Experiment No.4
Apply DML commands for the specified system
Date of Performance:
Date of Submission:

Aim :- Write insert query to insert rows for each table created of your database management system. Use update and delete commands to manipulate the inserted values in the table.

Objective :- To learn commands of Data Manipulation Language(DML) to insert, update or delete the values in the database system.

Theory:

Data Manipulation Language (DML) is a subset of SQL (Structured Query Language) used for managing data within relational database management systems (RDBMS). DML commands are used to perform operations such as inserting, updating, and deleting data from database tables.

1. Inserting Data

The INSERT statement is used to add new rows of data into a table. It specifies the table to insert data into and provides values or expressions for each column in the new row. If a column list is not specified, values must be provided for all columns in the table in the order they were defined.

Syntax:-

INSERT INTO table_name (column1, column2, column3) VALUES (value1, value2, value3);

2. Updating Data

The UPDATE statement is used to modify existing data within a table. It allows you to change the values of one or more columns in one or more rows based on specified conditions. If no condition is specified, all rows in the table will be updated.

Syntax:

UPDATE table_name SET column1 = value1, column2 = value2 WHERE condition;

3. Deleting Data

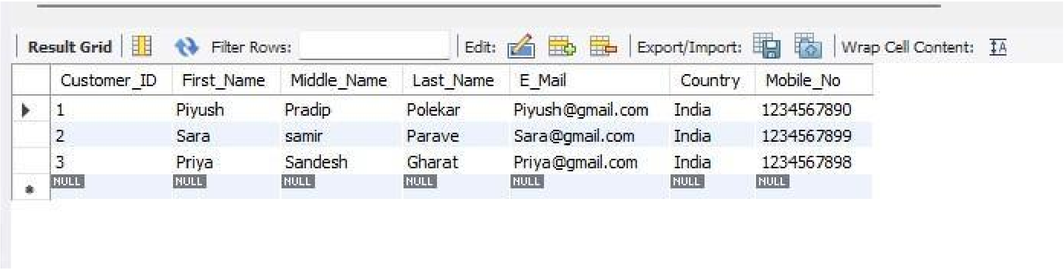
The DELETE statement is used to remove one or more rows from a table based on specified conditions. If no condition is specified, all rows in the table will be deleted.

Syntax:

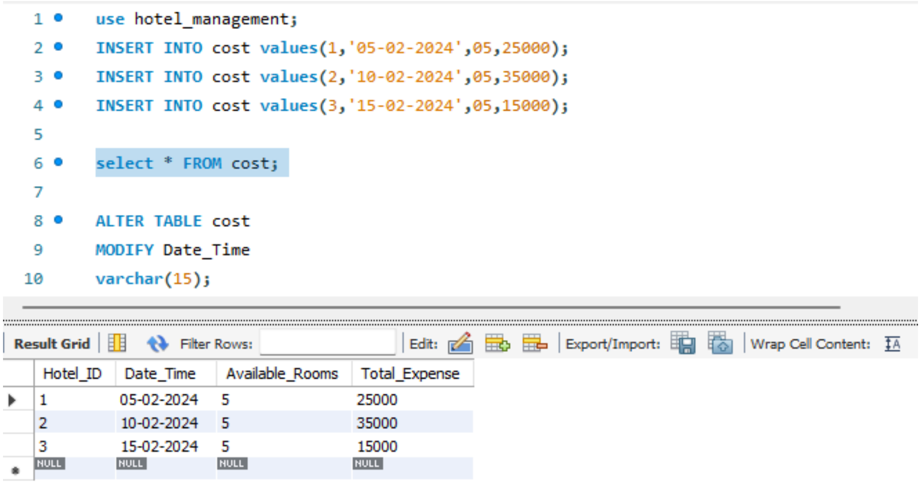
DELETE FROM table_name WHERE condition;

Implementation:

