

DBMS Practical Implementation, Lab 7.

0. Inserting Necessary Values:

```
INSERT INTO `client_master` (`client_no`, `name`, `address`, `city`, `pincode`,  
`state`, `Bal_due`) VALUES ('C0004', 'Rahul Desai', 'The Height, Indra Nagar',  
'Mumbai', '400080', 'Maharashtra', '2000');
```

```
INSERT INTO `product_master` (`product_no`, `description`, `profit_percent`,  
`unit_measure`, `qty_on_hand`, `reorder_level`, `cost_price`, `new_price`)  
VALUES ('1014', '1.44 Drive', '12', '1', '250', '500', '2500', '3000');
```

```
INSERT INTO `sales_order` (`order_no`, `order_date`, `client_no`, `dely_addr`,  
`salesman_no`, `dely_type`, `billed_yn`, `Dely_date`, `order_status`) VALUES  
('O191', '1995-04-01', 'C0004', 'The height, Indra Nagar', 'S001', 'Y', 'Y', '1995-  
04-14', 'Shipped');
```

```
INSERT INTO `sales_order_details` (`order_no`, `product_no`, `qty_ordered`,  
`qty_disp`, `product_rate`) VALUES ('O191', '1014', '25', '25', '3000');
```

1. Find the product no. and description of non-moving products i.e. products not being sold.

```
SELECT product_master.description, product_master.product_no
FROM product_master
WHERE product_master.product_no NOT IN(
SELECT sales_order_details.product_no FROM sales_order_details
);
```

The screenshot shows the phpMyAdmin web interface. The left sidebar displays a database structure tree with various databases like 'cresendo_webgeeks', 'dbmspracs', 'client_master', 'dept', 'employee', 'product_master', 'sales_order', 'sales_order_details', 'hospital', 'information_schema', 'mysql', 'performance_schema', 'phpmyadmin', 'students', 'test', 'todolist', 'university', and 'vinyas'. The main panel is titled 'Table: product_master' and shows the results of an SQL query. The query is: `SELECT product_master.description, product_master.product_no FROM product_master WHERE product_master.product_no NOT IN(SELECT sales_order_details.product_no FROM sales_order_details)`. The results show 3 rows: 'Ceiling Fan' (1002), 'Face Mask' (1007), and 'chair' (7065). Below the results, there are options to 'Check all', 'With selected', 'Edit', 'Copy', 'Delete', and 'Export'. At the bottom, there is a 'Bookmark this SQL query' section with a label field and a checkbox 'Let every user access this bookmark'.

description	product_no
Ceiling Fan	1002
Face Mask	1007
chair	7065

2. Find the customer name, address for the client who has placed order no 'O191'

```
SELECT client_master.name, client_master.address
```

```
FROM client_master
```

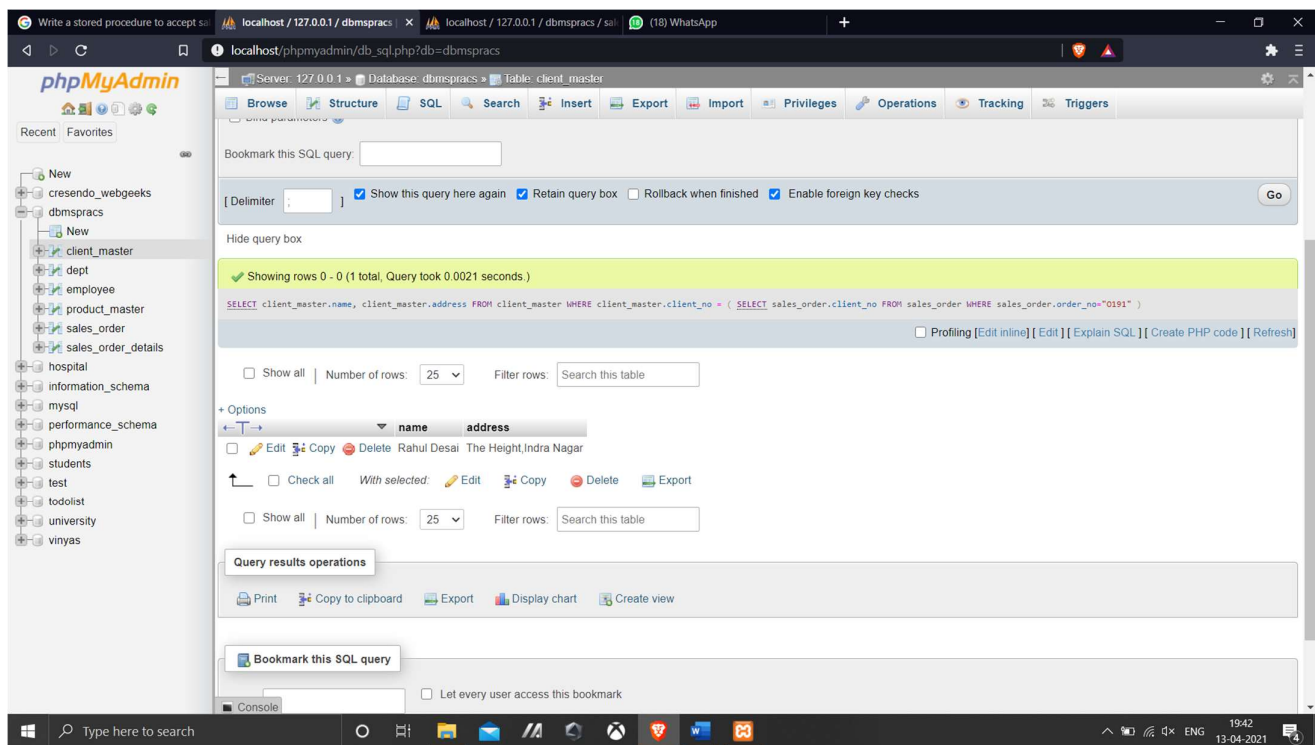
```
WHERE client_master.client_no = (
```

```
SELECT sales_order.client_no
```

```
FROM sales_order
```

```
WHERE sales_order.order_no="O191"
```

```
);
```



