DBMS LAB
Lab Experiment number 04

Name: Aamir Ansari **Batch:** A **Roll no.** 01

Aim: Experiment to study Data Manipulation Language Commands.

Theory:

The Data Manipulation Language (DML) is used to retrieve, insert and modify database information. These commands will be used by all database users during the routine operation of the database. Let’s take a brief look at the basic DML commands:

INSERT

The INSERT command in SQL is used to add records to an existing table. Returning to the personal_info example from the previous section, let&#39;s imagine that our HR department needs
to add a new employee to their database. They could use a command similar to the one shown
below:

Syntax: insert into table tablename values(values);

Example:

INSERT INTO employee
values(‘bart’, ‘simpson’, 12345, $45000)

These correspond to the table attributes in the order they were defined: first_name, last_name, employee_id, and salary.

SELECT

The SELECT command is the most commonly used command in SQL. It allows database users to retrieve the specific information they desire from an operational database.
Let’s take
a look at a few examples, again using the personal_info table from our employees database.

Syntax: select <attribute list> from <list if tables> where predicate;

Example:

SELECT * FROM employee

Alternatively, users may want to limit the attributes that are retrieved from the database. For example, the Human Resources department may require a list of the last names of all employees in the company. The following SQL command would retrieve only that information.

SELECT last_name FROM employee

Finally, the WHERE clause can be used to limit the records that are retrieved to those that meet specified criteria.

SELECT *FROM employee WHERE salary &gt; 50000

UPDATE

The UPDATE command can be used to modify information contained within a table, either in bulk or individually.

Syntax: update tablename set predicate;

Example:

```
UPDATE employee SET salary = salary * 1.03
```

On the other hand, our new employee Bart Simpson has demonstrated performance above and beyond the call of duty. Management wishes to recognize his stellar accomplishments with a $5,000 raise. The WHERE clause could be used to single out Bart for this raise:

```
UPDATE employee SET salary = salary + $5000
WHERE employee_id = 12345
```

Give 10 % raise in salary who are working on railway project and working for IT department

```
Update employee set salary=salary*1.1 where dno=(select dno from dept where dname=”it”)
and ssn=(select ssn from workson where pno=(select pno from project where pname=”railway”))
```

DELETE

The DELETE command with a WHERE clause can be used to remove his record from the employee table:

Syntax: delete from tablename where predicate;

Example:

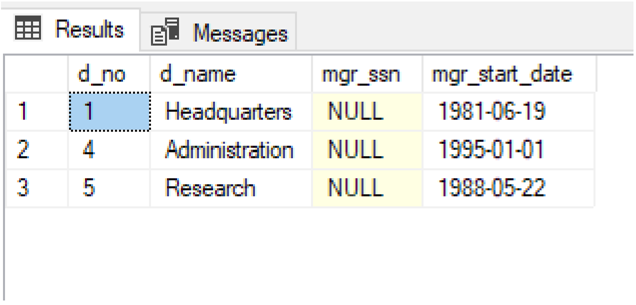
```
DELETE FROM employee WHERE employee_id = 12345
```

Delete employees working for IT department

```
Delete from employee where dno=(select dno from dept where dname=”IT”)
```