INSERT:-

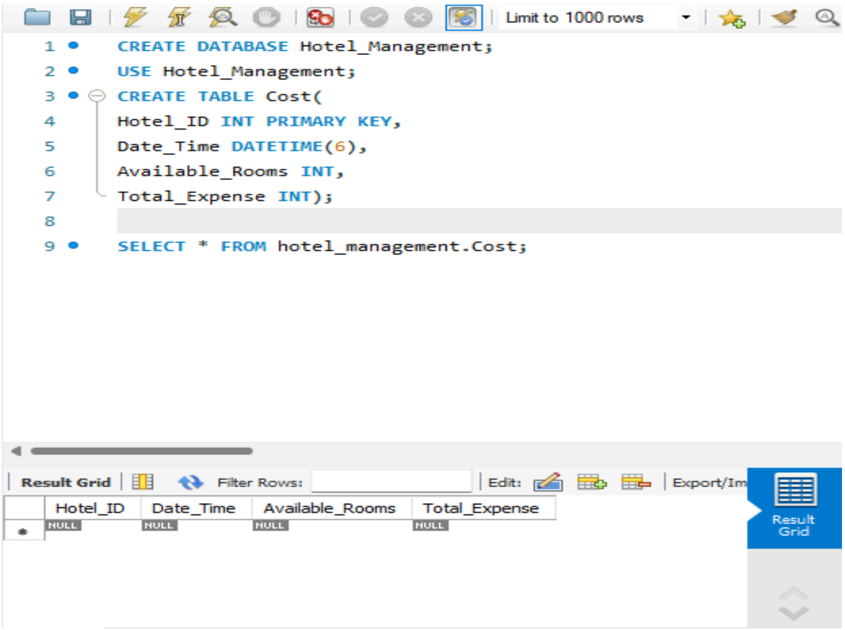


Customer_ID	First_Name	Middle_Name	Last_Name	E_Mail	Country	Mobile_No
1	Piyush	Pradip	Polekar	Piyush@gmail.com	India	1234567890
2	Sara	samir	Parave	Sara@gmail.com	India	1234567899
3	Priya	Sandesh	Gharat	Priya@gmail.com	India	1234567898
NULL	NULL	NULL	NULL	NULL	NULL	NULL



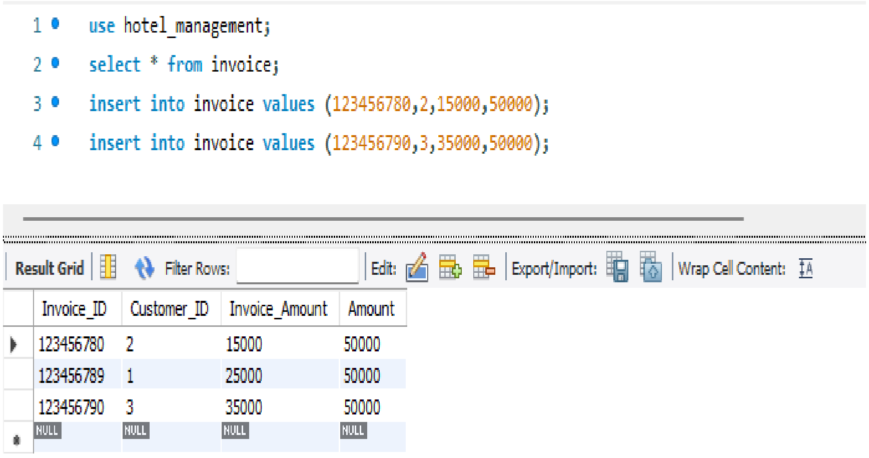
```
use hotel_management;
INSERT INTO cost values(1,'05-02-2024',05,25000);
INSERT INTO cost values(2,'10-02-2024',05,35000);
INSERT INTO cost values(3,'15-02-2024',05,15000);

select * FROM cost;

ALTER TABLE cost
MODIFY Date_Time
varchar(15);
```

