28/10/2020

DBMS Lab Assignment -2

- Abdullah Jamal - 2018UCP1712

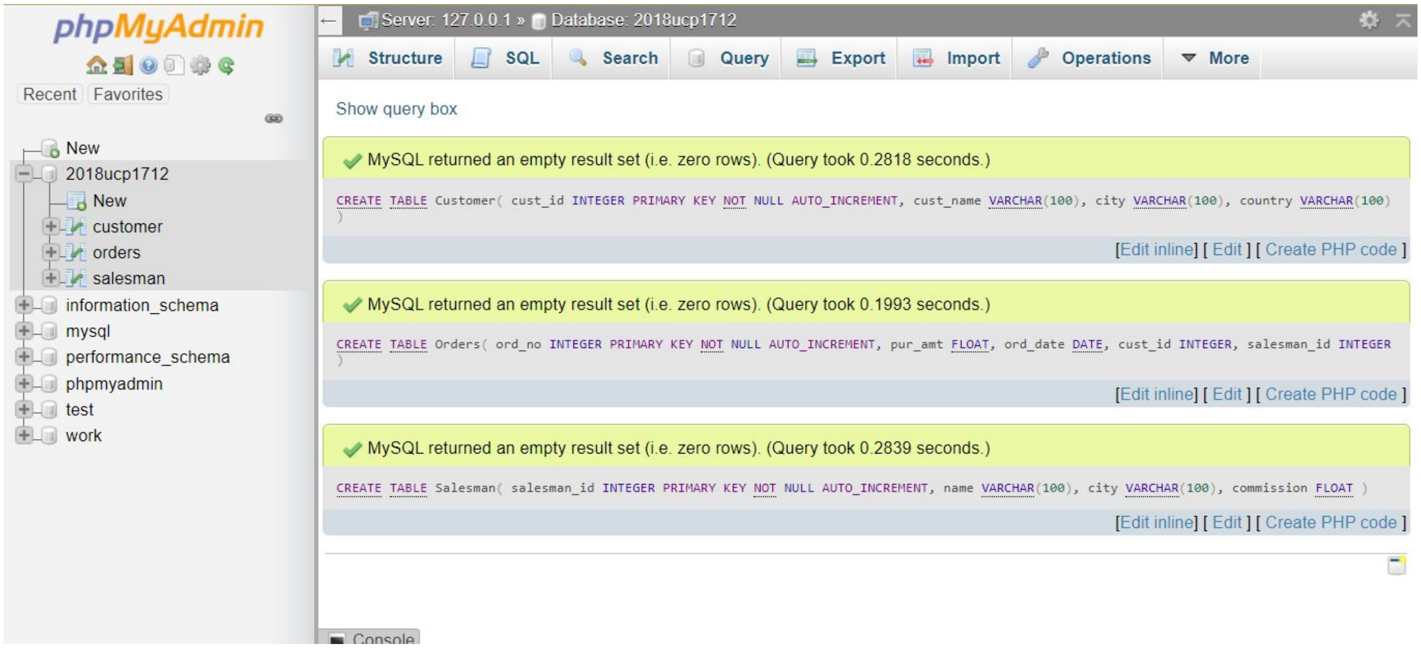
1. Create a database with your ID (example - 2018ucp2123). Then, create the following tables in that database.
• Customer (cust_id, cust_name, city, country)
• Orders (ord_no, pur_amt, ord_date, cust_id, salesman_id)
• Salesman (salesman_id, name, city, commission)

MySQL Query

```
CREATE TABLE Customer(
   cust_id INTEGER PRIMARY KEY NOT NULL AUTO_INCREMENT,
   cust_name VARCHAR(100),
   city VARCHAR(100),
   country VARCHAR(100)
   );

CREATE TABLE Orders(
   ord_no INTEGER PRIMARY KEY NOT NULL AUTO_INCREMENT,
   pur_amt FLOAT,
   ord_date DATE,
   cust_id INTEGER,
   salesman_id INTEGER
   );

CREATE TABLE Salesman(
   salesman_id INTEGER PRIMARY KEY NOT NULL AUTO_INCREMENT,
   name VARCHAR(100),
   city VARCHAR(100),
   commission FLOAT
   );
```



