

Rupesh Dharme

31124

TE 01

Assignment 03

Page No. _____
Date: / /

Title : SQL Queries all types of join, sub Query and view.

Aim : Write atleast 10 SQL queries for suitable database application using SQL DML statements

Instructor will design the queries which demonstrate use of concepts like join, subquery and queue

Objective: To understand

: Types of joins, subquery and it's types, complex views.

S/W H/W : MySQL, Windows 10 (64 bit)

Reference : Koith H database system concepts McGraw Hill

Theory

SQL Join

The ability of relational join operator is an important feature of relational systems. a join makes it possible to select data from more than one table by means of a single statement.

Teacher's Signature _____

Types: Inner, Outer (Left or Right) Full, cross

Inner join: also known as equi join. Statement compares two columns with equivalence operator =

selects rows that have values in common in columns specified in on clause

Outer join: similar to inner join but more flexible. selects all rows from table on left/right/full irrespective of other table values (NULL wherever missing)

Cross join: known as cartesian Product. combines every row from left with every row in right

Self join: known to join to itself, as against 2 separate tables.

Conclusion: Successfully implemented all joins on the tables.

```
-- Rupesh Dharne
-- 31124
-- TE 01
-- K1
-- Assignment 03

create database assignment03;
use assignment03;

create table departments(
    dept_id int auto_increment not null,
    dept_name varchar(15) not null,
    primary key(dept_id)
);

create table professors(
    prof_id int not null auto_increment,
    prof_fname varchar(20) not null,
    prof_lname varchar(20),
    dept_id int,
    designation varchar(20),
    salary int,
    doj date,
    email varchar(255),
    phone varchar(10),
    city varchar(15),
    primary key(prof_id),
    foreign key(dept_id) references departments(dept_id) on delete cascade
);

create table works(
    prof_id int not null,
    dept_id int not null,
    duration_hr int,
    primary key(prof_id, dept_id),
    foreign key(prof_id) references professors(prof_id) on delete cascade,
    foreign key(dept_id) references departments(dept_id) on delete cascade
);

create table shifts(
    prof_id int not null,
    shift varchar(10) not null,
    working_hours int not null,
    primary key(prof_id),
    foreign key(prof_id) references professors(prof_id) on delete cascade
);

show tables;
desc departments;
desc professors;
desc shifts;
desc works;

insert into departments (dept_name)
values ('Computer'),
       ('IT'),
       ('EnTC');

select * from departments;
```

```

insert into professors (prof_fname, prof_lname, dept_id, designation, salary, doj
, email, phone, city)
values ('Prof1', 'PLast1', 1, 'Professor', 100000, '2020-06-
19', 'email@email.com', '9999888877', 'Pune'),
      ('Prof2', 'PLast2', 2, 'Assistant Professor', 100000, '2015-01-
01', 'email1@email.com', '9999888877', 'Mumbai'),
      ('Prof3', 'PLast3', 3, 'Professor', 100000, '2008-02-
07', 'email2@email.com', '8799888877', 'Mumbai'),
      ('Prof4', 'PLast4', 1, 'Assistant Professor', 100000, '2020-06-
30', 'email3@email.com', '9999888877', 'Pune'),
      ('Prof5', 'PLast5', 2, 'Professor', 100000, '2015-01-
01', 'email4@email.com', '9999888877', 'Mumbai'),
      ('Prof6', 'PLast6', 3, 'Assistant Professor', 100000, '2019-07-
07', 'email5@email.com', '9998788877', 'Mumbai'),
      ('Prof7', 'PLast7', 1, 'Professor', 100000, '2019-02-
02', 'email6@email.com', '9993338877', 'Pune');

select * from professors;

insert into shifts
values (1, 'Morning', 7),
      (2, 'Afternoon', 7),
      (3, 'Evening', 4),
      (4, 'Morning', 7),
      (5, 'Afternoon', 7),
      (6, 'Evening', 4),
      (7, 'Morning', 7);
select * from shifts;

insert into works
values
      (1, 1, 5),
      (2, 2, 7),
      (3, 3, 2),
      (4, 1, 6),
      (5, 2, 4),
      (6, 3, 3),
      (7, 1, 6);

select * from works;

-
- 1. Find the professor details and department details using NATURAL JOIN.
select * from professors
natural join departments;

-- 2. Find the prof_id, prof_name and shift. (INNER JOIN)
select professors.prof_id, prof_fname, shift
from professors
inner join shifts
on professors.prof_id = shifts.prof_id;

-
- 3. List all the department details and the corresponding names of professors in
the same
-- department.(left outer join)
select departments.dept_id, dept_name, prof_fname
from departments
left join professors