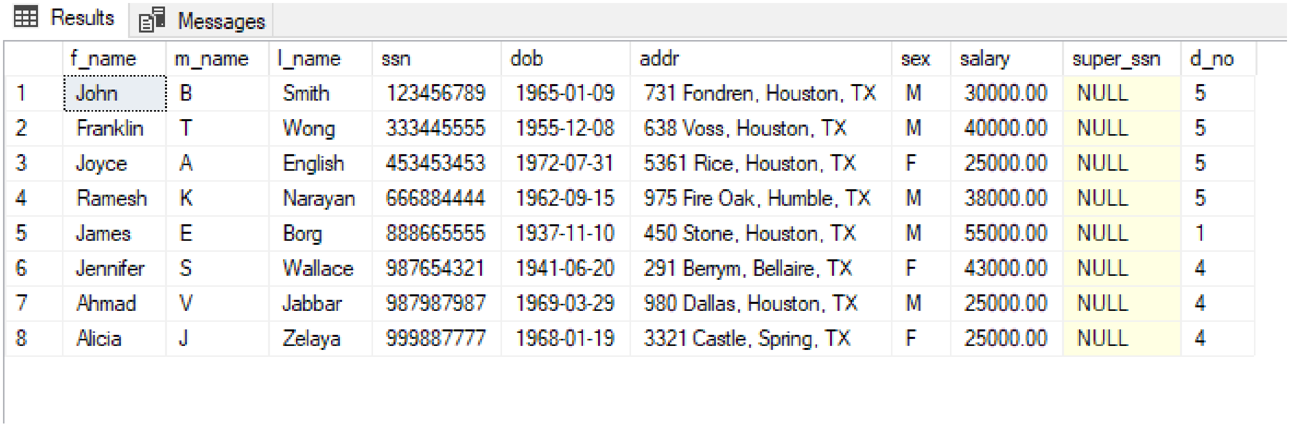
Code:

```
-- Insert value in department table
INSERT INTO Department (d_no, d_name, mgr_ssn, mgr_start_date)
VALUES (
        5,
        'Research',
        NULL,
        '1988-05-22'
),
(
        4,
        'Administration',
        NULL,
        '1995-01-01'
),
(
        1,
        'Headquarters',
        NULL,
        '1981-06-19'
)
SELECT * FROM Department;
```



	d_no	d_name	mgr_ssn	mgr_start_date
1	1	Headquarters	NULL	1981-06-19
2	4	Administration	NULL	1995-01-01
3	5	Research	NULL	1988-05-22

```
-- Insert values in employee table
INSERT INTO Employee (f_name, m_name, l_name, ssn, dob, addr, sex, salary, super_ssn, d_no)
VALUES (
        'John', 'B', 'Smith',
        123456789,
        '1965-01-09',
        '731 Fondren, Houston, TX',
        'M',
        30000,
        NULL,
        5
),
(
	'Franklin', 'T', 'Wong',
	33344555,
	'1955-12-08',
	'638 Voss, Houston, TX',
	'M',
	40000,
	NULL,
	5
),
(
	'Alicia', 'J', 'Zelaya',
	999887777,
	'1968-01-19',
	'3321 Castle, Spring, TX',
	'F',
	25000,
	NULL,
	4
),
(
	'Jennifer', 'S', 'Wallace',
	987654321,
	'1941-06-20',
	'291 Berrym, Bellaire, TX',
	'F',
	43000,
	NULL,
	4
),
(
	'Ramesh', 'K', 'Narayan',
	666884444,
	'1962-09-15',
	'975 Fire Oak, Humble, TX',
	'M',
	38000,
	NULL,
	5
),
(
	'Joyce', 'A', 'English',
	453453453,
	'1972-07-31',
	'5361 Rice, Houston, TX',
	'F',
	25000,
        NULL,
        5
),
(
        'Ahmad', 'V', 'Jabbar',
        987987987,
        '1969-03-29',
        '980 Dallas, Houston, TX',
        'M',
        25000,
        NULL,
        4
),
(
        'James', 'E', 'Borg',
        888665555,
        '1937-11-10',
        '450 Stone, Houston, TX',
        'M',
        55000,
        NULL,
        1
)
SELECT * FROM Employee;
```



	f_name	m_name	l_name	ssn	dob	addr	sex	salary	super_ssn	d_no
1	John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000.00	NULL	5
2	Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000.00	NULL	5
3	Joyce	A	English	453453453	1972-07-31	5361 Rice, Houston, TX	F	25000.00	NULL	5
4	Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000.00	NULL	5
5	James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000.00	NULL	1
6	Jennifer	S	Wallace	987654321	1941-06-20	291 Berrym, Bellaire, TX	F	43000.00	NULL	4
7	Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000.00	NULL	4
8	Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000.00	NULL	4