MySQL returned an empty result set (i.e. zero rows). (Query took 0.2818 seconds.)

```
CREATE TABLE Customer( cust_id INTEGER PRIMARY KEY NOT NULL AUTO_INCREMENT, cust_name VARCHAR(100), city VARCHAR(100), country VARCHAR(100)
)
```

MySQL returned an empty result set (i.e. zero rows). (Query took 0.1993 seconds.)

```
CREATE TABLE Orders( ord_no INTEGER PRIMARY KEY NOT NULL AUTO_INCREMENT, pur_amt FLOAT, ord_date DATE, cust_id INTEGER, salesman_id INTEGER
)
```

MySQL returned an empty result set (i.e. zero rows). (Query took 0.2839 seconds.)

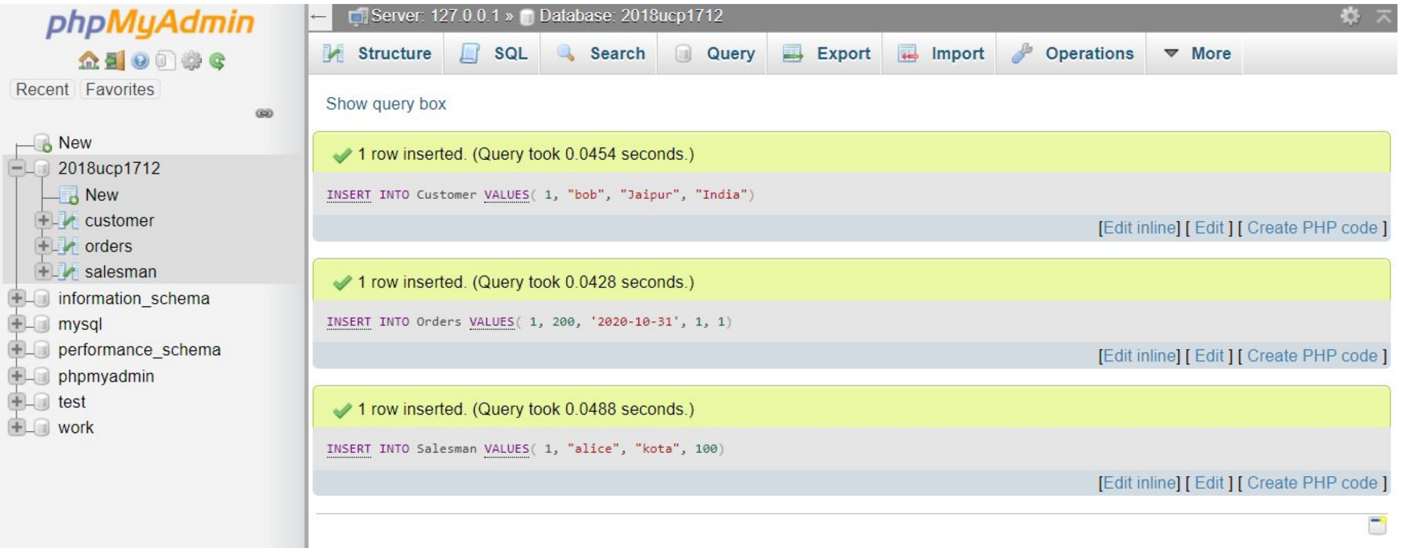
```
CREATE TABLE Salesman( salesman_id INTEGER PRIMARY KEY NOT NULL AUTO_INCREMENT, name VARCHAR(100), city VARCHAR(100), commission FLOAT )
```

2. WAQ (separate for each table) to insert single row of data in the table.

INSERT INTO Customer VALUES(1, "bob", "Jaipur", "India");

INSERT INTO Orders VALUES(1, 200, '2020-10-31', 1, 1);

INSERT INTO Salesman VALUES(1, "alice", "kota", 100);



1 row inserted. (Query took 0.0454 seconds.)

```
INSERT INTO Customer VALUES( 1, "bob", "Jaipur", "India")
```

1 row inserted. (Query took 0.0428 seconds.)

```
INSERT INTO Orders VALUES( 1, 200, '2020-10-31', 1, 1)
```

1 row inserted. (Query took 0.0488 seconds.)

