

Name Ritik Shah

Roll No. 2005533

Batch CSE-06

Branch CSE

Subject DBMS

Assignment - 6

Ques 1 Display employee John Smith's department

select employee.fname, employee.lname, dept.deptname from
employee INNER JOIN Dept ON employee.deptid = dept.deptid
where employee.fname = (select fname from employee
where fname = 'John');

```
SQL> select employee.fname, employee.lname, dept.deptname from employee INNER JOIN Dept ON  
2 employee.deptid = dept.deptid where employee.fname= (select fname from employee where fname='John');
```

FNAME	LNAME	DEPTNAME
John	Smith	Sales

Ques 2 Find name of the supervisor for employee number 433.

select supervisor from employee where employeeid =
(select employeeid from employee where employeeid = 433);

```
SQL> select supervisor from employee where employeeid = (select employeeid from employee where employeeid = 433);
```

SUPERVISOR

Mark

Ques3 Who has same qualification as Larry.

select employee.fname, employee.lname, qualification. qualdesc from employee INNER JOIN qualification ON qualification.QualID = Employee.QualID where qualification. qual.ID = (select employee.qualID from employee where employee.fname = 'Larry') AND fname <> 'Larry';

```
SQL> select employee.fname, employee.lname, qualification.qualdesc from employee INNER JOIN
  2 qualification ON qualification.QualID = Employee.QualID where qualification.qualid =
  3 (select qualid
  4 from employee where fname='Larry') AND fname != 'Larry';
```

FNAME	LNAME	QUALDESC
Nikhil	Gupta	Masters

Ques4 Which department has more employees than department 20.

select deptid, count(*) from employee group by deptid having count(*) > (select count(*) from employee where deptid = 10);

```
SQL> select deptid, count(*) from employee group by deptid having count(*)>
  2 (select count(*) from employee where deptid=10);
```

DEPTID	COUNT(*)
20	2

Ques5 Which employees are working in the company longer than Larry.

select fname, hiredate from employee where hiredate >
(select hiredate from employee where fname = 'Larry');

```
SQL> select fname, hiredate from employee where hiredate >  
2 (select hiredate from employee where fname = 'Larry');
```

FNAME	HIREDATE
Nikhil	12-FEB-91
Derek	15-MAR-95
Alex	05-MAY-97

Ques6 Find all employees in sales department by using a nested query.

select concat(employee.fname, concat(',', employee.lname))
as "Employee Name", dept.deptname from Employee
INNER JOIN Dept ON Employee.deptID = dept.deptID
where dept.deptName = "Sales";

```
SQL> select concat(employee.fname, concat(',', employee.lname)) as "Employee Name", dept.deptname  
2 from Employee INNER JOIN Dept ON Employee.deptid = dept.deptid where dept.deptname = 'Sales';
```

Employee Name	DEPTNAME
John,Smith	Sales

Ques 7 Create a new table, EMP30, & populate it with employees in department 30, using an existing table & a subquery. Use employeeid, lname, fname, Hiredate, Salary

create table emp30 as select * from employee where deptid=30;

select * from emp30;

```
SQL> select * from emp30;
```

EMPLOYEEID	LNAME	FNAME	POSITIONID	SUPERVISOR	HIREDATE	SALARY	COMMISSION	DEPTID	QUALID
111	Smith	John	1	Scott	15-APR-60	320650	35000	30	1
246	Houston	Larry	2	Emma	19-MAY-67	10000	10000	40	2

Ques 8 Add more rows to EMP30 table with employee in department 40

insert into emp30(employeeid, lname, fname, positionid, supervisor, hiredate, commission, deptid, qualid) select employeeid, lname, fname, positionid, supervisor, hiredate, commission, deptid, qualid from employee where deptid=40;

```
SQL> insert into emp30 (employeeid, lname, fname, positionid, supervisor, hiredate, commission, deptid, qualid)
2 select employeeid, lname, fname, positionid, supervisor, hiredate, commission, deptid, qualid from employee where deptid=40;
1 row created.
```


Ques 9 Update salary of newly transferred employee from employee table to EMP30 table with MERGE statement, & Insert employees who are not in EMP30 table.