Hotel_ID	Date_Time	Available_Rooms	Total_Expense
1	05-02-2024	5	25000
2	10-02-2024	5	35000
3	15-02-2024	5	15000
NULL	NULL	NULL	NULL



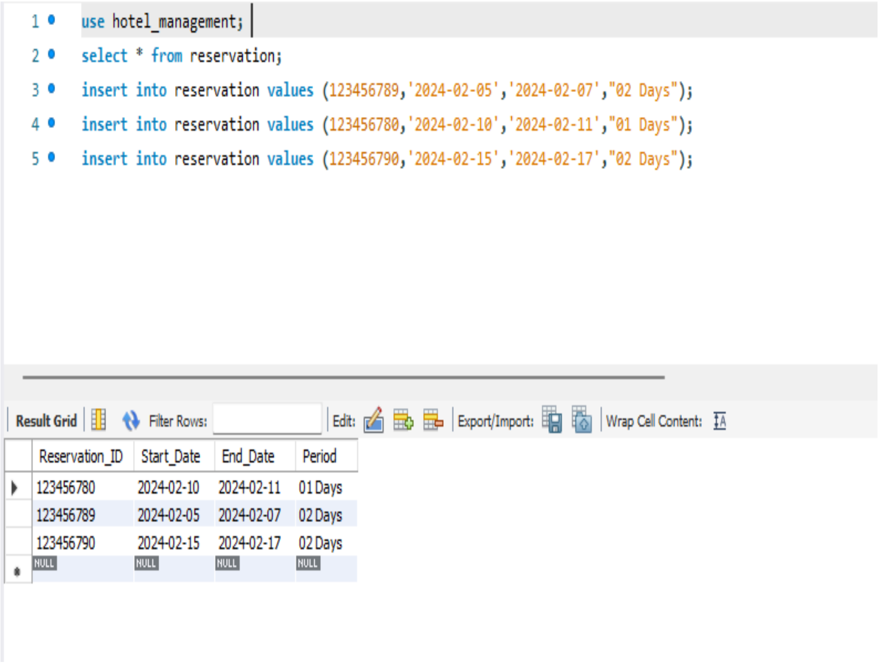
```
CREATE DATABASE Hotel_Management;
USE Hotel_Management;
CREATE TABLE Cost(
Hotel_ID INT PRIMARY KEY,
Date_Time DATETIME(6),
Available_Rooms INT,
Total_Expense INT);

SELECT * FROM hotel_management.Cost;
```

Hotel_ID	Date_Time	Available_Rooms	Total_Expense
NULL	NULL	NULL	NULL



```
use hotel_management;
select * from invoice;
insert into invoice values (123456780,2,15000,50000);
insert into invoice values (123456790,3,35000,50000);
```

