



“Only Knowledge can provide salvation”

DBMS PRACTICAL

QUESTION 2

Name- Vishav Kumar
Course- BSc(Hons) Computer Science
Section- B
Semester- 4
Roll no.- 10807
Subject- DBMS

QUESTIONS

1. Display all the details of all employees working in the company.

```
mysql> select * from employee;
```

essn	fname	minit	lname	bdate	address	gender	salary	superssn	dno
123456789	John	B	Smith	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
333445555	Franklin	T	Wong	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
453453453	Joyce	A	English	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5
666884444	Ramesh	K	Narayan	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
888665555	James	E	Borg	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1
987654321	Jennifer	S	Wallace	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
987987987	Ahmad	V	Jabbar	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
999887777	Alicia	J	Zelaya	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4

8 rows in set (0.00 sec)

```
mysql> select * from employee,department where employee.dno=department.dnumber;
```

essn	fname	minit	lname	bdate	address	gender	salary	superssn	dno	dnumber	dname	mgrssn	mgrstartdate
123456789	John	B	Smith	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5	5	Research	333445555	1988-05-22
333445555	Franklin	T	Wong	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5	5	Research	333445555	1988-05-22
453453453	Joyce	A	English	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5	5	Research	333445555	1988-05-22
666884444	Ramesh	K	Narayan	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5	5	Research	333445555	1988-05-22
888665555	James	E	Borg	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1	1	Headquarters	888665555	1981-06-19
987654321	Jennifer	S	Wallace	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4	4	Administration	987654321	1995-01-01
987987987	Ahmad	V	Jabbar	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4	4	Administration	987654321	1995-01-01
999887777	Alicia	J	Zelaya	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4	4	Administration	987654321	1995-01-01

8 rows in set (0.00 sec)

```
mysql>
```

2. Display ssn, lname, fname, address of employees who work in department no 7

```
mysql> select essn,fname,lname,address from employee where dno=5;
```

essn	fname	lname	address
123456789	John	Smith	731 Fondren, Houston, TX
333445555	Franklin	Wong	638 Voss, Houston, TX
453453453	Joyce	English	5631 Rice, Houston, TX
666884444	Ramesh	Narayan	975 Fire Oak, Humble, TX

4 rows in set (0.00 sec)

```
mysql>
```

3. Retrieve the birthdate and address of the employee whose name is 'Franklin T. Wong'

```
mysql> select bdate,address from employee where fname="Franklin" and minit="T" and lname="Wong";
```

bdate	address
1955-12-08	638 Voss, Houston, TX

1 row in set (0.04 sec)

```
mysql>
```

4. Retrieve the name and salary of every employee

```
mysql> select fname,lname,salary from employee;
```

fname	lname	salary
John	Smith	30000
Franklin	Wong	40000
Joyce	English	25000
Ramesh	Narayan	38000
James	Borg	55000
Jennifer	Wallace	43000
Ahmad	Jabbar	25000
Alicia	Zelaya	25000

8 rows in set (0.00 sec)

```
mysql>
```

5. Retrieve all distinct salary values

```
mysql> select distinct salary from employee;
+-----+
| salary |
+-----+
| 30000  |
| 40000  |
| 25000  |
| 38000  |
| 55000  |
| 43000  |
+-----+
6 rows in set (0.05 sec)

mysql>
```

6. Retrieve all employee names whose address is in 'Bellaire'

```
mysql> select concat(fname, ' ', lname) as Name from employee where address like "%Bellaire%";
+-----+
| Name          |
+-----+
| Jennifer Wallace |
+-----+
1 row in set (0.00 sec)

mysql>
```

7. Retrieve all employees who were born during the 1950s

```
mysql>
mysql> select essn, fname, lname from employee where year(bdate) between 1950 and 1960;
+-----+-----+-----+
| essn      | fname   | lname |
+-----+-----+-----+
| 333445555 | Franklin | Wong  |
+-----+-----+-----+
1 row in set (0.00 sec)

mysql> █
```

8. Retrieve all employees in department 5 whose salary is between 50,000 and 60,000(inclusive)

```
mysql> select essn, fname, lname from employee where dno=5 and salary between 50000 and 60000;
Empty set (0.00 sec)

mysql> select essn, fname, lname from employee where dno=1 and salary between 50000 and 60000;
+-----+-----+-----+
| essn      | fname   | lname |
+-----+-----+-----+
| 888665555 | James   | Borg  |
+-----+-----+-----+
1 row in set (0.00 sec)

mysql> █
```

