
```



CustomerID	CustFirstName	CustLastName
1011	Alaina	Hallmark
1023	Julia	Schnebly
NULL	NULL	NULL

3) Show me the customer orders that have a bike but do not have a helmet.

```
        select
        order_details.OrderNumber,
        order_details.ProductNumber,
        order_details.QuantityOrdered,
        order_details.QuotedPrice
        from order_details inner join
        products on
        products.ProductNumber =
        order_details.ProductNumber
where CategoryID=2 and ProductName not
like '%Helmet'
```

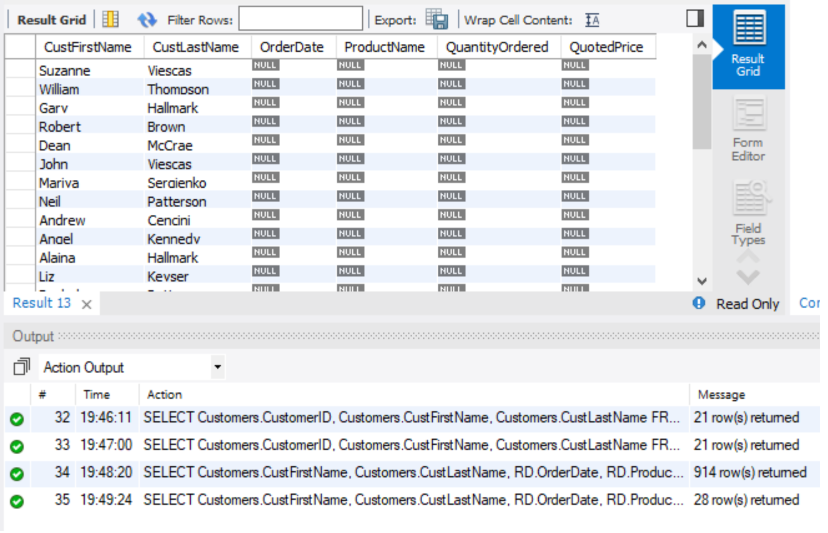


OrderNumber	ProductNumber	QuantityOrdered	QuotedPrice
1	1	2	1200.00
3	1	5	1164.00
4	1	4	1200.00
5	1	4	1200.00
10	1	2	1200.00
11	1	1	1200.00
13	1	2	1200.00
14	1	5	1164.00
15	1	2	1200.00
17	1	2	1200.00
19	1	5	1164.00
20	1	3	1200.00

```
   #   Time      Action                                                                        Message
✓  28  19:26:41  SELECT Customers.CustFirstName, Customers.CustLastName, RD.OrderDate, RD.Produc...   914 row(s) returned
✗  29  19:28:17  SELECT Customers.CustomerID, Customers.CustFirstName, Customers.CustLastName, R...   Error Code: 1054. Unknown column 'RD.OrderDate' in 'field list'
✓  30  19:28:30  SELECT Customers.CustomerID, Customers.CustFirstName, Customers.CustLastName FR...   21 row(s) returned
✓  31  19:43:07  select order_details.OrderNumber, order_details.ProductNumber, order_details.QuantityOr...   909 row(s) returned
```

4) Display the customers and their orders that have a bike and a helmet in the same order.