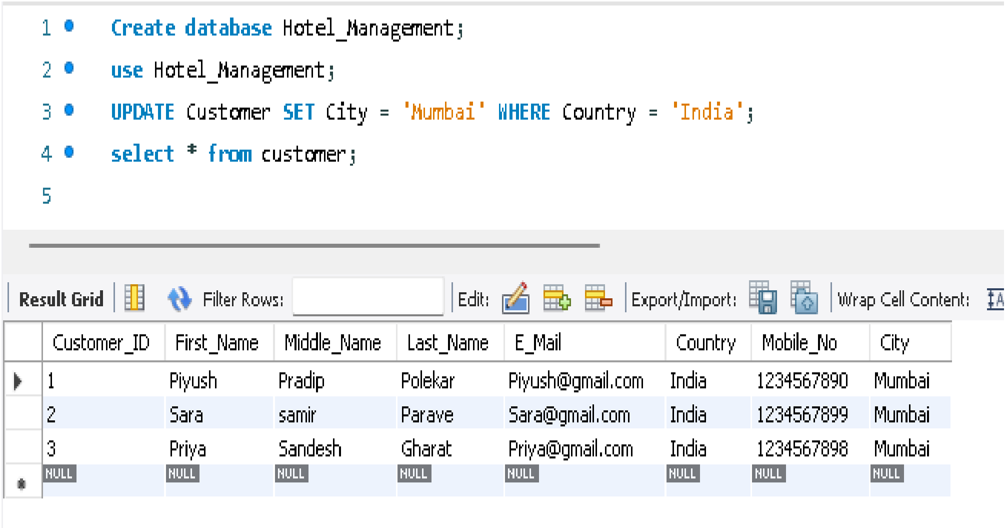
Invoice_ID	Customer_ID	Invoice_Amount	Amount
123456780	2	15000	50000
123456789	1	25000	50000
123456790	3	35000	50000
NULL	NULL	NULL	NULL



```sql
use hotel_management;
select * from reservation;
insert into reservation values (123456789,'2024-02-05','2024-02-07',"02 Days");
insert into reservation values (123456780,'2024-02-10','2024-02-11',"01 Days");
insert into reservation values (123456790,'2024-02-15','2024-02-17',"02 Days");
```

Reservation_ID	Start_Date	End_Date	Period
123456780	2024-02-10	2024-02-11	01 Days
123456789	2024-02-05	2024-02-07	02 Days
123456790	2024-02-15	2024-02-17	02 Days
NULL	NULL	NULL	NULL

UPDATE:-



```sql
Create database Hotel_Management;
use Hotel_Management;
UPDATE Customer SET City = 'Mumbai' WHERE Country = 'India';
select * from customer;
```

Customer_ID	First_Name	Middle_Name	Last_Name	E_Mail	Country	Mobile_No	City
1	Piyush	Pradip	Polekar	Piyush@gmail.com	India	1234567890	Mumbai
2	Sara	samir	Parave	Sara@gmail.com	India	1234567899	Mumbai
3	Priya	Sandesh	Gharat	Priya@gmail.com	India	1234567898	Mumbai
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Conclusion

1. **Explain the role of database constraints in enforcing data integrity during DML operations.**

Database constraints play a crucial role in enforcing data integrity during Data Manipulation Language (DML) operations by imposing rules and conditions on the data being inserted, updated, or deleted. Here's how they ensure data integrity:

a) Preventing Invalid Data: Constraints such as primary key, unique, and check constraints ensure that only valid and permissible data is inserted into the database. They restrict the insertion of duplicate values, enforce data format and range constraints, and prevent the insertion of data that does not meet specified criteria.

b) Maintaining Relationships: Foreign key constraints maintain referential integrity by ensuring that relationships between tables are upheld. They prevent the insertion of data that references non-existent records in related tables, thus maintaining consistency and preventing orphaned records.

c) Enforcing Business Rules: Constraints enforce business rules and requirements defined for the database schema. They ensure that data adheres to predefined business logic, preventing data inconsistencies and ensuring that the database accurately represents the real-world domain it models.

d) Automatic Validation: Constraints automatically validate data during DML operations, providing immediate feedback if data violates integrity rules. This proactive validation prevents the insertion of erroneous data, reducing the likelihood of data corruption or inconsistencies.

2. **How do you update multiple columns in a table using a single UPDATE statement?**
To update multiple columns in a table using a single UPDATE statement, you specify the column names and their new values within the UPDATE command's SET clause. Here's a short example:

```
UPDATE table_name
SET column1 = value1, column2 = value2, column3 = value3
WHERE condition;
```

In this syntax:
table_name is the name of the table you want to update.
column1, column2, column3, etc., are the names of the columns you want to update.
value1, value2, value3, etc., are the new values you want to assign to the respective columns.
condition is an optional condition that specifies which rows should be updated. If omitted, all rows in the table will be updated.
This single UPDATE statement will modify the specified columns in the table according to the provided values, subject to the specified condition.