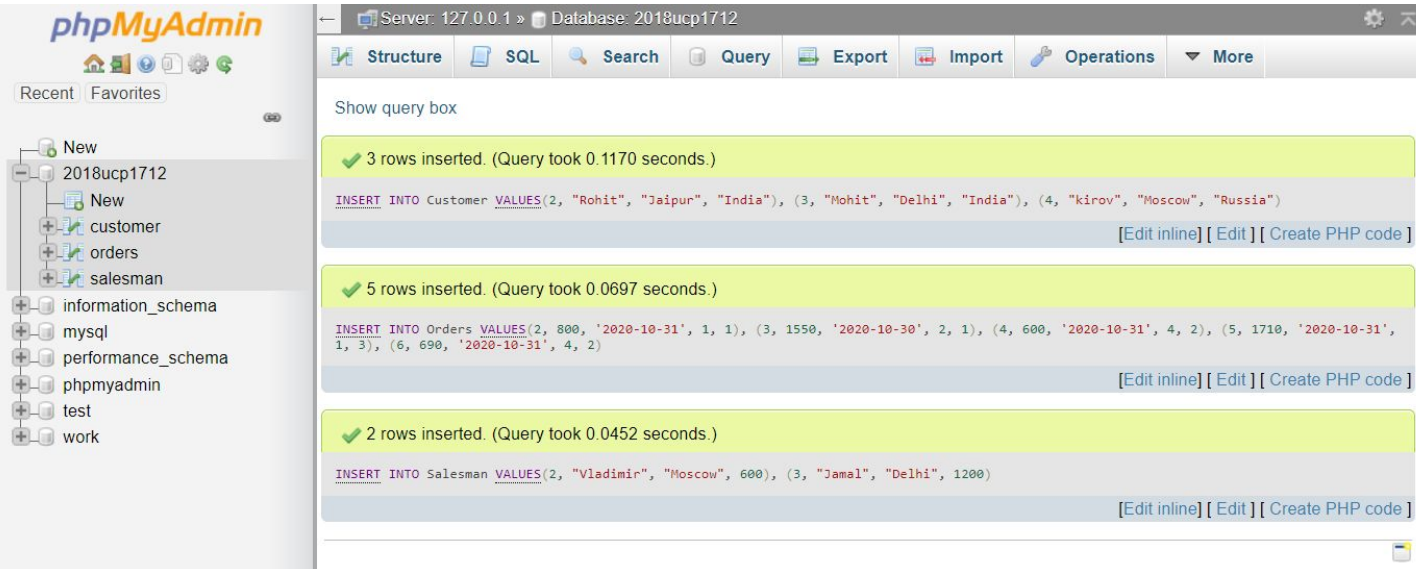
```
INSERT INTO Salesman VALUES( 1, "alice", "kota", 100)
```

3. WAQ (separate for each table) to insert multiple row of data in the table.

```
INSERT INTO Customer VALUES(2, "Rohit", "Jaipur", "India"),
                                        (3, "Mohit", "Delhi", "India"),
                                       (4, "kirov", "Moscow", "Russia");

INSERT INTO Orders VALUES(2, 800, '2020-10-31', 1, 1),
                                (3, 1550, '2020-10-30', 2, 1),
                                 (4, 600, '2020-10-31', 4, 2),
                                (5, 1710, '2020-10-31', 1, 3),
                                (6, 690, '2020-10-31', 4, 2);

INSERT INTO Salesman VALUES(2, "Vladimir", "Moscow", 600),
                                        (3, "Jamal", "Delhi", 1200);
```



✓ 3 rows inserted. (Query took 0.1170 seconds.)

```
INSERT INTO Customer VALUES(2, "Rohit", "Jaipur", "India"), (3, "Mohit", "Delhi", "India"), (4, "kirov", "Moscow", "Russia")
```

[Edit inline] [Edit] [Create PHP code]

✓ 5 rows inserted. (Query took 0.0697 seconds.)

```
INSERT INTO Orders VALUES(2, 800, '2020-10-31', 1, 1), (3, 1550, '2020-10-30', 2, 1), (4, 600, '2020-10-31', 4, 2), (5, 1710, '2020-10-31',
1, 3), (6, 690, '2020-10-31', 4, 2)
```

