



VIT[®]
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

Course Code: BCSE302P

Course Name: Database Systems Lab

Assessment – 2

Name: Shivam Dave

Reg. No.: 21BCB0107

Aim: To understand different types of function in SQL

```

Administration  ADMINISTRATION  administration
Headquater     HEADQUATER     headquater
Finance        FINANCE       finance
IT             IT            it

SQL> SELECT first_name, last_name, SUBSTRING(address, 5, 6)
      2 FROM employee;
SELECT first_name, last_name, SUBSTRING(address, 5, 6)
      *
ERROR at line 1:
ORA-00904: "SUBSTRING": invalid identifier

SQL> select * from employee;

FIRST_NAME      MI LAST_NAME      SSN_NUMBE BIRTHDAY  ADDRESS
-----
Doug            E  Gilbert          123       09-JUN-68 Chennai
Joyce           PAN
Franklin        T  Wong             125       08-DEC-72 Delhi
Jennifer        S  Wallace          564       20-JUN-83 Chennai
John            B  Smith            678       09-JAN-87 Madurai
Ramesh          K  Narayan          234       15-SEP-85 Bangalore
Rahul           S  Ram              555       10-OCT-73 Vellore

7 rows selected.

SQL> select * from project;

PROJECT_NAME    PROJECT_NO PROJECT_LOCATIO  DEPT_NO
-----
ProjectA        3388 Delhi           1
ProjectB        1945 Hyderabad    1
ProjectC        6688 Chennai      2
ProjectD        2423 Chennai      2
ProjectE        7745 Bangalore    3
ProjectF        Chennai         3
ProjectF        Chennai         3
ProjectF        1234 Chennai      4

8 rows selected.

SQL> select * from deptt;

DEPT_NAME      DEPT_NO MANAGERSS  MANAGER_D
-----
Administration  2 564       03-JAN-12
Headquater      1 678       16-DEC-14
Finance         3 234       18-MAY-13
IT              4 123       12-JUN-15

SQL>

```

1. Display the employee names whose bdate is on or before 1978.

Query:

Select * from employee where birthday<('01-JAN-1978');

Output:

```
SQL> Select * from employee where birthday<('01-JAN-1978');
```

FIRST_NAME	MI	LAST_NAME	SSN_NUMBE	BIRTHDAY	ADDRESS
Doug	E	Gilbert	123	09-JUN-68	Chennai
Joyce		PAN	124	07-FEB-73	Vellore
Franklin	T	Wong	125	08-DEC-72	Delhi
Rahul	S	Ram	555	10-OCT-73	Vellore

```
SQL>
```

2. Display all the department names in upper case and lower case.

Query: SELECT dept_name, UPPER(dept_name), LOWER(dept_name)
FROM deptt;

Output:

```
SQL> SELECT dept_name, UPPER(dept_name), LOWER(dept_name)
2 FROM deptt;
```

DEPT_NAME	UPPER(DEPT_NAME)	LOWER(DEPT_NAME)
Administration	ADMINISTRATION	administration
Headquater	HEADQUATER	headquater
Finance	FINANCE	finance
IT	IT	it

```
SQL>
```

3. Display the first four characters and last four of the department names using substring function.

Query: select substr(dept_name,1,4), substr(dept_name,-4) from deptt;

Output:

```
SQL> SELECT SUBSTR(dept_name,1, 4), SUBSTR(dept_name, -4)
2 FROM deptt;
```

SUBS	SUBS
Admi	tion
Head	ater
Fina	ance
IT	

```
SQL> |
```

4. Display the substring of the Address (starting from 5th position to 11 th position) of all employees.

Query:

Select substr(address,5,11) from employee;

Output:

```
SQL> SELECT SUBSTR(dept_name,1, 4), SUBSTR(dept_name, -4)
       2 FROM deptt;
```

```
SUBS SUBS
-----
Admi tion
Head ater
Fina ance
IT
```

```
SQL> |
```

5. Display the Mgrstartdate on adding three months to it.**Query:**

Select add_months(manager_doj,3) from deptt;

Output:

```
SQL> select add_months(manager_doj,3) from deptt;
```

```
ADD_MONTH
-----
03-APR-12
16-MAR-15
18-AUG-13
12-SEP-15
```

```
SQL> |
```