```

```

on departments.dept_id = professors.dept_id;

-
- 4. List all the professors and the corresponding names of department.(right outer join)
select prof_fname, dept_name
from professors
right join departments
on professors.dept_id = departments.dept_id;

-
- 5. Display professor name, dept_name, shift, salary where prof_id = 101;(multiple join)
select prof_fname, dept_name, shift, salary
from professors
inner join departments
on professors.dept_id = departments.dept_id
inner join shifts
on professors.prof_id = shifts.prof_id
where professors.prof_id = 5;

-
- 6. list the total number of professor in each department.(count and any join, group by)
select dept_name, count(*)
from professors
inner join departments
on professors.dept_id = departments.dept_id
group by dept_name;

-
- 7. List the prof_id associated department and the dept_name having name 'computer';(subquery)
select prof_fname, prof_id, professors.dept_id, dept_name
from professors
inner join departments
on professors.dept_id = departments.dept_id
where dept_name in (
    select dept_name from departments
);

-
- Find the names of all departments where the professors joined in year 2015 (or date of joining
-- is 1-1-2015)
select dept_name
from professors
inner join departments
on professors.dept_id = departments.dept_id
where professors.doj = '2015-01-01';

```

```
mysql> show databases;
```

Database
assignment02
information_schema
mysql
performance_schema
sys

```
5 rows in set (0.03 sec)
```

```
mysql> create database assignment03;  
Query OK, 1 row affected (0.04 sec)
```

```
mysql> use assignment03;  
Database changed
```

```
mysql>
```

```
mysql> create table departments(  
->     dept_id int auto_increment not null,  
->     dept_name varchar(15) not null,  
->     primary key(dept_id)  
-> );
```

```
Query OK, 0 rows affected (0.08 sec)
```

```
mysql>
```

```
mysql> create table professors(  
->     prof_id int not null auto_increment,  
->     prof_fname varchar(20) not null,  
->     prof_lname varchar(20),  
->     dept_id int,  
->     designation varchar(20),  
->     salary int,  
->     doj date,  
->     email varchar(255),  
->     phone varchar(10),  
->     city varchar(15),  
->     primary key(prof_id),  
->     foreign key(dept_id) references departments(dept_id) on delete cascade  
-> );
```

```
Query OK, 0 rows affected (0.07 sec)
```

```
mysql>
```

```
mysql> create table works(  
->     emp_id int not null auto_increment,  
->     prof_id int not null,  
->     dept_id int,  
->     emp_fname varchar(20) not null,  
->     emp_lname varchar(20),  
->     emp_salary int,  
->     emp_doj date,  
->     emp_email varchar(255),  
->     emp_phone varchar(10),  
->     emp_city varchar(15),  
->     primary key(emp_id),  
->     foreign key(prof_id) references professors(prof_id) on delete cascade,  
->     foreign key(dept_id) references departments(dept_id) on delete cascade  
-> );
```

```
Query OK, 0 rows affected (0.08 sec)
```

```
mysql>
```

```
mysql>
```

```
mysql>
```

```
mysql>
```

```
mysql>
```

```
mysql>
```



```
mysql>
mysql> create table works(
->   prof_id int not null,
->   dept_id int not null,
->   duration_hr int,
->   primary key(prof_id, dept_id),
->   foreign key(prof_id) references professors(prof_id) on delete cascade,
->   foreign key(dept_id) references departments(dept_id) on delete cascade
-> );
Query OK, 0 rows affected (0.08 sec)
```

```
mysql>
mysql> create table shifts(
->   prof_id int not null,
->   shift varchar(10) not null,
->   working_hours int not null,
->   primary key(prof_id),
->   foreign key(prof_id) references professors(prof_id) on delete cascade
-> );
Query OK, 0 rows affected (0.06 sec)
```

```
mysql>
mysql> show tables;
+-----+
| Tables_in_assignment03 |
+-----+
| departments             |
| professors              |
| shifts                  |
| works                   |
+-----+
```