[Edit inline] [Edit] [Create PHP code]

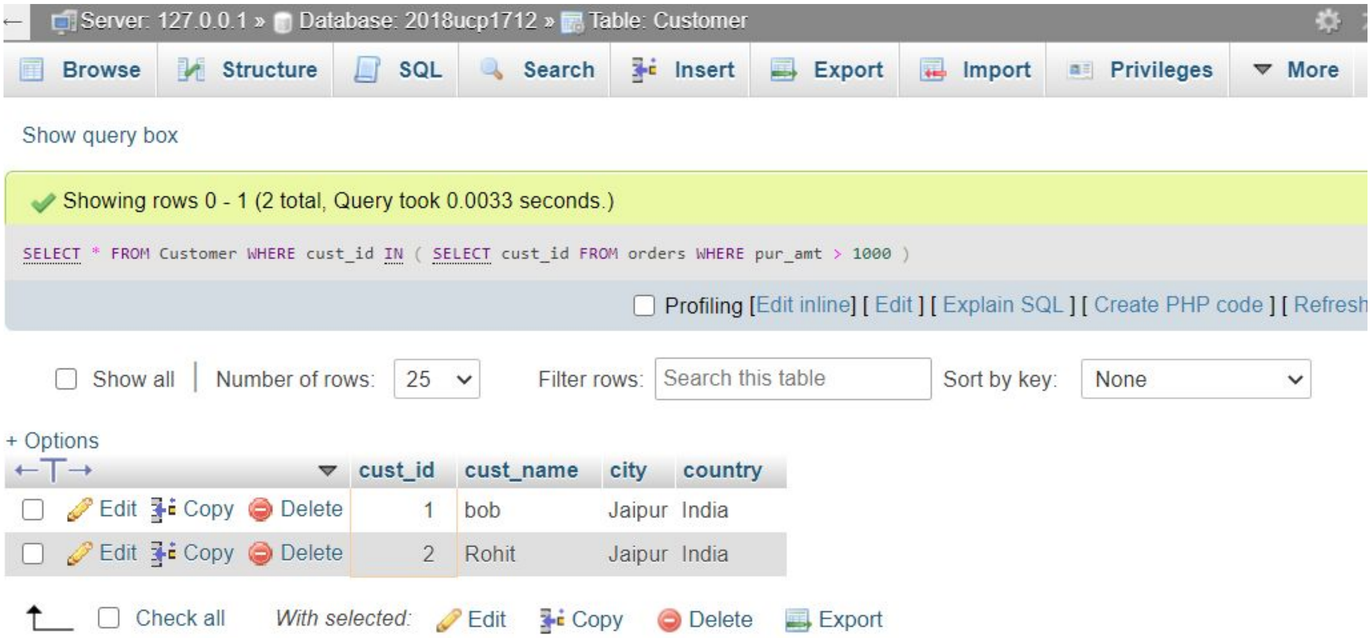
✓ 2 rows inserted. (Query took 0.0452 seconds.)

```
INSERT INTO Salesman VALUES(2, "Vladimir", "Moscow", 600), (3, "Jamal", "Delhi", 1200)
```

[Edit inline] [Edit] [Create PHP code]

4. Write a query to list all customers with order amount greater than 1000 using subquery.

```
SELECT * FROM Customer WHERE cust_id IN ( SELECT cust_id FROM orders WHERE
pur_amt > 1000 );
```


Server: 127.0.0.1 » Database: 2018ucp1712 » Table: Customer
Browse
Structure
SQL
Search
Insert
Export
Import
Privileges
More
Show query box
Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh
□ Show all
Number of rows: 25
Filter rows: Search this table
Sort by key: None
+ Options
□ Check all
With selected: Edit Copy Delete Export