6. Display the age of all the employees rounded to two digits.

Query: select abs(round((months_between(sysdate,birthday)/12),2)) from employee;

Output:

```
SQL> select abs(round((months_between(sysdate,birthday)/12),2)) from employee;
```

```
ABS(ROUND((MONTHS_BETWEEN(SYSDATE,BIRTHDAY)/12),2))
-----
54.98
50.32
50.48
39.95
36.4
37.71
49.64
```

```
7 rows selected.
```

```
SQL> |
```

7. Display the bdate of all employees in the format 'DDthMonthYYYY'.

Query: SELECT first_name, last_name, TO_CHAR(birthday, 'DDth Mon YYYY')
AS birthday

FROM employee;

Output:

```
SQL> SELECT first_name, last_name, TO_CHAR(birthday, 'DDth Mon YYYY') AS birthday
2 FROM employee;
```

FIRST_NAME	LAST_NAME	BIRTHDAY
Doug	Gilbert	09TH Jun 1968
Joyce	PAN	07TH Feb 1973
Franklin	Wong	08TH Dec 1972
Jennifer	Wallace	20TH Jun 1983
John	Smith	09TH Jan 1987
Ramesh	Narayan	15TH Sep 1985
Rahul	Ram	10TH Oct 1973

7 rows selected.

```
SQL> |
```

8. Find the last day and next day of the month in which each manager has joined.

Query:

Select manager_doj, last_Day(manager_Doj) from deptt;

Output:

```
SQL> select manager_doj, last_Day(manager_Doj) from deptt;
```

MANAGER_D	LAST_DAY(
03-JAN-12	31-JAN-12
16-DEC-14	31-DEC-14
18-MAY-13	31-MAY-13
12-JUN-15	30-JUN-15

```
SQL> |
```

9. Print a substring from the string 'Harini'.

Query:

Select substr('harini',1,4) from dual;

Output:

```
SQL> select substr('Harini',1,4) from dual;

SUBS
-----
Hari
SQL>
```

10. Replace the string 'ni' from 'Harini' by 'sh'.

Query: select replace('harini','ni','sh') from dual;

Output:

```
SQL> select replace('harini','ni','sh') from dual;

REPLAC
-----
harish
SQL>
```

11. Print the length of all the department names.

Query: select length(dept_name) from deptt;

Output:

```
SQL> select length(dept_name) from deptt;

LENGTH(DEPT_NAME)
-----
14
10
7
2
SQL>
```

12. Display the date after 10 months from current date.

Query:

Select add_months(sysdate,10) from dual;

Output:

```
SQL> select add_months(sysdate,10) from dual;

ADD_MONTH
-----
01-APR-24

SQL> |
```

13. Display the next occurrence of Friday in this month.

Query:

Select next_day(sysdate,'friday') from dual;

Output:

```
SQL> select next_day(sysdate,'friday') from dual;

NEXT_DAY(
-----
02-JUN-23

SQL>
```

14. Display the project location padded with **** on left side.

Query: select Lpad(project_location,length(project_location)+3,'****') from project;

Output:

```
SQL> select Lpad(project_location,length(project_location)+3,'****') from project;

LPAD(PROJECT_LOCATION,LENGTH(PROJECT_LOCATION)+3,'****')
-----
***Delhi
***Hyderabad
***Chennai
***Chennai
***Bangalore
***Chennai
***Chennai
***Chennai

8 rows selected.

SQL>
```

15. Display the day of next Saturday.

Query:

Select next_day(sysdate, 'Saturday') from dual;

Output:

```
SQL> select next_day(sysdate,'saturday') from dual;

NEXT_DAY(
-----
03-JUN-23
```

16. Display the last date of Feb 2022.

Query: select last_day('01-Feb-2022') from dual;

Output:

```
SQL> select last_day('01-Feb-2022') from dual;

LAST_DAY(
-----
28-FEB-22

SQL>
```

17. Count the number of days present between today and Sunday.

Query:

