Name	Sahil Shah
UID No.	2021300115
Experiment No.	3

AIM:	To create a database and populate using SQL commands (with constraints): • Data Manipulation Language - Insert, Update, Delete, Select.
Program 1	
PROBLEM STATEMENT:	To create a Hotel Database Management System on SQL and populate it using the SQL commands.
THEORY:	Structured Query Language (SQL), as we all know, is the database language using which we can perform certain operations on the existing database. We can also use this language to create a database. SQL uses certain commands like Create, Drop, Insert, etc. to carry out the required tasks. These SQL commands are mainly categorized into four categories: 1. DDL – Data Definition Language 2. DQL – Data Query Language 3. DML – Data Manipulation Language 4. DCL – Data Control Language Though many resources claim there to be another category of SQL clauses **TCL – Transaction Control Language**. **DML (Data Manipulation Language):** The SQL commands that deal with the manipulation of data present in the database belong to DML or Data Manipulation Language and this includes most of the SQL statements. It is the component of the SQL statement that controls access to data and to the database. Basically, DCL statements are grouped with DML statements. List of DML commands: • **INSERT:** It is used to insert data into a table. • **UPDATE:** It is used to update existing data within a table. • **DELETE:** It is used to delete records from a database table. • **SELECT:** It is used to retrieve data from the database. **1) The SQL INSERT INTO Statement:**

The INSERT INTO statement is used to insert new records in a table.

INSERT INTO Syntax:
It is possible to write the INSERT INTO statement in two ways:

1. Specify both the column names and the values to be inserted:
   INSERT INTO table_name (column1, column2, column3, ...)
   VALUES (value1, value2, value3, ...);

2. If you are adding values for all the columns of the table, you do not need to specify the column names in the SQL query. However, make sure the order of the values is in the same order as the columns in the table. Here, the INSERT INTO syntax would be as follows:
   INSERT INTO table_name
   VALUES (value1, value2, value3, ...);

2) The SQL UPDATE Statement:

The UPDATE statement is used to modify the existing records in a table.

UPDATE Syntax:

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

3) The SQL DELETE Statement:

The DELETE statement is used to delete existing records in a table.

DELETE Syntax:

DELETE FROM table_name WHERE condition;

4) The SQL SELECT Statement:

The SELECT statement is used to select data from a database.

The data returned is stored in a result table, called the result-set.

SELECT Syntax:

SELECT column1, column2, ...
FROM table_name;

Here, column1, column2, ... are the field names of the table you want to select data from. If you want to select all the fields available in the table, use the following syntax:

SELECT * FROM table_name;

CODE:

```
-- Creating the Database
CREATE DATABASE HOTEL;
USE HOTEL;

-- Creating the Table (Hotel)
CREATE TABLE Hotel (
   HotelName varchar(255) NOT NULL,
   ContactNumber int NOT NULL,
   LocationStreetName varchar(255) NOT NULL,
   LocationPincode int NOT NULL,
   LocationCity varchar(255) NOT NULL,
   HotelID int NOT NULL,
   Rating int,
   PRIMARY KEY (HotelID)
);

-- Inserting a Row into Table
INSERT INTO Hotel VALUES('Aparna', 999999999, 'Salisbury Road', 400987, 'Mumbai', 103, 4);
SELECT * FROM Hotel;

-- Creating the Table (Room)
CREATE TABLE Room (
   RoomNumber int,
   RoomAvailability varchar(5) NOT NULL,
   RoomSize varchar(50) NOT NULL,
   RoomType varchar(50) NOT NULL,
   PRIMARY KEY (RoomNumber),
```

```
   HotelID int NOT NULL,
   FOREIGN KEY (HotelID) REFERENCES Hotel(HotelID)
);

-- Inserting Rows into Table
INSERT INTO Room VALUES(237, 'NO', '2 persons', 'A.C', 103);
INSERT INTO Room VALUES(069, 'YES', '2 persons', 'Deluxe', 103);
INSERT INTO Room VALUES(235, 'YES', '1 person', 'A.C', 103);
INSERT INTO Room VALUES(123, 'YES', '4 persons', 'Non-A.C', 103);
INSERT INTO Room VALUES(420, 'YES', '3 persons', 'A.C', 103);
INSERT INTO Room VALUES(666, 'YES', '3 persons', 'A.C', 103);
SELECT * FROM Room;

-- Order (Ascending)
SELECT * FROM Room
ORDER BY RoomType ASC;

-- Update
UPDATE Room
SET RoomAvailability = 'NO', RoomSize = '2 persons'
WHERE RoomNumber = 069;
UPDATE Room
SET RoomAvailability = 'NO', RoomSize = '4 persons'
WHERE RoomNumber = 420;
SELECT * FROM Room;

-- Delete Row
DELETE FROM Room WHERE RoomNumber = 666;
SELECT * FROM Room;

-- Wildcard
SELECT * FROM Room
WHERE RoomType LIKE '%A%';

-- Where
SELECT * FROM Room
WHERE RoomType = 'A.C' AND RoomSize = '2 persons';
```

QUERIES:

Using Create, Use, Insert Into, Select commands:

