

TITLE: DML, DQL AND REFERENCE INTEGRITY

THEORY:

- 1. DML:** Data Manipulation Language (DML) is the SQL commands that deals with the manipulation of data present in the database, and this includes most of the SQL commands.
- 2. DQL:** Data Query Language (DQL) is the SQL commands that is used to fetch the data from the database.
- 3. Insert:** This command is used to insert data into a table.
- 4. Update:** This command is used to update existing data with a table.
- 5. Delete:** This command is used to delete records from a database table.
- 6. Select:** This command is used to select data from a database.
- 7. Primary key:** It is the key which is used to identify one and only one instance of an entity uniquely.
- 8. Foreign key:** They are the column of the table which is used to point to the primary key of another table.
- 9. Relational algebra:** It is the procedural query language used to query to database to access in different ways.
- 10. Select Operation:** The select operation selects tuples that satisfy a given predicate. It is denoted by σ .
- 11. Project operation:** This operation shows the list of those whose attributes that we wish to appear in the result. It is denoted by π .
- 12. Reference integrity:** It is a property of data stating that all its references are valid. In the context of relational databases, it requires that if a value of one attribute of a relation references a value of another attribute, then the referenced value must exist.
 - a. Primary key constraint:** This constraint uniquely identifies each record in a table. Primary keys must contain unique values and cannot contain null values.
 - b. Foreign key constraint:** This constraint species that the key can only contain values that are in the referenced primary key and thus ensures the referential integrity of data is joined on the two keys.

1.

DISCUSSION:

At first, we created a database named 'customer2' and we created a table 'customer1', 'item1', and 'orders' inside it using 'create' command. Inside these tables, we created some attributes by defining their datatypes. Finally, we displayed the created tables using the 'select' command.

OUTPUT:

Results		Messages		
c_id	c_name	c_address	c_phone	c_age
i_id	i_name	i_price		
o_id	c_id	i_id		

2.

```
insert into customer1 values(1, 'Danny', 'London', 9841246687, 20);
insert into customer1 values(1, 'Jacob', 'Kathmandu', 9841225687, 22);
insert into customer1 values(3, 'Hari', 'Pokhara', 9841845687, 21);
insert into customer1 values(4, 'Lily', 'Chitwan', 9841345687, 23);
insert into customer1 values(5, 'Gopal', 'Hetauda', 9841241687, 22);

select *from customer1;
select *from item1;
select *from orders;
```

DISCUSSION:

Here, we provided values to the table customer using 'insert' command. We defined c_id as a primary key while creating the table so the value of primary key must be NOT NULL and Unique. At first, we provided same value of c_id as '1' twice which resulted in a **Primary key constraint**. Even though other values were also same, only the primary key value will create an error. So, we again executed this code by providing unique values of c_id and this error was resolved, and the code executed successfully.

Messages

Msg 2627, Level 14, State 1, Line 23
Violation of PRIMARY KEY constraint 'PK_customer_213EE77454D9B89D'. Cannot insert duplicate key in object 'dbo.customer1'. The duplicate key value is (1).
The statement has been terminated.

Completion time: 2023-01-16T20:10:51.9187854+05:45

```
insert into customer1 values(1, 'Danny', 'London', 9841246687, 20);
insert into customer1 values(2, 'Jacob', 'Kathmandu', 9841225687, 22);
insert into customer1 values(3, 'Hari', 'Pokhara', 9841845687, 21);
insert into customer1 values(4, 'Lily', 'Chitwan', 9841345687, 23);
insert into customer1 values(5, 'Gopal', 'Hetauda', 9841241687, 22);
```

Results		Messages			
	c_id	c_name	c_address	c_phone	c_age
1	1	Danny	London	9841246687	20
2	2	Jacob	Kathmandu	9841225687	22
3	3	Hari	Pokhara	9841845687	21
4	4	Lily	Chitwan	9841345687	23
5	5	Gopal	Hetauda	9841241687	22

3.

```
insert into orders values(1,6,1);
insert into orders values(2,2,2);
insert into orders values(3,3,3);
insert into orders values(4,4,4);
select *from customer1;
select *from item1;
select *from orders;
```

DISCUSSION:

Here, the orders table has two foreign keys that references the column consisting primary key of two different tables. While inserting the values into this table, we tried to provide the value of foreign key that is not present in the primary key column which resulted in a **foreign key constraint**. Here, we inserted '6' which is not present in the c_id column of the customer1 table which gave us an error. So, we again executed this code by providing the value of foreign key that is present in the primary key column and this error was resolved and the code ran successfully.

OUTPUT:

Messages

Msg 547, Level 16, State 0, Line 28

The INSERT statement conflicted with the FOREIGN KEY constraint "FK_orders_c_id_4CA06362". The conflict occurred in database "customer2", table "dbo.customer1", column 'c_id'.
The statement has been terminated.