.

```
-- Insert foreign key super_ssn in Emplyee table
UPDATE Employee SET super_ssn = 333445555
WHERE ssn = 123456789;

UPDATE Employee SET super_ssn = 888665555
WHERE ssn = 333445555;

UPDATE Employee SET super_ssn = 987654321
WHERE ssn = 999887777;
```

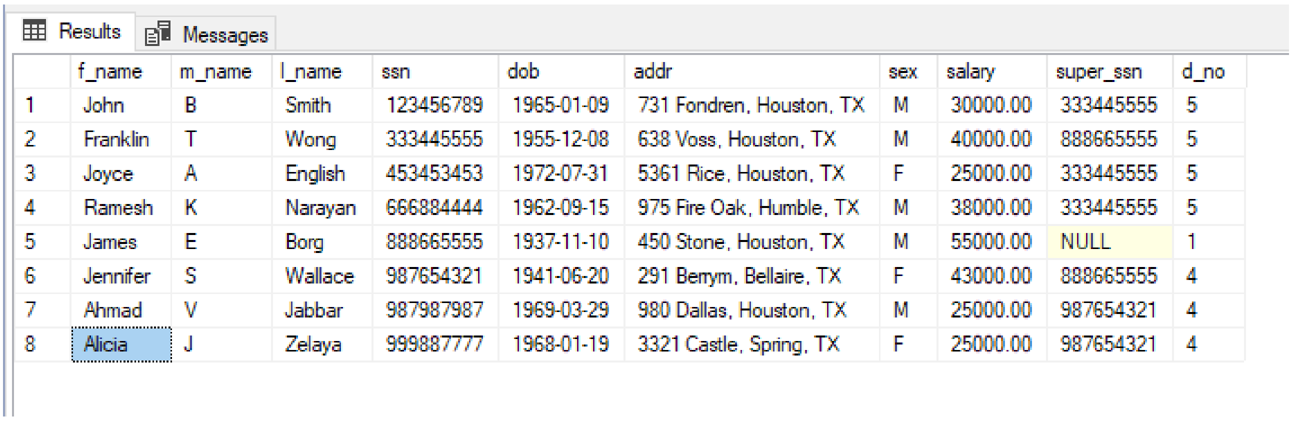
```
UPDATE Employee SET super_ssn = 888665555
WHERE ssn = 987654321;

UPDATE Employee SET super_ssn = 333445555
WHERE ssn = 666884444;

UPDATE Employee SET super_ssn = 333445555
WHERE ssn = 453453453;

UPDATE Employee SET super_ssn = 987654321
WHERE ssn = 987987987;

SELECT * FROM Employee
```



	f_name	m_name	l_name	ssn	dob	addr	sex	salary	super_ssn	d_no
1	John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000.00	333445555	5
2	Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000.00	888665555	5
3	Joyce	A	English	453453453	1972-07-31	5361 Rice, Houston, TX	F	25000.00	333445555	5
4	Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000.00	333445555	5
5	James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000.00	NULL	1
6	Jennifer	S	Wallace	987654321	1941-06-20	291 Berrym, Bellaire, TX	F	43000.00	888665555	4
7	Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000.00	987654321	4
8	Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000.00	987654321	4

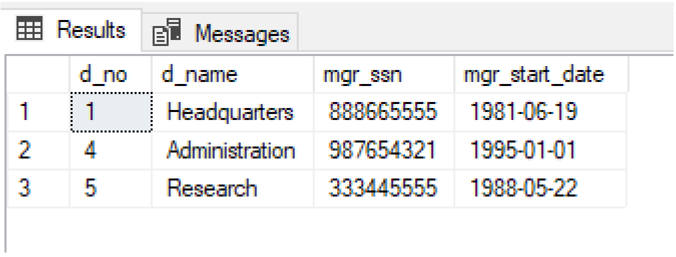
.

```
-- Insert foreign key mgs_ssn in Department Table
UPDATE Department SET mgr_ssn = 333445555
WHERE d_no = 5;

UPDATE Department SET mgr_ssn = 987654321
WHERE d_no = 4;

UPDATE Department SET mgr_ssn = 888665555
WHERE d_no = 1;

SELECT * FROM Department
```