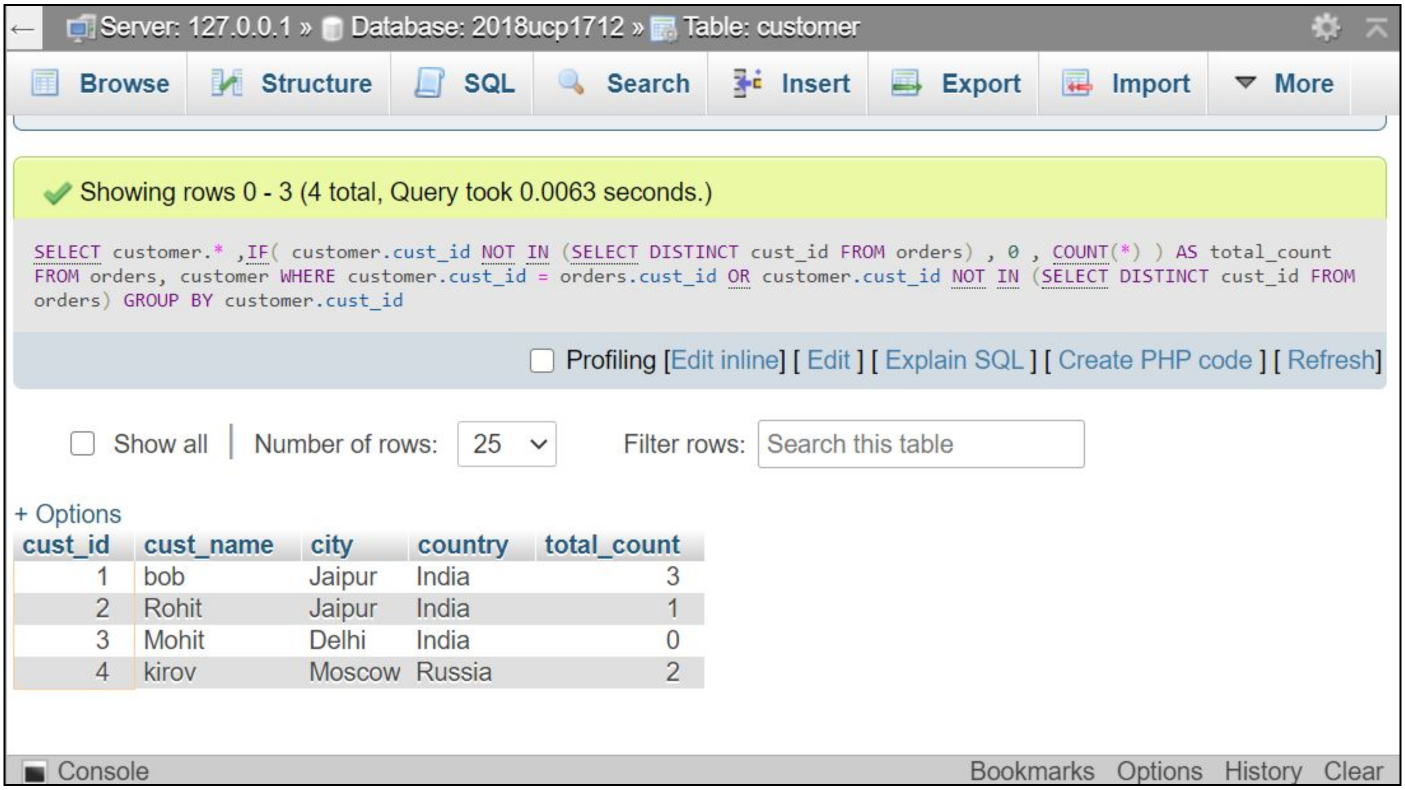
✓ Showing rows 0 - 1 (2 total, Query took 0.0033 seconds.)

```
SELECT * FROM Customer WHERE cust_id IN ( SELECT cust_id FROM orders WHERE pur_amt > 1000 )
```

	cust_id	cust_name	city	country
□ Edit Copy Delete	1	bob	Jaipur	India
□ Edit Copy Delete	2	Rohit	Jaipur	India

5. Write a query to list all customers with their total number of orders using subquery.

```
SELECT customer.* ,IF( customer.cust_id NOT IN
                      (SELECT DISTINCT cust_id FROM orders)
                     , 0 , COUNT(*)
                    ) AS total_count
FROM orders, customer
WHERE customer.cust_id = orders.cust_id
OR customer.cust_id NOT IN (SELECT DISTINCT cust_id FROM orders)
GROUP BY customer.cust_id;
```