3. Find the clients names who have placed orders before the month of May'96

```
SELECT *  
FROM client_master  
WHERE client_master.client_no = (  
SELECT sales_order.client_no  
FROM sales_order  
WHERE sales_order.order_date < "1996-05-01"  
);
```

The screenshot shows the phpMyAdmin web interface. The left sidebar displays a database structure with tables like 'client_master', 'sales_order', and 'sales_order_details'. The main panel shows the 'client_master' table with a query executed: `SELECT * FROM client_master WHERE client_master.client_no = (SELECT sales_order.client_no FROM sales_order WHERE sales_order.order_date < "1996-05-01")`. The query results show one row with the following data:

client_no	name	address	city	pincode	state	Bal_due
C0004	Rahul Desai	The Height, Indra Nagar	Mumbai	400080	Maharashtra	2000.00

Below the table, there are options to 'Print', 'Copy to clipboard', 'Export', 'Display chart', and 'Create view'. At the bottom, there is a 'Bookmark this SQL query' section with a label input field and a checkbox 'Let every user access this bookmark'.

4. Find out if the product '1.44 Drive' has been ordered by any client and print the client_no, name to whom it was sold

```
SELECT client_master.client_no, client_master.name, client_master.address  
FROM client_master WHERE client_master.client_no = (
```

```
    SELECT sales_order.client_no FROM sales_order WHERE  
sales_order.order_no = (
```

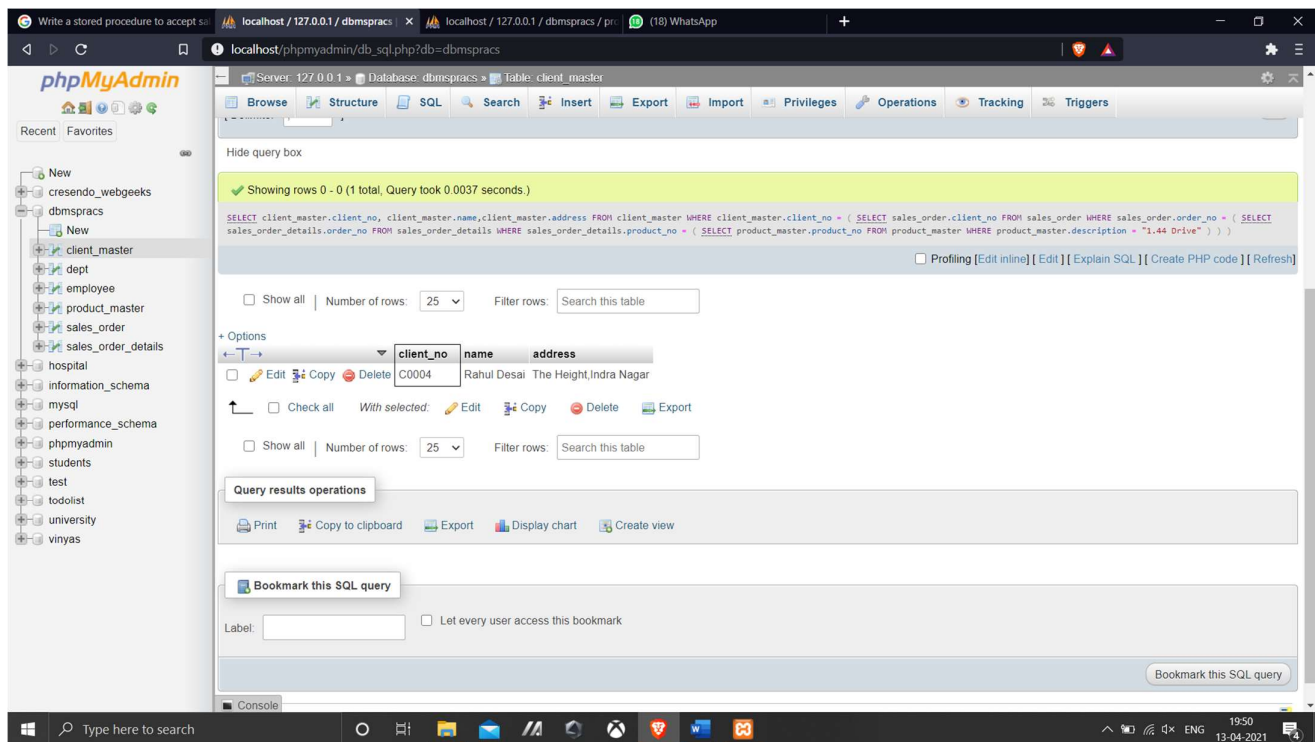
```
    SELECT sales_order_details.order_no FROM sales_order_details WHERE  
sales_order_details.product_no = (
```

```
        SELECT product_master.product_no FROM product_master  
WHERE product_master.description = "1.44 Drive"
```

```
)
```

```
)
```

```
);
```