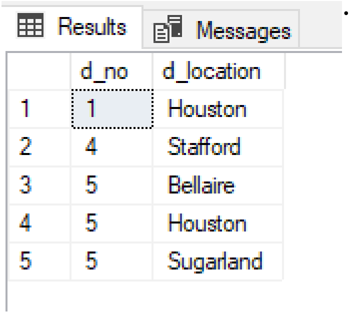


	d_no	d_name	mgr_ssn	mgr_start_date
1	1	Headquarters	888665555	1981-06-19
2	4	Administration	987654321	1995-01-01
3	5	Research	333445555	1988-05-22

.

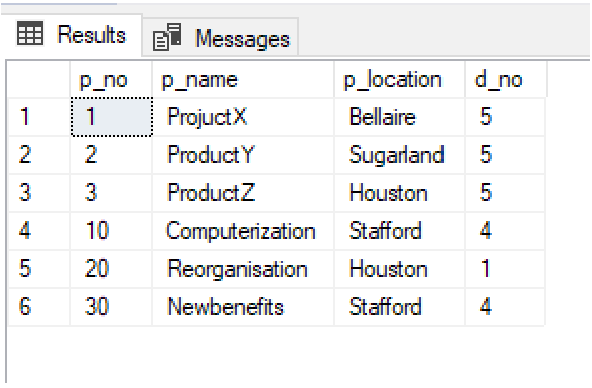
```
-- Insert values in Department_location table
```

```
INSERT INTO Department_location(d_no, d_location)
VALUES (
	1, 'Houston'
),
(
	4, 'Stafford'
),
(
	5, 'Bellaire'
),
(
	5, 'Sugarland'
),
(
	5, 'Houston'
)
SELCT * FROM  Department_location;
```



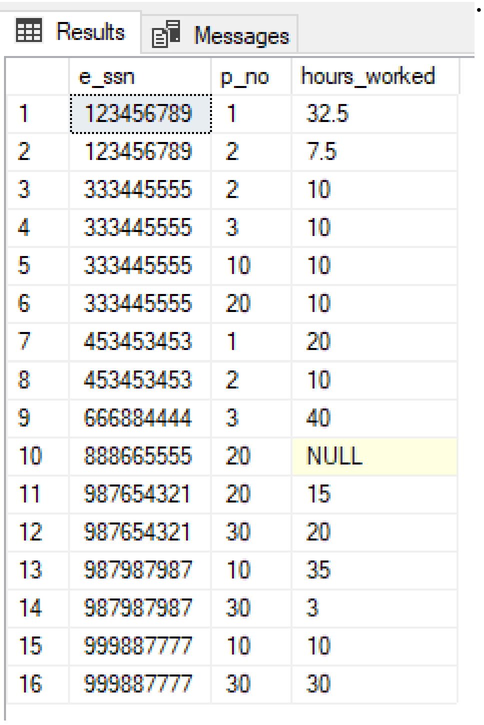
	d_no	d_location
1	1	Houston
2	4	Stafford
3	5	Bellaire
4	5	Houston
5	5	Sugarland

```
-- Insert values in Project table
INSERT INTO Project(p_no, p_name, p_location, d_no)
VALUES (
	1, 'ProjuctX', 'Bellaire', 5
),
(
	2, 'ProductY', 'Sugarland', 5
),
(
	3, 'ProductZ', 'Houston', 5
),
(
	10, 'Computerization', 'Stafford', 4
),
(
	20, 'Reorganisation', 'Houston', 1
),
(
	30, 'Newbenefits', 'Stafford', 4 )
SELECT * FROM Project
```