4 rows in set (0.04 sec)

```
mysql> desc departments;
+-----+-----+-----+-----+-----+-----+
| Field      | Type          | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| dept_id    | int           | NO   | PRI | NULL    | auto_increment |
| dept_name  | varchar(15)   | NO   |     | NULL    |                |
+-----+-----+-----+-----+-----+-----+
```

2 rows in set (0.03 sec)

```
mysql> desc professors;
+-----+-----+-----+-----+-----+-----+
```

```
MySQL 8.0 Command Line Client - Unicode
mysql> desc departments;
+-----+-----+-----+-----+-----+-----+
| Field      | Type      | Null | Key | Default | Extra      |
+-----+-----+-----+-----+-----+-----+
| dept_id    | int       | NO   | PRI | NULL    | auto_increment |
| dept_name  | varchar(15) | NO   |     | NULL    |              |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.03 sec)

mysql> desc professors;
+-----+-----+-----+-----+-----+-----+
| Field      | Type      | Null | Key | Default | Extra      |
+-----+-----+-----+-----+-----+-----+
| prof_id    | int       | NO   | PRI | NULL    | auto_increment |
| prof_fname | varchar(20) | NO   |     | NULL    |              |
| prof_lname | varchar(20) | YES  |     | NULL    |              |
| dept_id    | int       | YES  | MUL | NULL    |              |
| designation | varchar(20) | YES  |     | NULL    |              |
| salary     | int       | YES  |     | NULL    |              |
| doj        | date      | YES  |     | NULL    |              |
| email      | varchar(255) | YES  |     | NULL    |              |
| phone      | varchar(10) | YES  |     | NULL    |              |
| city       | varchar(15) | YES  |     | NULL    |              |
+-----+-----+-----+-----+-----+-----+
10 rows in set (0.01 sec)

mysql> desc shifts;
+-----+-----+-----+-----+-----+-----+
| Field      | Type      | Null | Key | Default | Extra      |
+-----+-----+-----+-----+-----+-----+
| prof_id    | int       | NO   | PRI | NULL    |              |
| shift      | varchar(10) | NO   |     | NULL    |              |
| working_hours | int       | NO   |     | NULL    |              |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> desc works;
+-----+-----+-----+-----+-----+-----+
| Field      | Type      | Null | Key | Default | Extra      |
+-----+-----+-----+-----+-----+-----+
| prof_id    | int       | NO   | PRI | NULL    |              |
| dept_id    | int       | NO   | PRI | NULL    |              |
| duration_hr | int       | YES  |     | NULL    |              |
+-----+-----+-----+-----+-----+-----+
```



```
MySQL 8.0 Command Line Client - Unicode
mysql>
mysql> insert into departments (dept_name)
-> values ('Computer'),
->        ('IT'),
->        ('EnTC');
Query OK, 3 rows affected (0.03 sec)
Records: 3  Duplicates: 0  Warnings: 0

mysql>
mysql> select * from departments;
+-----+-----+
| dept_id | dept_name |
+-----+-----+
|      1 | Computer  |
|      2 | IT        |
|      3 | EnTC      |
+-----+-----+
3 rows in set (0.00 sec)

mysql>
mysql> insert into professors (prof_fname, prof_lname, dept_id, designation, salary, doj, email, phone, city)
-> values ('Prof1', 'PLast1', 1, 'Professor', 100000, '2020-06-19', 'email@email.com', '9999888877', 'Pune'),
->        ('Prof2', 'PLast2', 2, 'Assistant Professor', 100000, '2015-01-01', 'email1@email.com', '9999887877', 'Mumbai'),
->        ('Prof3', 'PLast3', 3, 'Professor', 100000, '2008-02-07', 'email2@email.com', '8799888877', 'Mumbai'),
->        ('Prof4', 'PLast4', 1, 'Assistant Professor', 100000, '2020-06-30', 'email3@email.com', '9999888877', 'Pune'),
->        ('Prof5', 'PLast5', 2, 'Professor', 100000, '2015-01-01', 'email4@email.com', '9999887877', 'Mumbai'),
->        ('Prof6', 'PLast6', 3, 'Assistant Professor', 100000, '2019-07-07', 'email5@email.com', '9998788877', 'Mumbai'),
->        ('Prof7', 'PLast7', 1, 'Professor', 100000, '2019-02-02', 'email6@email.com', '9993338877', 'Pune');
Query OK, 7 rows affected (0.02 sec)
Records: 7  Duplicates: 0  Warnings: 0

mysql>
mysql> select * from professors;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| prof_id | prof_fname | prof_lname | dept_id | designation          | salary | doj       | email           | phone       | city       |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|      1 | Prof1      | PLast1     |      1 | Professor            | 100000 | 2020-06-19 | email@email.com | 9999888877 | Pune      |
|      2 | Prof2      | PLast2     |      2 | Assistant Professor  | 100000 | 2015-01-01 | email1@email.com | 9999887877 | Mumbai    |
|      3 | Prof3      | PLast3     |      3 | Professor            | 100000 | 2008-02-07 | email2@email.com | 8799888877 | Mumbai    |
|      4 | Prof4      | PLast4     |      1 | Assistant Professor  | 100000 | 2020-06-30 | email3@email.com | 9999888877 | Pune      |
|      5 | Prof5      | PLast5     |      2 | Professor            | 100000 | 2015-01-01 | email4@email.com | 9999887877 | Mumbai    |
|      6 | Prof6      | PLast6     |      3 | Assistant Professor  | 100000 | 2019-07-07 | email5@email.com | 9998788877 | Mumbai    |
|      7 | Prof7      | PLast7     |      1 | Professor            | 100000 | 2019-02-02 | email6@email.com | 9993338877 | Pune      |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+