SELECT (CASE WHEN TO_CHAR(SYSDATE, 'D') >= 1 THEN 7 - TO_CHAR(SYSDATE, 'D') + 1 ELSE 1 - TO_CHAR(SYSDATE, 'D') END) AS days_until_Sunday from dual;

Output:

```
SQL> SELECT (CASE WHEN TO_CHAR(SYSDATE, 'D') >= 1 THEN 7 - TO_CHAR(SYSDATE, 'D') + 1 ELSE 1 -
2 FROM dual;

DAYS_UNTIL_SUNDAY
-----
2

SQL> |
```

18. Display all the employees whose experience is greater than 14 yrs.

Query:

Select Manager_doj, First_Name from Employee natural join Deptt where (months_between(sysdate, Manager_doj) /12)>14;

Output:


```
SQL> Select Manager_doj, First_Name from Employee natural join Deptt where (months_between(sysdate, Manager_doj) /12)>14;
no rows selected
SQL> |
```

Group Functions

19. Find the total salary of all employees

Query: Select sum(salary) from employee;

Output:

```
SQL> select sum(salary) from employee;
SUM(SALARY)
-----
    351000
SQL> |
```

20. Display the min, max salary given to employees

Query: select max(salary),min(salary) from employee;

Output:

```
SQL> select max(salary),min(salary) from employee;
MAX(SALARY) MIN(SALARY)
-----
    80000    30000
SQL>
```

21. Display the avg salary of all employees

Query: Select avg(salary) from employee;

Output:

```
SQL> select avg(salary) from employee;
AVG(SALARY)
-----
    50142.8571
SQL> |
```

22. Display the total number of employees

Query: select count(first_name) from employee;

Output:

```
SQL> select count(first_name)from employee;
COUNT(FIRST_NAME)
-----
              7
SQL> |
```

23. How many different departments are there in the 'employee' table

Query: select count(dept_no) from employee;

Output:

```
SQL> select count(dept_no) from employee;
COUNT(DEPT_NO)
-----
              7
SQL> |
```

24. For each department display the minimum and maximum employee salaries

Query: select dept_no, min(salary) , max(salary) from employee group by dept_no;

Output:

```
SQL> select dept_no, min(salary) , max(salary) from employee group by dept_no;
DEPT_NO MIN(SALARY) MAX(SALARY)
-----
      1         30000         80000
      2         40000         43000
      5         70000         70000
      3         38000         50000
SQL> |
```

25. Print the average annual salary.

Query: select avg(salary)*12 from employee;

Output:

```
SQL> select avg(salary)*12 from employee;

AVG(SALARY)*12
-----
601714.286

SQL> |
```

26. Count the number of employees over 30 age.

Query: select count(*) from employee
where(months_between(sysdate,birthday))>30*12;

Output:

```
SQL> select count(*) from employee where(months_between(sysdate,birthday))>30*12;

COUNT(*)
-----
7

SQL> |
```

27. Print the Department name and average salary of each department.

Query: select d.dept_name, avg(e.salary) from deptt d
join employee e ON d.dept_no=e.dept_no
group by d.dept_name;

Output:

```
SQL> select d.dept_name, avg(e.salary) from deptt d
2 join employee e ON d.dept_no=e.dept_no
3 group by d.dept_name;

DEPT_NAME      AVG(E.SALARY)
-----
Headquater      55000
Administration  41500
Finance         44000

SQL> |
```

28. Display the department name which contains more than 2 employees.

Query: select d.dept_name from deptt d
join employee e ON d.dept_no=e.dept_no
group by d.dept_name

having count(*)>2;

Output:

```
SQL> select d.dept_name from deptt d
      2 join employee e ON d.dept_no=e.dept_no
      3 group by d.dept_name
      4 having count(*)>2;
```

```
DEPT_NAME
-----
Headquater
```

```
SQL>
```

29. Calculate the average salary of employees by department and age

Query: select dept_no, avg(salary) from (employee natural join deptt) group by dept_no;

Output:

```
SQL> select dept_no, avg(salary) from (employee natural join deptt) group by dept_no;
```