Merge Into emp30 o USING (Select EmployeeID, Lname, Fname, Hiredate, Salary from Employee) e ON

(o. EmployeeID = e. EmployeeID) When Matched Then

Update Set o. Salary = e. Salary When NOT MATCHED

Then INSERT (o. EmployeeID, o. Lname, o. Fname, o. Hiredate, o. Salary) values (e. EmployeeID, e. Lname, e. Fname, e. Hiredate, e. Salary);

```
SQL> MERGE INTO emp30 o USING (SELECT EmployeeId, Lname, Fname, HireDate, Salary, deptid FROM employee)
  2 e ON (o.EmployeeId = e.EmployeeId) WHEN MATCHED THEN UPDATE
  3 SET o.Salary = e.Salary WHEN NOT MATCHED THEN INSERT (o.EmployeeId, o.Lname, o.Fname, o.HireDate, o.Salary, o.deptid)
  4 VALUES (e.EmployeeId, e.Lname, e.Fname, e.HireDate, e.Salary, e.deptid);
```

6 rows merged.

```
SQL> select * from emp30;
```

EMPLOYEEID	LNAME	FNAME	POSITIONID	SUPERVISOR	HIREDATE	SALARY	COMMISSION	DEPTID	QUALID
111	Smith	John	1	Scott	15-APR-60	320650	35000	30	1
433	McCall	Alex			05-MAY-97	66500		20	
543	Dave	Derek			15-MAR-95	80000		20	
123	Gupta	Nikhil			12-FEB-91	75000		10	
246	Houstan	Larry	2	Emma	19-MAY-67	150000	10000	40	2
246	Houstan	Larry	2	Emma	19-MAY-67	150000	10000	40	2

6 rows selected.

Ques 10 Find employees with minimum salary in their own department with the use of correlated subquery.

select lname, fname, salary, deptID from employee o
 where salary = (select Min(salary) from employee
 where DeptID = o.DeptID group by deptID);

```
SQL> SELECT Lname, Fname, Salary, DeptId FROM employee o WHERE Salary = (SELECT MIN(Salary)
2 FROM employee WHERE DeptId = o.DeptId GROUP BY DeptId);
```

LNAME	FNAME	SALARY	DEPTID
Smith	John	320650	30
Houstan	Larry	150000	40
Gupta	Nikhil	75000	10
McCall	Alex	66500	20

Ques 11 Use multiple level subquery to display dependent information for employee who belongs to finance department

select * from dependent where employeeid = (select
 employeeid from dept where deptname = 'Finance');

```
SQL> select * from dependent where employeeid =
2 (select employeeid from dept where deptname='Finance');
```

EMPLOYEEID	DEPENDENTID	DEPDOB	RELATION
123	2	23-AUG-97	Son

Ques 12 Use set operator & subquery to find employees who do not have any dependents.

select lname, fname from employee where employeeid =
ANY(select employeeid from employee minus select
distinct employeeid from dependent);

```
SQL> (select employeeid from employee minus select distinct employeeid from dependent);
```

```
EMPLOYEEID
```

```
-----  
246
```

```
433
```

Ques 13 Write subquery that finds average salary by each department. Check to find if employee 543's salary satisfies =ANY, <ANY, >ANY, <ALL, or >ALL condition against those departmental average salaries.

=ANY
select lname, fname from employee where employeeid =
543 & salary = ANY (select avg(salary) from employee
group by deptid);

```
SQL> select lname, fname from employee where employeeid=543 and salary =ANY  
2 (select avg(salary) from employee group by deptid);
```

```
no rows selected
```

> ANY

select lname, fname from employee where employeeid=543 and salary > ANY (select avg(salary) from employee group by deptid);

```
SQL> select lname, fname from employee where employeeid=543 and salary >ANY
      2 (select avg(salary) from employee group by deptid);
```