```

```
mysql>
mysql> insert into shifts
-> values (1,'Morning', 7),
->         (2,'Afternoon', 7),
->         (3,'Evening', 4),
->         (4,'Morning', 7),
->         (5,'Afternoon', 7),
->         (6,'Evening', 4),
->         (7,'Morning', 7);
Query OK, 7 rows affected (0.02 sec)
Records: 7  Duplicates: 0  Warnings: 0
```

```
mysql> select * from shifts;
```

prof_id	shift	working_hours
1	Morning	7
2	Afternoon	7
3	Evening	4
4	Morning	7
5	Afternoon	7
6	Evening	4
7	Morning	7

```
7 rows in set (0.00 sec)
```

```
mysql>
mysql> insert into works
-> values
->     (1, 1, 5),
->     (2, 2, 7),
->     (3, 3, 2),
->     (4, 1, 6),
->     (5, 2, 4),
->     (6, 3, 3),
->     (7, 1, 6);
Query OK, 7 rows affected (0.02 sec)
Records: 7  Duplicates: 0  Warnings: 0
```

```
mysql>
mysql> select * from works;
```

prof_id	dept_id	duration_hr
---------	---------	-------------

```
mysql>
mysql> -- 1. Find the professor details and department details using NATURAL JOIN.
mysql> select * from professors
    -> natural join departments;
```

dept_id	prof_id	prof_fname	prof_lname	designation	salary	doj	email	phone	city	dept_name
1	1	Prof1	PLast1	Professor	100000	2020-06-19	email@email.com	9999888877	Pune	Computer
1	4	Prof4	PLast4	Assistant Professor	100000	2020-06-30	email3@email.com	9999888877	Pune	Computer
1	7	Prof7	PLast7	Professor	100000	2019-02-02	email6@email.com	9993338877	Pune	Computer
2	2	Prof2	PLast2	Assistant Professor	100000	2015-01-01	email1@email.com	9999887877	Mumbai	IT
2	5	Prof5	PLast5	Professor	100000	2015-01-01	email4@email.com	9999887877	Mumbai	IT
3	3	Prof3	PLast3	Professor	100000	2008-02-07	email2@email.com	8799888877	Mumbai	Electrical
3	6	Prof6	PLast6	Assistant Professor	100000	2019-07-07	email5@email.com	9998788877	Mumbai	Electrical