Server: 127.0.0.1 » Database: 2018ucp1712 » Table: customer
Browse
Structure
SQL
Search
Insert
Export
Import
More
Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
Console
Bookmarks
Options
History
Clear

Showing rows 0 - 3 (4 total, Query took 0.0063 seconds.)

```
SELECT customer.* ,IF( customer.cust_id NOT IN (SELECT DISTINCT cust_id FROM orders) , 0 , COUNT(*) ) AS total_count
FROM orders, customer WHERE customer.cust_id = orders.cust_id OR customer.cust_id NOT IN (SELECT DISTINCT cust_id FROM
orders) GROUP BY customer.cust_id
```

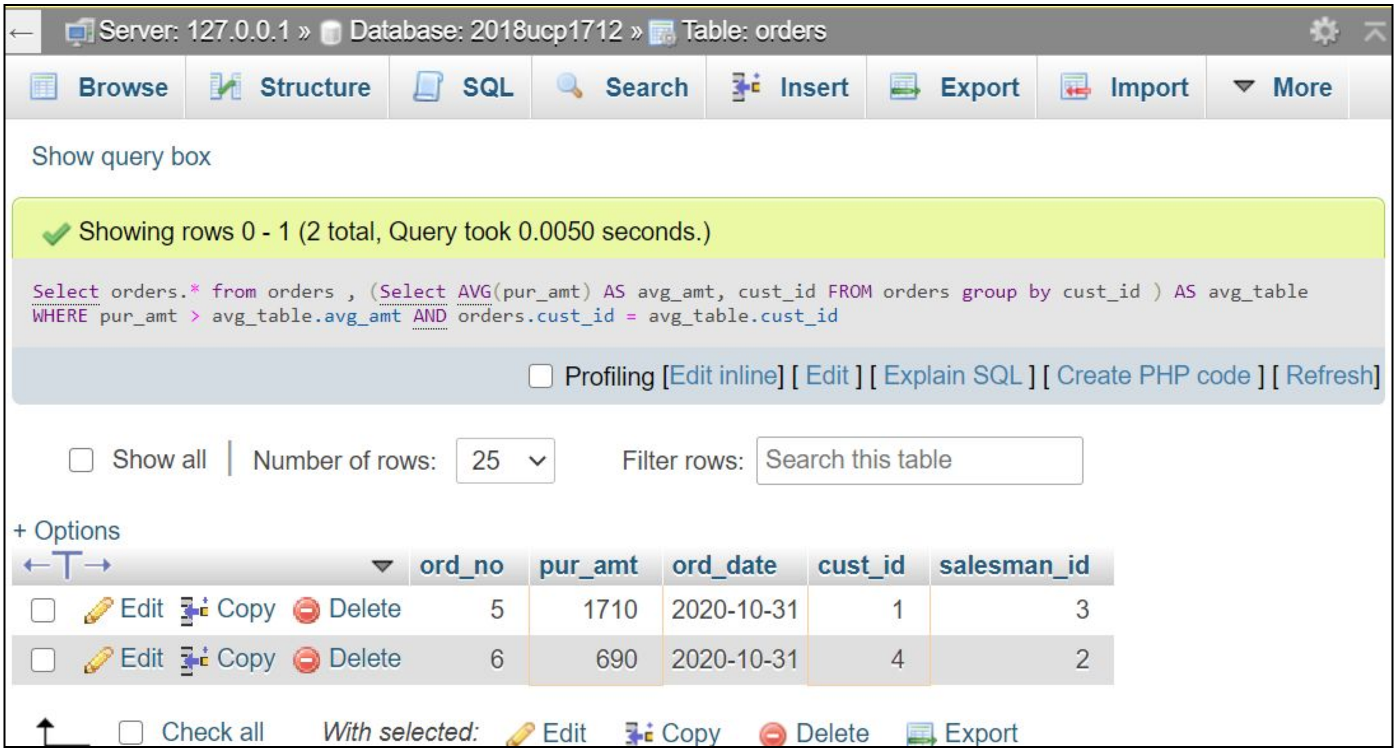
□ Show all | Number of rows: 25 Filter rows: Search this table

+ Options

cust_id	cust_name	city	country	total_count
1	bob	Jaipur	India	3
2	Rohit	Jaipur	India	1
3	Mohit	Delhi	India	0
4	kirov	Moscow	Russia	2

6. Write a query to find all the orders with order amounts which are above average amount for their customers using subquery.