Completion time: 2023-01-16T20:20:34.5624303+05:45

```
insert into orders values(1,1,1);
insert into orders values(2,2,2);
insert into orders values(3,3,3);
insert into orders values(4,4,4);
```

Results Messages

	o_id	c_id	i_id
1	1	1	1
2	2	2	2
3	3	3	3
4	4	4	4

4.

```
insert into orders values(1,1,1);
insert into orders values(2,2,2);
insert into orders values(3,3,3);
insert into orders values(4,4,4);
delete from customer1 where c_id=1;
select *from customer1;
select *from item1;
select *from orders;
```

DISCUSSION:

Here, customer1 is a base table with primary key that is providing data to its child table orders1 with foreign key. We tried to delete a value from the base table and there is a possibility that the child table may be accessing data from the base table, so while deleting a value from the base table will result **reference constraint**. We cannot insert data into the child table, and we cannot delete data from the base table because this will cause an error. Now, we again executed this code by deleting data from the child table and the code ran successfully.

OUTPUT:

Messages

Msg 547, Level 16, State 0, Line 42
The DELETE statement conflicted with the REFERENCE constraint "FK_orders_c_id_4CA06362". The conflict occurred in database "customer2", table "dbo.orders", column 'c_id'.
The statement has been terminated.

Completion time: 2023-01-16T20:32:42.7887247+05:45

```
delete from orders where c_id=1;
select *from customer1;
select *from item1;
select *from orders;
```

Results		Messages	
	o_id	c_id	i_id
1	2	2	2
2	3	3	3
3	4	4	4

5.

```
update customer1 set c_name='David' where c_name='Hari';
select *from customer1;
select *from item1;
select *from orders;
```

DISCUSSION:

Here, we updated the value in c_name from the table customer1 with another value using the 'update' command. We provided the attribute which is to be updated with existing value and with the new value. After using the 'select' command, we displayed the updated table with new value in c_name.

OUTPUT:

Results		Messages			
	c_id	c_name	c_address	c_phone	c_age
1	1	Danny	London	9841246687	20
2	2	Jacob	Kathmandu	9841225687	22
3	3	David	Pokhara	9841845687	21
4	4	Lily	Chitwan	9841345687	23
5	5	Gopal	Hetauda	9841241687	22

6.

```
select c_name, c_phone from customer1;
select *from customer1;
select *from item1;
select *from orders;
```

DISCUSSION:

Here, we used the projection operation. We selected a certain column: c_name and c_phone from table customer1 and displayed only the selected columns using the 'select' command.

OUTPUT:

Results		Messages	
	c_name	c_phone	
1	Danny	9841246687	
2	Jacob	9841225687	
3	David	9841845687	
4	Lily	9841345687	
5	Gopal	9841241687	

RELATIONAL ALGEBRA:

$\Pi_{c_name, c_phone}(\text{customer1})$

- a. Display the information of the customer whose age is greater than 21.

```
--display the information of customer whose age is greater than 21  
select *from customer1 where c_age>21;
```

Results		Messages			
	c_id	c_name	c_address	c_phone	c_age
1	2	Jacob	Kathmandu	9841225687	22
2	4	Lily	Chitwan	9841345687	23
3	5	Gopal	Hetauda	9841241687	22

RELATIONAL ALGEBRA:

$\sigma_{c_age>21}(\text{customer1})$

- b. Display the information of the customer whose age is between 20 and 23.

```
--display the information of customer whose age is between 20 and 23  
select *from customer1 where c_age>20 and c_age<23;
```

Results		Messages			
	c_id	c_name	c_address	c_phone	c_age
1	2	Jacob	Kathmandu	9841225687	22
2	3	David	Pokhara	9841845687	21
3	5	Gopal	Hetauda	9841241687	22

RELATIONAL ALGEBRA:

$\sigma_{c_age>20 \wedge c_age<23}(\text{customer1})$

- c. Display the info of the customer whose name starts with J and ends with b.

```
--display the information of customer whose name begins with J and ends with b  
select * from customer1 where c_name like 'J%' and c_name like '%b';
```

Results		Messages			
	c_id	c_name	c_address	c_phone	c_age
1	2	Jacob	Kathmandu	9841225687	22

RELATIONAL ALGEBRA:

$\sigma_{c_name \text{ startswith 'J' } \wedge c_name \text{ endswith 'b'}}(\text{customer1})$

- d. Display the name of items whose price is greater than 300.

```
--display the name of items whose price is greater than 300  
select i_name, i_price from item1 where i_price > 300;
```

	i_name	i_price
1	Mango	400
2	Pineapple	600

RELATIONAL ALGEBRA:

$\Pi_{i_name, i_price}(\sigma_{i_price > 300}(\text{item1}))$

DISCUSSION:

Here, we studied about the projection and selection operations separately and combinedly. We also wrote the relational algebra for each operation performed. Finally, we displayed the table.

CONCLUSION:

In this lab, we studied about the Data Manipulation Language (DML) commands, Data Query Language (DQL) commands and about the referential integrity. We learned about the constraints and how to resolve the errors generated by the constraints.