7 rows in set (0.00 sec)

```
mysql>
mysql> -- 2. Find the prof_id, prof_name and shift. (INNER JOIN)
mysql> select professors.prof_id, prof_fname, shift
    -> from professors
    -> inner join shifts
    -> on professors.prof_id = shifts.prof_id;
```

prof_id	prof_fname	shift
1	Prof1	Morning
2	Prof2	Afternoon
3	Prof3	Evening
4	Prof4	Morning
5	Prof5	Afternoon
6	Prof6	Evening

7 rows in set (0.00 sec)

```
mysql>
mysql> -- 3. List all the department details and the corresponding names of professors in the same
```

```
mysql>
mysql> -- 3. List all the department details and the corresponding names of professors in the same
mysql> -- department.(left outer join)
mysql> select departments.dept_id, dept_name, prof_fname
-> from departments
-> left join professors
-> on departments.dept_id = professors.dept_id;
```

dept_id	dept_name	prof_fname
1	Computer	Prof1
1	Computer	Prof4
1	Computer	Prof7
2	IT	Prof2
2	IT	Prof5
3	EnTC	Prof3
3	EnTC	Prof6

7 rows in set (0.00 sec)

```
mysql>
mysql> -- 4. List all the professors and the corresponding names of department.(right outer join)
mysql> select prof_fname, dept_name
-> from professors
-> right join departments
-> on professors.dept_id = departments.dept_id;
```

prof_fname	dept_name
Prof1	Computer
Prof4	Computer
Prof7	Computer
Prof2	IT
Prof5	IT
Prof3	EnTC
Prof6	EnTC

7 rows in set (0.00 sec)

```
mysql>
mysql> -- 5. Display professor name, dept_name, shift, salary where prof_id = 101;(multitable join)
mysql> select prof_fname, dept_name, shift, salary
-> from professors
-> inner join departments
```



```
mysql>
mysql> -- 5. Display professor name, dept_name, shift, salary where prof_id = 101;(multitable join)
```

```
mysql> select prof_fname, dept_name, shift, salary
-> from professors
-> inner join departments
-> on professors.dept_id = departments.dept_id
-> inner join shifts
-> on professors.prof_id = shifts.prof_id
-> where professors.prof_id = 5;
```

prof_fname	dept_name	shift	salary
Prof5	IT	Afternoon	100000

```
1 row in set (0.03 sec)
```

```
mysql>
mysql> -- 6. list the total number of professor in each department.(count and any join,groupby)
```

```
mysql> select dept_name, count(*)
-> from professors
-> inner join departments
-> on professors.dept_id = departments.dept_id
-> group by departments.dept_name;
```

dept_name	count(*)
Computer	3
IT	2
EnTC	2

```
3 rows in set (0.01 sec)
```

```
mysql>
mysql> -- 7. List the prof_id associated department and the dept_name having name 'computer';(subquery)
```

```
mysql> select prof_fname, prof_id, professors.dept_id, dept_name
-> from professors
-> inner join departments
-> on professors.dept_id = departments.dept_id
-> where dept_name in (
-> select dept_name from departments
-> );
```

prof_fname	prof_id	dept_id	dept_name
Prof1	1	1	Computer
Prof4	4	1	Computer
Prof7	7	1	Computer
Prof2	2	2	IT
Prof5	5	2	IT
Prof3	3	3	EnTC

```
MySQL 8.0 Command Line Client - Unicode
mysql> -- 7. List the prof_id associated department and the dept_name having name 'computer';(subquery)
mysql> select prof_fname, prof_id, professors.dept_id, dept_name
-> from professors
-> inner join departments
-> on professors.dept_id = departments.dept_id
-> where dept_name in (
->   select dept_name from departments
-> );
```

prof_fname	prof_id	dept_id	dept_name
Prof1	1	1	Computer
Prof4	4	1	Computer
Prof7	7	1	Computer
Prof2	2	2	IT
Prof5	5	2	IT
Prof3	3	3	EnTC
Prof6	6	3	EnTC

```
7 rows in set (0.01 sec)

mysql>
mysql> -- Find the names of all departments where the professors joined in year 2015 (or date of joining
mysql> -- is 1-1-2015)
mysql> select dept_name
-> from professors
-> inner join departments
-> on professors.dept_id = departments.dept_id
-> where professors.doj = '2015-01-01';
```

dept_name
IT
IT

```
2 rows in set (0.01 sec)

mysql> _
```