```
        SELECT
        Customers.CustFirstName,
        Customers.CustLastName,
        RD.OrderDate,
        RD.ProductName,
         RD.QuantityOrdered,
        RD.QuotedPrice
        FROM Customers
        LEFT OUTER JOIN
        (SELECT Orders.CustomerID,
        Orders.OrderDate,
        Products.ProductName,
        Order_Details.QuantityOrdered,
        Order_Details.QuotedPrice
        FROM ((Orders
        INNER JOIN Order_Details
        ON Orders.OrderNumber =
        Order_Details.OrderNumber)
        INNER JOIN Products
        ON
        Order_Details.ProductNumber
        = Products.ProductNumber)
        INNER JOIN Categories
        ON Categories.CategoryID =
        Products.CategoryID
        WHERE
        Categories.CategoryDescription
        =
         'Bikes' and
        Categories.CategoryDescription
        =
         'Helmet')
        AS RD
ON Customers.CustomerID =
RD.CustomerID
```

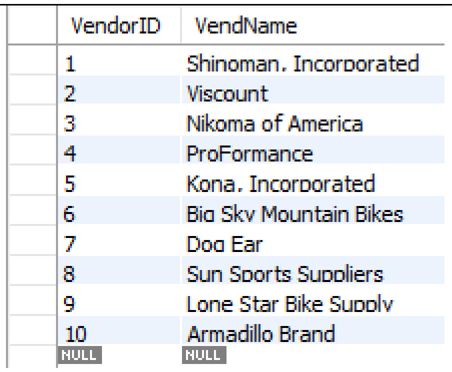


CustFirstName	CustLastName	OrderDate	ProductName	QuantityOrdered	QuotedPrice
Suzanne	Viescas	NULL	NULL	NULL	NULL
William	Thompson	NULL	NULL	NULL	NULL
Gary	Hallmark	NULL	NULL	NULL	NULL
Robert	Brown	NULL	NULL	NULL	NULL
Dean	McCrae	NULL	NULL	NULL	NULL
John	Viescas	NULL	NULL	NULL	NULL
Mariya	Sergienko	NULL	NULL	NULL	NULL
Neil	Patterson	NULL	NULL	NULL	NULL
Andrew	Cencini	NULL	NULL	NULL	NULL
Angel	Kennedy	NULL	NULL	NULL	NULL
Alaina	Hallmark	NULL	NULL	NULL	NULL
Liz	Keyser	NULL	NULL	NULL	NULL

```
   #   Time      Action                                                                        Message
✓  32  19:46:11  SELECT Customers.CustomerID, Customers.CustFirstName, Customers.CustLastName FR...   21 row(s) returned
✓  33  19:47:00  SELECT Customers.CustomerID, Customers.CustFirstName, Customers.CustLastName FR...   21 row(s) returned
✓  34  19:48:20  SELECT Customers.CustFirstName, Customers.CustLastName, RD.OrderDate, RD.Produc...   914 row(s) returned
✓  35  19:49:24  SELECT Customers.CustFirstName, Customers.CustLastName, RD.OrderDate, RD.Produc...   28 row(s) returned
```

5) Show the vendors who sell accessories, car racks, and clothing.

```
                SELECT vendors.VendorID, vendors.VendName
                FROM vendors
                WHERE EXISTS
                (SELECT *
                FROM (product_vendors
                INNER JOIN products
                ON product_vendors.ProductNumber =
                products.ProductNumber)
                INNER JOIN categories
                ON products.CategoryID =
                categories.CategoryID
                WHERE
                categories.CategoryDescription='Accessories' or
                categories.CategoryDescription='Clothing' or
                categories.CategoryDescription='Car racks'
                AND product_vendors.VendorID =
vendors.VendorID)
```



VendorID	VendName
1	Shinoman, Incorporated
2	Viscount
3	Nikoma of America
4	ProFormance
5	Kona, Incorporated
6	Big Sky Mountain Bikes
7	Dog Ear
8	Sun Sports Suppliers
9	Lone Star Bike Supply
10	Armadillo Brand
NULL	NULL

Question 2 : Python – Write a Python Script that will connect to the Sales Order database and execute queries from question 1. The python script will connect to the MySQL database using MySQL connector and then you will execute the query using cursor. To make it easier simply define the query in the beginning of the program. Submit complete python script.

1) Query 1

```
import mysql.connector

cnx = mysql.connector.connect(user='root', password='cloudichigo',
                              host='127.0.0.1',
                              database='salesordersexampletest')

cursor = cnx.cursor()

query1 = (“’ SELECT Customers.CustomerID, Customers.CustFirstName, Customers.CustLastName FROM
Customers WHERE NOT EXISTS (SELECT * FROM (Orders INNER JOIN Order_Details ON
Orders.OrderNumber = Order_Details.OrderNumber) INNER JOIN Products ON Products.ProductNumber
= Order_Details.ProductNumber WHERE Products.ProductName LIKE '%Bike' AND Orders.CustomerID =
Customers.CustomerID) AND NOT EXISTS (SELECT * FROM (Orders INNER JOIN Order_Details ON
Orders.OrderNumber = Order_Details.OrderNumber) INNER JOIN Products ON Products.ProductNumber
= Order_Details.ProductNumber WHERE Products.ProductName LIKE '%Tire' AND Orders.CustomerID =
Customers.CustomerID) “’)

cursor.execute(query1)

for (CustomerID, CustFirstName, CustLastName) in cursor:
    print(CustomerID, CustFirstName, CustLastName)