✓	2	11:47:52	CREATE DATABASE HOTEL	1 row(s) affected	0.032 sec
✓	3	11:47:52	USE HOTEL	0 row(s) affected	0.015 sec
✓	4	11:48:01	CREATE TABLE Hotel (HotelNa...	0 row(s) affected	0.031 sec
✓	5	11:48:08	INSERT INTO Hotel VALUES('Apa...	1 row(s) affected	0.000 sec
✓	6	11:48:44	SELECT * FROM Hotel LIMIT 0, 10...	1 row(s) returned	0.000 sec / 0.000 sec

Original Table

	HotelName	ContactNumber	LocationStreetName	LocationPincode	LocationCity	HotelID	Rating
▶	Aparna	999999999	Salisbury Road	400987	Mumbai	103	4
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Using Create, Insert Into, Select Commands:

✓	9	11:50:13	CREATE TABLE Room (RoomN...	0 row(s) affected	0.031 sec
✓	10	11:50:50	INSERT INTO Room VALUES(237...	1 row(s) affected	0.047 sec
✓	11	11:50:50	INSERT INTO Room VALUES(123...	1 row(s) affected	0.000 sec
✓	12	11:50:50	INSERT INTO Room VALUES(420...	1 row(s) affected	0.000 sec
✓	13	11:50:50	INSERT INTO Room VALUES(069...	1 row(s) affected	0.015 sec
✓	14	11:50:50	INSERT INTO Room VALUES(235...	1 row(s) affected	0.000 sec
✓	15	11:50:50	INSERT INTO Room VALUES(666...	1 row(s) affected	0.000 sec
✓	16	11:50:50	SELECT * FROM Room LIMIT 0, 1...	6 row(s) returned	0.000 sec / 0.000 sec

Original Table

	RoomNumber	RoomAvailability	RoomSize	RoomType	HotelID
▶	69	YES	2 persons	Deluxe	103
	123	YES	4 persons	Non-A.C	103
	235	YES	1 person	A.C	103
	237	NO	2 persons	A.C	103
	420	YES	3 persons	A.C	103
	666	YES	3 persons	A.C	103
*	NULL	NULL	NULL	NULL	NULL

Using Order command:

✓	66	19:27:09	SELECT * FROM Room ORDER BY RoomT...	6 row(s) returned	0.015 sec / 0.000 sec

	RoomNumber	RoomAvailability	RoomSize	RoomType	HotelID
▶	69	YES	2 persons	Deluxe	103
	123	YES	4 persons	Non-A.C	103
	235	YES	1 person	A.C	103
	237	NO	2 persons	A.C	103
	420	YES	3 persons	A.C	103
	666	YES	3 persons	A.C	103
*	NULL	NULL	NULL	NULL	NULL

Using Update, Select command:

```
62  11:57:53  UPDATE Room SET RoomAvailability = 'NO', RoomSi...  1 row(s) affected Rows matched: 1  Changed: 1  War...  0.047 sec
63  11:57:53  UPDATE Room SET RoomAvailability = 'NO', RoomSi...  1 row(s) affected Rows matched: 1  Changed: 1  War...  0.000 sec
64  11:57:53  SELECT * FROM Room LIMIT 0, 1000                   6 row(s) returned                                       0.000 sec / 0.000 sec
```

RoomNumber	RoomAvailability	RoomSize	RoomType	HotelID
69	NO	2 persons	Deluxe	103
123	YES	4 persons	Non-A.C	103
235	YES	1 person	A.C	103
237	NO	2 persons	A.C	103
420	NO	4 persons	A.C	103
666	YES	3 persons	A.C	103
NULL	NULL	NULL	NULL	NULL

Using Delete, Select command:

```
67  19:35:26  DELETE FROM Room WHERE RoomNumb...  1 row(s) affected   0.047 sec
68  19:35:26  SELECT * FROM Room LIMIT 0, 1000   5 row(s) returned   0.000 sec / 0.000 sec
```

RoomNumber	RoomAvailability	RoomSize	RoomType	HotelID
69	NO	2 Persons	Deluxe	103
123	YES	4 persons	Non-A.C	103
235	YES	1 person	A.C	103
237	NO	2 persons	A.C	103
420	NO	4 Persons	A.C	103
NULL	NULL	NULL	NULL	NULL

Using Wildcard command:

```
69  19:39:11  SELECT * FROM Room WHE...  4 row(s) returned   0.000 sec / 0.000 sec
```

RoomNumber	RoomAvailability	RoomSize	RoomType	HotelID
123	YES	4 persons	Non-A.C	103
235	YES	1 person	A.C	103
237	NO	2 persons	A.C	103
420	NO	4 Persons	A.C	103
NULL	NULL	NULL	NULL	NULL

Using Where, And command:

```
70  19:42:58  SELECT * FROM Room WHERE RoomType = 'A.C' AN...  1 row(s) returned   0.000 sec / 0.000 sec
```

RoomNumber	RoomAvailability	RoomSize	RoomType	HotelID
237	NO	2 persons	A.C	103
NULL	NULL	NULL	NULL	NULL

CONCLUSION:

In this experiment, I learned about the various DML commands and using that knowledge, I made changes to my existing database and also added a few like the Order, Wildcard and Where commands.