5. Find the names of clients who have placed orders worth Rs. 10000 or more

SELECT client_master.client_no, client_master.name, client_master.address

FROM client_master

WHERE client_master.client_no IN (

SELECT sales_order.client_no

FROM sales_order NATURAL JOIN sales_order_details

WHERE sales_order_details.product_rate*sales_order_details.qty_disp >= 10000

);

The screenshot shows the phpMyAdmin web interface. The left sidebar displays a database structure with the following items: New, cresendo_webgeeks, dbmspracs, New, client_master, dept, employee, product_master, sales_order, sales_order_details, hospital, information_schema, mysql, performance_schema, phpmyadmin, students, test, todolist, university, and vinyas. The main panel is titled 'Server: 127.0.0.1 - Database: dbmspracs - Table: client_master'. It shows the results of an SQL query: 'SELECT client_master.client_no, client_master.name, client_master.address FROM client_master WHERE client_master.client_no IN (SELECT sales_order.client_no FROM sales_order NATURAL JOIN sales_order_details WHERE sales_order_details.product_rate*sales_order_details.qty_disp >= 10000)'. The query returned 2 rows. The table below shows the results:

client_no	name	address
C0002	Ivan Sayross	16 Lok street, Malad
C0004	Rahul Desai	The Height, Indra Nagar

Below the table, there are options to 'Check all', 'With selected', 'Edit', 'Copy', 'Delete', and 'Export'. There is also a 'Query results operations' section with 'Print', 'Copy to clipboard', 'Export', 'Display chart', and 'Create view' buttons. At the bottom, there is a 'Bookmark this SQL query' section with a 'Label' input field and a 'Let every user access this bookmark' checkbox. The bottom status bar shows the date and time: 13-04-2021 19:56.

6. Retrieve all the orders placed by a client named 'Rahul Desai' from the sales_order table.

```
SELECT *
```

```
FROM sales_order
```

```
WHERE sales_order.client_no IN (
```

```
    SELECT client_master.client_no
```

```
    FROM client_master
```

```
    WHERE client_master.name = "Rahul Desai"
```

```
);
```

The screenshot shows the phpMyAdmin web interface. The left sidebar displays a database structure tree with 'dbmspracs' selected. The main panel shows the 'Table: sales_order' view. A SQL query is entered in the query box: `SELECT * FROM sales_order WHERE sales_order.client_no IN (SELECT client_master.client_no FROM client_master WHERE client_master.name = "Rahul Desai")`. The query results are displayed in a table with the following data:

order_no	order_date	client_no	dely_addr	salesman_no	dely_type	billed_yn	Dely_date	order_status
O191	1995-04-01	C0004	The height, Indra Nagar	S001	Y	Y	1995-04-14	Shipped

Below the table, there are options for 'Query results operations' including Print, Copy to clipboard, Export, Display chart, and Create view. At the bottom, there is a 'Bookmark this SQL query' section with a label input field and a checkbox 'Let every user access this bookmark'.