9. Retrieve the names of all employees who do not have supervisors

```
mysql> select concat(fname, ' ', minit, ' ', lname) as Name from employee where superssn is null;
+-----+
| Name          |
+-----+
| James E Borg  |
+-----+
1 row in set (0.03 sec)

mysql> █
```

10. Retrieve SSN and department name for all employees

```
mysql> select essn,dname as Department_Name from employee,department where employee.dno=department.dnumber;
+-----+-----+
| essn   | Department_Name |
+-----+-----+
| 888665555 | Headquarters    |
| 987654321 | Administration   |
| 987987987 | Administration   |
| 999887777 | Administration   |
| 123456789 | Research         |
| 333445555 | Research         |
| 453453453 | Research         |
| 666884444 | Research         |
+-----+-----+
8 rows in set (0.00 sec)
```

11. Retrieve the name and address of all employees who work for the 'Research' department

```
mysql> select concat(fname,' ',minit,' ',lname)as Name,address from employee,department where employee.dno=department.dnumber and dname="Research";
+-----+-----+
| Name           | address          |
+-----+-----+
| John B Smith   | 731 Fondren, Houston, TX |
| Franklin T Wong | 638 Voss, Houston, TX   |
| Joyce A English | 5631 Rice, Houston, TX  |
| Ramesh K Narayan | 975 Fire Oak, Humble, TX |
+-----+-----+
4 rows in set (0.00 sec)

mysql>
```

12. For every project located in 'Stafford', list the project number, the controlling department number, and the department manager's last name, address, and birthdate.

```
mysql> select lname,address,bdate,pno,project.dno from employee,department,project where employee.essn=department.mgrssn and project.dno=department.dnumber and project.location="Stafford";
+-----+-----+-----+-----+-----+
| lname | address          | bdate   | pno | dno |
+-----+-----+-----+-----+-----+
| Wallace | 291 Berry, Bellaire, TX | 1941-06-20 | 10 | 4 |
| Wallace | 291 Berry, Bellaire, TX | 1941-06-20 | 30 | 4 |
+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)

mysql>
```

13. For each employee, retrieve the employee's name, and the name of his or her immediate supervisor.

```
mysql> select concat(e1.fname,' ',e1.minit,' ',e1.lname) as Employee , concat(e2.fname,' ',e2.minit,' ',e2.lname) as Supervisor from employee as e1,employee e2 where e1.superssn=e2.essn;
+-----+-----+
| Employee           | Supervisor        |
+-----+-----+
| John B Smith       | Franklin T Wong   |
| Franklin T Wong    | James E Borg      |
| Joyce A English    | Franklin T Wong   |
| Ramesh K Narayan   | Franklin T Wong   |
| Jennifer S Wallace | James E Borg      |
| Ahmad V Jabbar     | Jennifer S Wallace |
| Alicia J Zelaya    | Jennifer S Wallace |
+-----+-----+
7 rows in set (0.00 sec)

mysql>
```

14. Retrieve all combinations of Employee Name and Department Name

```
mysql> select concat(fname,' ',minit,' ',lname) as Employee_Name,dname from employee,department where employee.dno=department.dnumber;
+-----+-----+
| Employee_Name | dname |
+-----+-----+
| John B Smith  | Research |
| Franklin T Wong | Research |
| Joyce A English | Research |
| Ramesh K Narayan | Research |
| James E Borg   | Headquarters |
| Jennifer S Wallace | Administration |
| Ahmad V Jabbar  | Administration |
| Alicia J Zelaya | Administration |
+-----+-----+
8 rows in set (0.00 sec)

mysql>
```

15. Make a list of all project numbers for projects that involve an employee whose last name is 'Narayan' either as a worker or as a manager of the department that controls the project.

```
mysql> select pno from works_on,employee where works_on.essn=employee.essn and employee.lname="Narayan" UNION select pno from department,employee,project where department.dnumber=project.dno and department.mgrssn=employee.essn and employee.lname="Narayan";
+----+
| pno |
+----+
| 3 |
+----+
1 row in set (0.00 sec)

mysql>
```

16. Increase the salary of all employees working on the 'ProductX' project by 15%. Retrieve employee name and increased salary of these employees.

```
mysql> update employee set salary=1.15*salary where essn in(select essn from works_on,project where project.pno=works_on.pno and project.pname='ProductX');
Query OK, 2 rows affected (0.06 sec)
Rows matched: 2 Changed: 2 Warnings: 0

mysql> select e.essn,e.salary from employee e,project p,works_on w where e.essn=w.essn and w.pno=p.pno and p.pname='ProductX';
+-----+-----+
| essn | salary |
+-----+-----+
| 123456789 | 45000 |
| 453453453 | 37500 |
+-----+-----+
2 rows in set (0.00 sec)
```