DEPT_NO	AVG(SALARY)
1	60000
2	41500
3	44000

```
SQL> |
```

30. Count separately the number of employees in the finance and administration department.

Query:

select first_name from (employee natural join deptt) where dept_name='Finance';

select count(*) from (employee natural join deptt) where dept_name='Finance';

select first_name from (employee natural join deptt) where dept_name='Administration';

select count(*) from (employee natural join deptt) where dept_name='Administration';

Output:

```
SQL> select first_name from(employee natural join deptt) where dept_name='Finance';
FIRST_NAME
-----
Ramesh
Rahul

SQL> select count(*) from (employee natural join deptt) where dept_name='Finance';
COUNT(*)
-----
2

SQL> select first_name from(employee natural join deptt) where dept_name='Administration';
FIRST_NAME
-----
Franklin
Jennifer

SQL> select count(*) from (employee natural join deptt) where dept_name='Administration';
COUNT(*)
-----
2

SQL> |
```

31. List out the employees based on their seniority.

Query: select first_name,round(months_between(sysdate,manager_doj))as experience, dept_name from (employee natural join deptt) where deptt.managerssn=employee.ssn_number order by experience;

Output:

```
SQL> select first_name,round(months_between(sysdate,manager_doj))as experience, dept_name from
FIRST_NAME      EXPERIENCE  DEPT_NAME
-----
John              102 Headquater
Ramesh            121 Finance
Jennifer          137 Administration

SQL> |
```

32. Display the total number of employees who are assigned to a department.

Query: select dept_no, count(*) from (Deptt natural join Employee) group by dept_no;

Output:

```
SQL> select dept_no, count(*) from (Deptt natural join Employee) group by dept_no;
```

DEPT_NO	COUNT(*)
1	3
2	2
3	2

```
SQL> |
```

33. Display only department numbers that are paying a total salary of more than 30000 to its

Employees

Query: select dept_no,sum(salary) from employee group by dept_no having sum(salary)>30000;

Output:

```
SQL> select dept_no,sum(salary) from employee group by dept_no having sum(salary)>30000;
```

DEPT_NO	SUM(SALARY)
1	180000
2	83000
3	88000

```
SQL> |
```

34. Display all the employees working under the department 1.

Query: select first_name, dept_no from employee where dept_no=1;

Output:

```
SQL> select first_name, dept_no from employee where dept_no=1;
```

FIRST_NAME	DEPT_NO
Doug	1
Joyce	1
John	1

```
SQL>
```

35. Display employees whose salary is between 5000 and 10000 in department no 1.

Query: select first_name, dept_no, salary from employee where dept_no=1 and salary>5000 and salary<10000;

Output:

```
SQL> select first_name, dept_no, salary from employee where dept_no=1 and salary>5000 and salary<6000;
no rows selected
SQL>
```

SUB QUERY AND VIEW IN SQL

Aim: To understand different types of function in SQL

Consider the schema (used in EX2):

1. Find the employee who is getting highest salary in the department headquarters.

Query: select first_name,dept_name, Salary from(Employee natural join deptt) where salary=(select max(salary) from employee) and dept_name='Headquater';

Output:

```
SQL> select first_name,dept_name, Salary from(Employee natural join deptt) where salary=(select max(salary) from employee) and dept_name='Headquater';
FIRST_NAME      DEPT_NAME      SALARY
-----
Doug            Headquater      80000
SQL>
```

2. Find the employees who earn the same salary as the minimum salary for each Department.

Query:

Select First_Name, Dept_name,Dept_No, Salary from (Employee natural join Deptt) where Salary in(Select min(Salary) from Employee group by Dept_No);

Output:

```
SQL> Select First_Name, Dept_name,Dept_No, Salary from (Employee natural join Deptt) where Salary in(Select min(Salary) from Employee group by Dept_No);
FIRST_NAME      DEPT_NAME      DEPT_NO      SALARY
-----
John            Headquater      1            30000
Franklin        Administration  2            40000
Ramesh          Finance         3            38000
SQL>
```

3. Find the employee whose salary is greater than average salary of department 2.

Query:

Select First_Name, Dept_Name, Dept_No, Salary from (Employee natural join Deptt) where Salary > (Select avg(Salary) from Employee where Dept_No = 2);

Output:

```
SQL> Select First_Name, Dept_Name, Dept_No, Salary from (Employee natural join Deptt) where Sa
```