LNAME	FNAME
Dave	Derek

ANY

select lname, fname from employee where employeeid=543 and salary < ANY (select AVG(salary) from employee group by deptid);

```
SQL> select lname, fname from employee where employeeid=543 and salary <ANY
      2 (select avg(salary) from employee group by deptid);
```

LNAME	FNAME
Dave	Derek

> ALL

select lname, fname from employee where employeeid = 543 and salary > ALL (select avg(salary) from employee group by deptid);

```
SQL> select lname, fname from employee where employeeid=543 and salary >ALL
      2 (select avg(salary) from employee group by deptid);
no rows selected
```

< ALL

select lname, fname from employee where employeeid = 543 and salary < ALL (select avg(salary) from employee group by deptid);

```
SQL> select lname, fname from employee where employeeid=543 and salary <ALL
      2 (select avg(salary) from employee group by deptid);
no rows selected
```

Ques 14 Display student Brian Lee faculty advisor's name & phone number.

select faculty.name, faculty.phone from faculty where
faculty.facultyid=(select facultyid from student where
lname='Lee' and fname='Brian');

```
SQL> select faculty.name, faculty.phone from faculty where faculty.facultyid =  
2 (select facultyid from student where lname='Lee' and fname='Brian');  
  
no rows selected
```

Ques 15 Find rooms with bottom 2 capacities. Do not include office rooms.

select * from location where roomtype != 'O' and
capacity <= ANY (select capacity from (select
capacity from location where roomtype != 'O' group
by capacity order by capacity) where rownum <= 2);

```
SQL> select * from location where roomtype != 'O' and capacity <= ANY  
2 (select capacity from (select capacity from location where roomtype != 'O' group by capacity order by capacity)  
3 where rownum <= 2);
```

ROOMID	BUILDING	ROOMNO	CAPACITY	ROOMTYPE
2	WingB	602	3	C
4	WingC	802	2	VCO
5	WingD	401	3	ADT

Ques 16 Find Spring 2022 course sections with top three maximum count numbers

select * from (select * from crssection where termID = 'SP22' order by maxcount desc) where rownum <= 3;

```
SQL> select * from (select * from crssection where termid='SP22'
2 order by maxcount desc) where rownum<=3;
```

CSID	COURSEID	SECTION	TERMID	FACULTYID	DAY
1101	CS1001	01	SP22	111	Mon

Ques 17 Find all information regarding classrooms
(RoomType = 'C')

select * from location INNER JOIN ROOM ON ROOM.
RoomType = Location.RoomType where Location.RoomType =
'C';

```
SQL> select * from location INNER JOIN ROOM ON Room.RoomType =
2 Location.RoomType where location.RoomType = 'C';
```

ROOMID	BUILDING	ROOMNO	CAPACITY	ROOMTYPE	ROOMTYPE	ROOMDESC
2	WingB	602	3	C	C	Classroom

Ques 18 Create a new table, SP03SECT, for Spring 2003 semester course sections using a sub query. Include courseID, section, FacultyID & RoomID columns only.

Create table SP03SECT as select CourseID, section, facultyID, RoomID from crssection;

select * from sp03sect;

```
SQL> select * from SP03SECT;
```

COURSEID	SECTION	FACULTYID	ROOMID
DB2002	01	123	13
WT1006	03	345	14
CO1004	02	345	14

Ques 19 Delete rows from SP03SECT table for faculty Rajdeep.

delete from SP03SECT where facultyID = (select facultyID from faculty where name = 'Rajdeep');

```
SQL> delete from SP03SECT where facultyID = (select facultyID from faculty where name='Rajdeep');
```

```
2 rows deleted.
```

Ques 20 Find faculty member who do not teach any course in Spring 2003 semester Use correlated subquery with NOT EXISTS operator on 8P038ECT table.

select f.name from faculty f where NOT EXISTS
(select * from crseccion c where c.facultyID =
f.facultyID and c.termID = '8P22');

```
SQL> SELECT f.name from faculty f WHERE NOT EXISTS  
2 (select * from crssection c where c.facultyid = f.facultyid and c.termid = 'SP22');  
  
NAME  
-----  
Biswajit  
HimansuDas  
Williams  
Collins
```