*Select orders.**
from orders , (Select AVG(pur_amt) AS avg_amt, cust_id FROM orders group by cust_id) AS avg_table
WHERE pur_amt > avg_table.avg_amt
AND orders.cust_id = avg_table.cust_id;


Server: 127.0.0.1 » Database: 2018ucp1712 » Table: orders
Browse
Structure
SQL
Search
Insert
Export
Import
More
Show query box
Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
Show all
Number of rows: 25
Filter rows: Search this table
+ Options
Check all
With selected: Edit Copy Delete Export

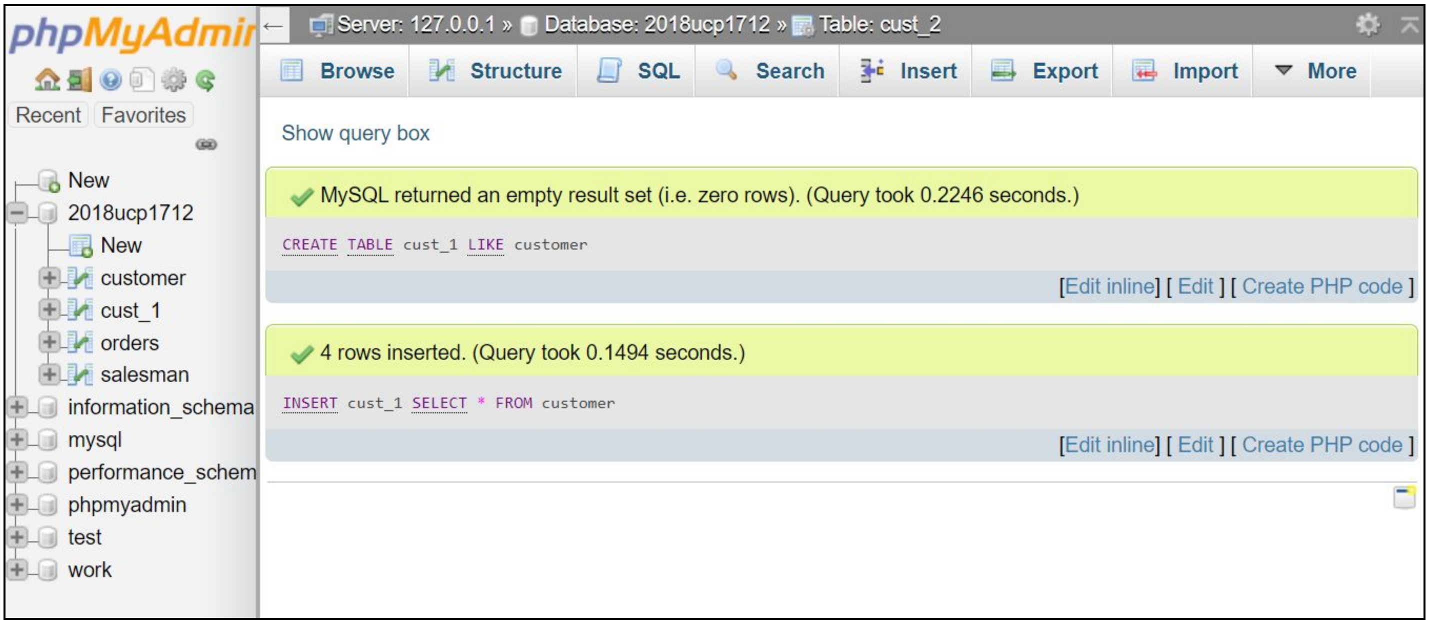
Showing rows 0 - 1 (2 total, Query took 0.0050 seconds.)

```
Select orders.* from orders , (Select AVG(pur_amt) AS avg_amt, cust_id FROM orders group by cust_id ) AS avg_table
WHERE pur_amt > avg_table.avg_amt AND orders.cust_id = avg_table.cust_id
```

	ord_no	pur_amt	ord_date	cust_id	salesman_id
□ Edit Copy Delete	5	1710	2020-10-31	1	3
□ Edit Copy Delete	6	690	2020-10-31	4	2

7. Given that you have a table named Customer in your database. Write a Create table statement to create another table Cust_1 with same structure and data as that of Customer.