17. Retrieve a list of employees and the project name each works in, ordered by the employee's department, and within each department ordered alphabetically by employee first name.

```
mysql> select d.dname,e.fname,p.pname from employee e,department d,project p,works_on w where e.dno=d.dnumber and p.pno=w.pno and e.essn=w.essn order by d.dnumber,fname;
+-----+-----+-----+
| dname | fname | pname |
+-----+-----+-----+
| Headquarters | James | Reorganization |
| Administration | Ahmad | Computerization |
| Administration | Ahmad | Newbenefits |
| Administration | Alicia | Newbenefits |
| Administration | Alicia | Computerization |
| Administration | Jennifer | Reorganization |
| Administration | Jennifer | Newbenefits |
| Research | Franklin | ProductY |
| Research | Franklin | Reorganization |
| Research | Franklin | ProductZ |
| Research | John | ProductY |
| Research | John | ProductX |
| Research | Joyce | ProductY |
| Research | Joyce | ProductX |
| Research | Ramesh | ProductZ |
+-----+-----+-----+
15 rows in set (0.00 sec)

mysql>
```

18. Select the names of employees whose salary does not match with salary of any employee in department 10.

```
mysql> select fname from employee where salary not in(select salary from employee where dno=5);
+-----+
| fname |
+-----+
| James |
| Jennifer |
| Ahmad |
| Alicia |
+-----+
4 rows in set (0.00 sec)

mysql>
```

19. Retrieve the name of each employee who has a dependent with the same first name and same sex as the employee.

```
mysql> select e.fname from employee e,dependent d where e.essn=d.essn and e.fname=d.dependent_name and e.gender=d.gender;
Empty set (0.00 sec)
```

20. Retrieve the employee numbers of all employees who work on project located in Bellaire, Houston, or Stafford.

```
mysql> select distinct(e.essn) from employee e,works_on w,project p where e.essn=w.essn and w.pno=p.pno and p.plocation in("Bellaire","Stafford","Houston");
+-----+
| essn |
+-----+
| 123456789 |
| 453453453 |
| 333445555 |
| 666884444 |
| 987987987 |
| 999887777 |
| 888665555 |
| 987654321 |
+-----+
8 rows in set (0.00 sec)

mysql>
```

21. Find the sum of the salaries of all employees, the maximum salary, the minimum salary, and the average salary. Display with proper headings.

```
mysql> select sum(salary) as "Total Salary",avg(salary) as "Average Salary",max(salary) as "Maximum Salary",min(salary) as "Minimum Salary" from employee;
+-----+-----+-----+-----+
| Total Salary | Average Salary | Maximum Salary | Minimum Salary |
+-----+-----+-----+-----+
| 308500 | 38562.5000 | 55000 | 25000 |
+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

22. Find the sum of the salaries and number of employees of all employees of the 'Marketing' department, as well as the maximum salary, the minimum salary, and the average salary in this department.

```
mysql> select count(*) as "Number of employee", sum(salary), max(salary), min(salary), avg(salary) from employee, department where employee.dno=department.dnumber and department.dname="Research";
```

Number of employee	sum(salary)	max(salary)	min(salary)	avg(salary)
4	160500	45000	37500	40125.0000

```
1 row in set (0.00 sec)
```

23. Select the names of employees whose salary is greater than the average salary of all employees in department 10.

```
mysql> select essn, fname from employee where salary > (select avg(salary) from employee where dno=5);
```

essn	fname
123456789	John
888665555	James
987654321	Jennifer

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

24. For each department, retrieve the department number, the number of employees in the department, and their average salary.

```
mysql> select dno, dname, count(*), avg(salary) from employee, department where employee.dno=department.dnumber group by dnumber;
```

dno	dname	count(*)	avg(salary)
1	Headquarters	1	55000.0000
4	Administration	3	31000.0000
5	Research	4	40125.0000

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

25. For each project, retrieve the project number, the project name, and the number of employees who work on that project.

```
mysql> select project.pno, pname, count(*) as "Number of Employees" from project, works_on where project.pno=works_on.pno group by project.pno;
```

pno	pname	Number of Employees
1	ProductX	2
2	ProductY	3
3	ProductZ	2
10	Computerization	2
20	Reorganization	3
30	Newbenefits	3

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

26. Change the location and controlling department number for all projects having more than 5 employees to 'Bellaire' and 6 respectively.