7. Retrieve name, address, city of all the clients who have placed an order through salesman no 's001'.

```
SELECT client_master.client_no, client_master.name, client_master.address,  
client_master.city
```

```
FROM client_master
```

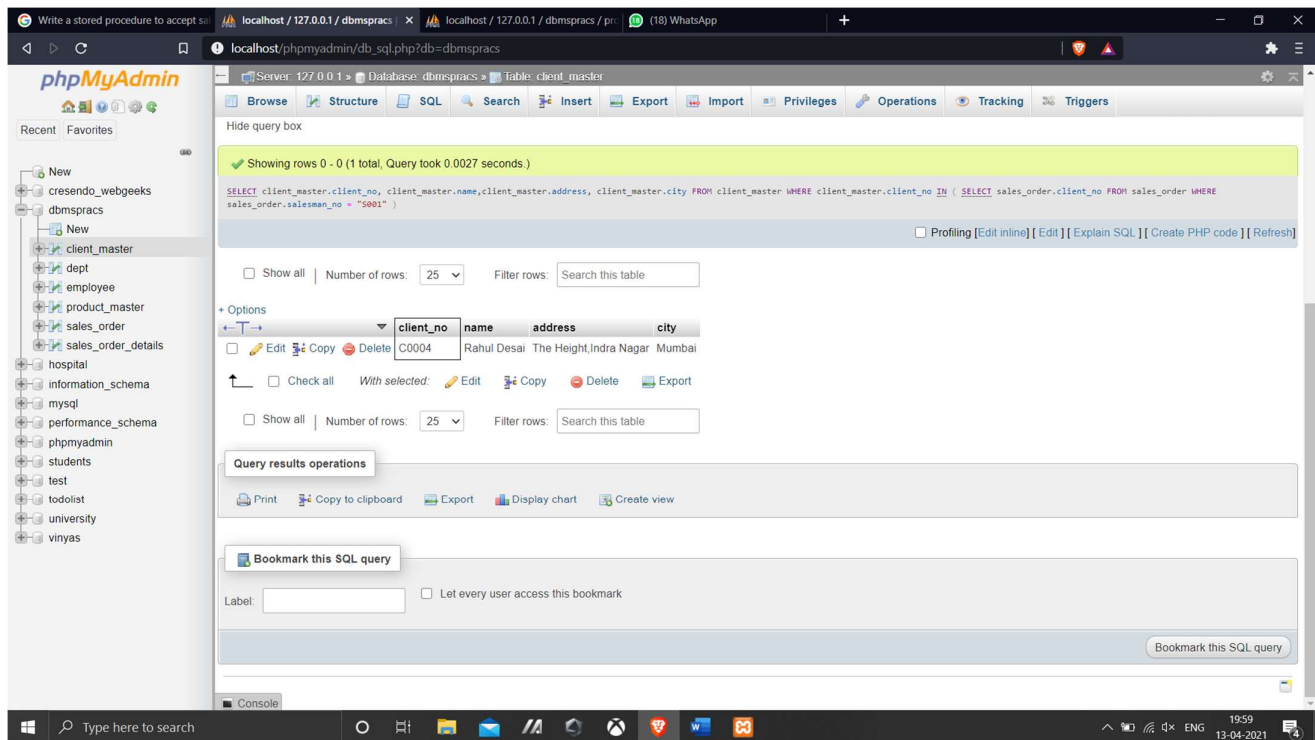
```
WHERE client_master.client_no IN (
```

```
    SELECT sales_order.client_no
```

```
FROM sales_order
```

```
WHERE sales_order.salesman_no = "S001"
```

```
);
```



Exp-7 - Portlab

Q) What is Incremental Updates?
Ans

An incremental update adds new records to a project data set from Hive table.

It performs a partial update of a project data set by selecting adding new and modified records. The data set should be a project data set.

The Incremental update operation fetches a subset of the records in the source Hive table. The subset is determined by using a filtering predicate that specifies the Hive table columns that holds the records and value of the records to fetch. The records in the subset batch are ingested as follows:

- If a record is brand new (does not exist in the data set), it is added to the data set.
- If a record already exists in the data set but its content has been changed, it replaces the record in the data set.

(2) What is UPDATE Cascade & DELETE Cascade with suitable example?

Ans

DELETE Cascade:- When we create a foreign key using this option, it deletes the referencing rows in the child table when the referenced row is deleted in the parent table which has a primary key.

UPDATE Cascade:- When we create a foreign key using UPDATE CASCADE the referencing rows are updated in the child table when the referenced row is updated in the parent table which has a primary key.

Example:-

Suppose that we have Two tables:- buildings and rooms. In this database model, each building has one or many rooms. However, each room belongs to one only building. A room would not exist without a building.

When you delete a row from the buildings table, you also want to delete all rows in the rooms table that references to the rows in the buildings table. For example, when we delete a row with building no. 1 in the building table, we also want the rows in the rooms table that refer to building no. 1 will be also removed.

Syntax:-

CREATE TABLE rooms (

room_no INT PRIMARY AUTO_INCREMENT;

room_name VARCHAR(255) NOT NULL;

building_no INT NOT NULL;

FOREIGN KEY (building_no)

REFERENCES buildings (building_no)

ON

DELETE CASCADE

);