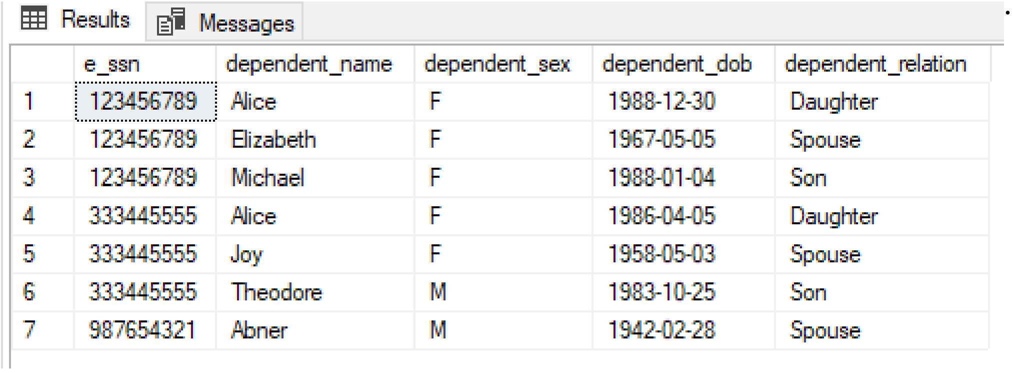
	p_no	p_name	p_location	d_no
1	1	ProjuctX	Bellaire	5
2	2	ProductY	Sugarland	5
3	3	ProductZ	Houston	5
4	10	Computerization	Stafford	4
5	20	Reorganisation	Houston	1
6	30	Newbenefits	Stafford	4

```
-- Insert values in Works_on table
INSERT INTO Works_on (e_ssn, p_no, hours_worked)
VALUES (
        123456789, 1, 32.5
),
(
        123456789, 2, 7.5
),
(
        666884444, 3, 40.0
),
(
        453453453, 1, 20.0
),
(
        453453453, 2, 10.0
),
(
        333445555, 2, 10.0
),
(
        333445555, 3, 10.0
),
(
        333445555, 10, 10.0
),
(
        333445555, 20, 10.0
),
(
        999887777, 30, 30.0
),
(
        999887777, 10, 10.0
),
(
        987987987, 10, 35.0
),
(
        987987987, 30, 3.0
),
(
        987654321, 30, 20.0
),
(
        987654321, 20, 15.0
),
(
        888665555, 20, NULL
)
SELECT * FROM Works_on
```



	e_ssn	p_no	hours_worked
1	123456789	1	32.5
2	123456789	2	7.5
3	333445555	2	10
4	333445555	3	10
5	333445555	10	10
6	333445555	20	10
7	453453453	1	20
8	453453453	2	10
9	666884444	3	40
10	888665555	20	NULL
11	987654321	20	15
12	987654321	30	20
13	987987987	10	35
14	987987987	30	3
15	999887777	10	10
16	999887777	30	30

```
--Insert values into Dependant table
INSERT INTO Dependant(e_ssn, dependent_name, dependent_sex, dependent_dob,
dependent_relation)
VALUES (
        333445555, 'Alice',
        'F',
        '1986-04-05',
        'Daughter'
),
(
        333445555, 'Theodore',
        'M',
        '1983-10-25',
        'Son'
),
(
        333445555, 'Joy',
        'F',
        '1958-05-03',
        'Spouse'
),
(
        987654321, 'Abner',
        'M',
        '1942-02-28',
        'Spouse'
),
(
        123456789, 'Michael',
        'F',
        '1988-01-04',
        'Son'
),
(
        123456789, 'Alice',
        'F',
        '1988-12-30',
        'Daughter'
),
(
        123456789, 'Elizabeth',
        'F',
        '1967-05-05',
        'Spouse'
)
```



	e_ssn	dependent_name	dependent_sex	dependent_dob	dependent_relation
1	123456789	Alice	F	1988-12-30	Daughter
2	123456789	Elizabeth	F	1967-05-05	Spouse
3	123456789	Michael	F	1988-01-04	Son
4	333445555	Alice	F	1986-04-05	Daughter
5	333445555	Joy	F	1958-05-03	Spouse
6	333445555	Theodore	M	1983-10-25	Son
7	987654321	Abner	M	1942-02-28	Spouse

Conclusion: Thus we have implemented different DML commands successfully.