```
mysql> update project set dno=6 and plocation="Bellaire" where pno in(select pno from works_on having count(*)>5);
Query OK, 0 rows affected (0.03 sec)
Rows matched: 1 Changed: 0 Warnings: 0
```


27. For each department having more than 10 employees, retrieve the department no, no of employees drawing more than 40,000 as salary.

```
mysql> select dno,count(*) from employee,department where employee.dno=department.dnumber and salary > 40000 group by dno having count(*)>10;
Empty set (0.00 sec)
```

28. Insert a record in Project table which violates referential integrity constraint with respect to Department number. Now remove the violation by making necessary insertion in the Department table.

```
mysql> insert into project values(12,"Ques28 Dummy_proj","Switzerland",12);
ERROR 1452 (23000): Cannot add or update a child row: a foreign key constraint fails (`sem4q2`.`project`, CONSTRAINT `project_ibfk_1` FOREIGN KEY (`dno`) REFERENCES `department` (`dnumber`))
mysql> insert into project values(12,"Ques28 Dummy_proj","Switzerland",1);
Query OK, 1 row affected (0.07 sec)

mysql> select * from project;
+----+-----+-----+-----+
| pno | pname          | plocation | dno |
+----+-----+-----+-----+
| 1   | ProductX       | Bellaire  | 1   |
| 2   | ProductY       | Sugarland | 5   |
| 3   | ProductZ       | Houston   | 5   |
| 10  | Computerization | Stafford  | 4   |
| 12  | Ques28 Dummy_proj | Switzerland | 1   |
| 20  | Reorganization | Houston   | 1   |
| 30  | Newbenefits     | Stafford  | 4   |
+----+-----+-----+-----+
7 rows in set (0.00 sec)

mysql> select * from department;
+-----+-----+-----+-----+
| dnumber | dname          | mgrssn | mgrstartdate |
+-----+-----+-----+-----+
| 1       | Headquarters   | 888665555 | 1981-06-19 |
| 4       | Administration | 987654321 | 1995-01-01 |
| 5       | Research       | 333445555 | 1988-05-22 |
+-----+-----+-----+-----+
3 rows in set (0.00 sec)
```

29. Delete all dependents of employee whose ssn is '123456789'.

```
mysql> delete from dependent where essn=123456789;
Query OK, 0 rows affected (0.00 sec)

mysql> select * from dependent;
+-----+-----+-----+-----+-----+
| essn | dependent_name | gender | bdate | relationship |
+-----+-----+-----+-----+-----+
| 333445555 | Alice | F | 1986-04-05 | Daughter |
| 333445555 | Joy | F | 1958-05-03 | Spouse |
| 333445555 | Theodore | M | 1983-10-25 | Son |
| 987654321 | Abner | M | 1942-02-28 | Spouse |
+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

30. Delete an employee from Employee table with ssn = '12345' (make sure that this employee has some dependents, is working on some project, is a manager of some department and is supervising some employees). Check and display the cascading effect on Dependent and Works on table. In Department table MGRSSN should be set to default value and in Employee table SUPERSSN should be set to NULL

30.

```
delete from dependent where essn=987654321;
delete from works_on where essn=987654321;
update department set mgrssn=default where mgrssn=987654321;
update employee set superssn=null where superssn=987654321;
depete from employee where essn=987654321;
|
```

31. Perform a query using alter command to drop/add field and a constraint in Employee table.

```
mysql> alter table employee add email varchar(250);
Query OK, 8 rows affected (0.21 sec)
Records: 8 Duplicates: 0 Warnings: 0

mysql> desc employee;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| essn  | int(11) | NO | PRI | NULL | |
| fname | varchar(50) | NO | | NULL | |
| minit | varchar(50) | YES | | NULL | |
| lname | varchar(50) | NO | | NULL | |
| bdate | date | NO | | NULL | |
| address | varchar(50) | NO | | NULL | |
| gender | char(10) | NO | | NULL | |
| salary | int(11) | NO | | NULL | |
| superssn | int(11) | YES | MUL | NULL | |
| dno | int(11) | YES | MUL | NULL | |
| email | varchar(250) | YES | | NULL | |
+-----+-----+-----+-----+-----+-----+
11 rows in set (0.03 sec)

mysql> alter table employee drop column email;
Query OK, 8 rows affected (0.21 sec)
Records: 8 Duplicates: 0 Warnings: 0
```