CREATE TABLE cust_1 LIKE customer;
*INSERT cust_1 SELECT * FROM customer;*



MySQL returned an empty result set (i.e. zero rows). (Query took 0.2246 seconds.)

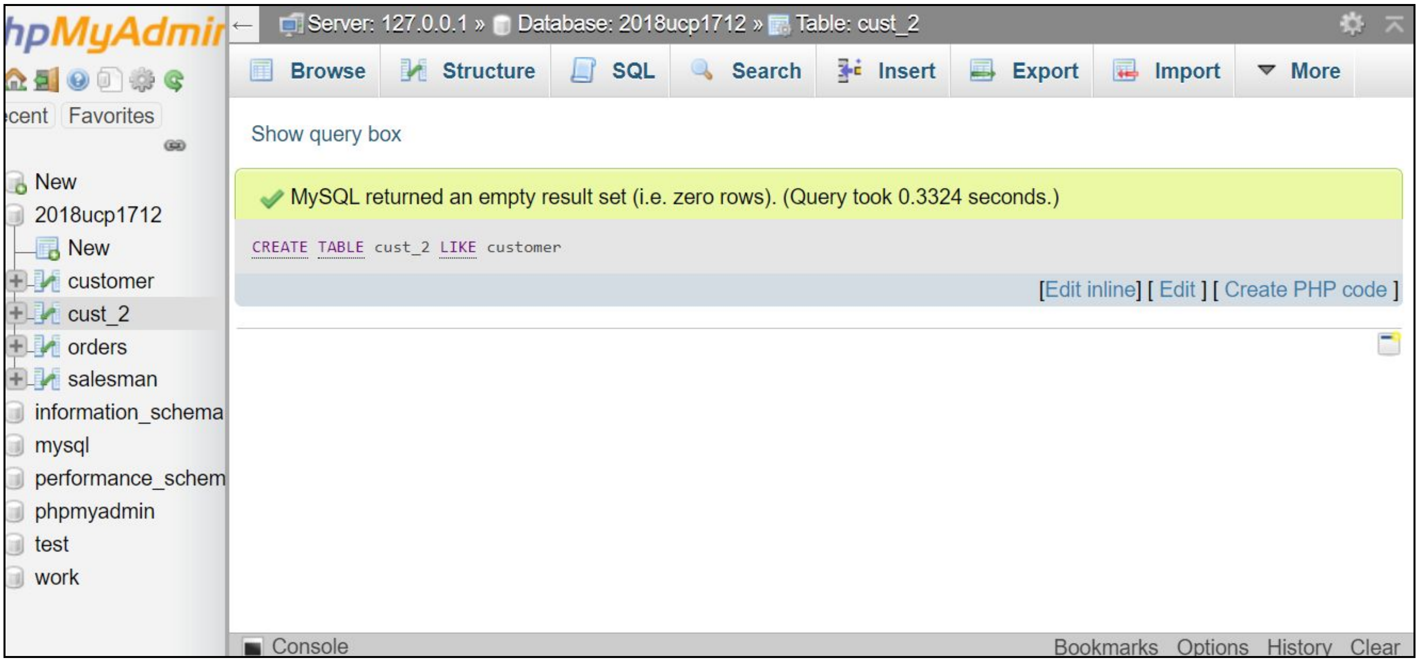
```
CREATE TABLE cust_1 LIKE customer
```

4 rows inserted. (Query took 0.1494 seconds.)

```
INSERT cust_1 SELECT * FROM customer
```

8. Given that you have a table named Customer in your database. Write a Create table statement to create another table Cust_2 with same structure (but with no data) as that of Customer.

CREATE TABLE cust_2 LIKE customer;



✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.3324 seconds.)

```
CREATE TABLE cust_2 LIKE customer
```

==
==