cursor.close()
cnx.close()
```

2) Query 2

```
import mysql.connector

cnx = mysql.connector.connect(user='root', password='cloudichigo',
                      host='127.0.0.1',
                      database='salesordersexampletest')

cursor = cnx.cursor()

query2 = (‘’’ SELECT Customers.CustomerID, Customers.CustFirstName, Customers.CustLastName FROM
Customers WHERE EXISTS (SELECT * FROM (Orders INNER JOIN Order_Details ON Orders.OrderNumber =
Order_Details.OrderNumber) INNER JOIN Products ON Products.ProductNumber =
Order_Details.ProductNumber WHERE Products.ProductName LIKE '%Bike' AND Orders.CustomerID =
Customers.CustomerID) AND NOT EXISTS (SELECT * FROM (Orders INNER JOIN Order_Details ON
Orders.OrderNumber = Order_Details.OrderNumber) INNER JOIN Products ON Products.ProductNumber
= Order_Details.ProductNumber WHERE Products.ProductName LIKE '%Helmet' AND Orders.CustomerID
= Customers.CustomerID) ‘’’)

cursor.execute(query2)

for (CustomerID, CustFirstName, CustLastName) in cursor:
   print(CustomerID, CustFirstName, CustLastName)

cursor.close()
cnx.close()
```

3) Query 3

```
import mysql.connector

cnx = mysql.connector.connect(user='root', password='cloudichigo',
                              host='127.0.0.1',
                              database='salesordersexampletest')

cursor = cnx.cursor()

query3 = (“’ select order_details.OrderNumber, order_details.ProductNumber,
order_details.QuantityOrdered, order_details.QuotedPrice from order_details inner join products on
products.ProductNumber = order_details.ProductNumber where CategoryID=2 and ProductName not
like "%Helmet" “’)

cursor.execute(query3)

for (OrderNumber, ProductNumber, QuantityOrdered, QuotedPrice) in cursor:
    print(OrderNumber, ProductNumber, QuantityOrdered, QuotedPrice)

cursor.close()
cnx.close()
```

4) Query 4

```
import mysql.connector

cnx = mysql.connector.connect(user='root', password='cloudichigo',
                              host='127.0.0.1',
                              database='salesordersexampletest')

cursor = cnx.cursor()

query4 = (‘’’ SELECT Customers.CustFirstName, Customers.CustLastName, RD.OrderDate,
RD.ProductName, RD.QuantityOrdered, RD.QuotedPrice FROM Customers LEFT OUTER JOIN (SELECT
Orders.CustomerID, Orders.OrderDate, Products.ProductName, Order_Details.QuantityOrdered,
Order_Details.QuotedPrice FROM ((Orders INNER JOIN Order_Details ON Orders.OrderNumber =
Order_Details.OrderNumber) INNER JOIN Products ON Order_Details.ProductNumber =
Products.ProductNumber) INNER JOIN Categories ON Categories.CategoryID = Products.CategoryID
WHERE Categories.CategoryDescription = 'Bikes' and Categories.CategoryDescription = 'Helmet') AS RD
ON Customers.CustomerID = RD.CustomerID ‘’’)

cursor.execute(query4)

for (CustFirstName, CustLastName, OrderDate, ProductName, QuantityOrdered, QuotedPrice) in cursor:
    print(CustFirstName, CustLastName, OrderDate, ProductName, QuantityOrdered, QuotedPrice))

cursor.close()
cnx.close()
```

5) Query 5

```
import mysql.connector

cnx = mysql.connector.connect(user='root', password='cloudichigo',
                              host='127.0.0.1',
                              database='salesordersexampletest')

cursor = cnx.cursor()

query5 = (‘’’ SELECT vendors.VendorID, vendors.VendName FROM vendors WHERE EXISTS (SELECT * FROM
(product_vendors INNER JOIN products ON product_vendors.ProductNumber = products.ProductNumber)
INNER JOIN categories ON products.CategoryID = categories.CategoryID WHERE
categories.CategoryDescription='Accessories' or categories.CategoryDescription='Clothing' or
categories.CategoryDescription='Car racks' AND product_vendors.VendorID = vendors.VendorID) ‘’’)

cursor.execute(query5)

for (VendorID, VendName) in cursor:
    print(VendorID, VendName))

cursor.close()
cnx.close()
```