FIRST_NAME	DEPT_NAME	DEPT_NO	SALARY
Doug	Headquater	1	80000
Joyce	Headquater	1	70000
Jennifer	Administration	2	43000
Rahul	Finance	3	50000

```
SQL>
```

4. List out all the department names with their individual employee's strength.

Query:

Select Dept_Name, Dept_No, count(*) from (Employee natural join Deptt) group by Dept_No, Dept_Name;

Output:

```
SQL> Select Dept_Name, Dept_No, count(*) from (Employee natural join Deptt) group by Dept_No,
```

DEPT_NAME	DEPT_NO	COUNT(*)
Finance	3	2
Administration	2	2
Headquater	1	3

```
SQL> |
```

5. Find out the department name having highest employee strength.

Query: Select Dept_name, Dept_No from Deptt where Dept_No in (Select Dept_No from Employee group by Dept_No having count(*) in (select max(Employees) from (Select count(*) Employees from Employee group by Dept_no)));

Output:


```
SQL> Select Dept_name, Dept_No from Deptt where Dept_No in (Select Dept_No from Employee gr
```

DEPT_NAME	DEPT_NO
Headquater	1

```
SQL> |
```

6. List out all the departments and average salary drawn by their employees.

Query: Select Dept_name, avg(Salary) from (Employee natural join Deptt) group by Dept_Name, Dept_No;

Output:

```
SQL> Select Dept_name, avg(Salary) from (Employee natural join Deptt) group by Dept_Name, Dep
```

DEPT_NAME	AUG(SALARY)
Administration	41500
Headquater	60000
Finance	44000

```
SQL>
```

7. Find average salary for each department.

Query: Select Dept_name, avg(Salary) from (Employee natural join Deptt) group by Dept_Name, Dept_No;

Output:

```
SQL> Select Dept_name, avg(Salary) from (Employee natural join Deptt) group by Dept_Name, Dep
```

DEPT_NAME	AUG(SALARY)
Administration	41500
Headquater	60000
Finance	44000

```
SQL>
```

8. Create a view to display the employee details who is working in Administration department.

Query: Create view Administration_Emp as Select * from Employee where Dept_No = (Select Dept_No from Deptt where Dept_Name = 'Administration');

Select * from Administration_Emp

Output:

```

SQL> select * from administration_emp;
FIRST_NAME      MI LAST_NAME      SSN_NUMBE BIRTHDAY
-----
ADDRESS          S      SALARY SUPERVISO
-----
DEPT_NO
-----
Franklin        T  Wong          125      08-DEC-72
Delhi           M      40000 123
2
Jennifer        S  Wallace        564      20-JUN-83
Chennai         F      43000 123
2
FIRST_NAME      MI LAST_NAME      SSN_NUMBE BIRTHDAY
-----
ADDRESS          S      SALARY SUPERVISO
-----
DEPT_NO
-----

SQL> set linesize 200;
SQL> select * from administration_emp;
FIRST_NAME      MI LAST_NAME      SSN_NUMBE BIRTHDAY ADDRESS
-----
Franklin        T  Wong          125      08-DEC-72 Delhi
Jennifer        S  Wallace        564      20-JUN-83 Chennai
SQL>

```

9. Create a logical table to store employee details who is getting salary more than 10000.

Query: Create view Salary10000 as Select * from Employee where salary>10000;

Select * from salary10000;

Output:

```

SQL> Create view Salary10000 as Select * from Employee where salary>10000;
View created.
SQL> select * from salary10000;
FIRST_NAME      MI LAST_NAME      SSN_NUMBE BIRTHDAY ADDRESS
-----
Doug            E  Gilbert        123      09-JUN-68 Chennai
Joyce           PAN 124      07-FEB-73 Vellore
Franklin        T  Wong          125      08-DEC-72 Delhi
Jennifer        S  Wallace        564      20-JUN-83 Chennai
John            B  Smith          678      09-JAN-87 Madurai
Ramesh          K  Narayan        234      15-SEP-85 Bangalore
Rahul           S  Ram            555      10-OCT-73 Vellore
7 rows selected.
SQL>

```

10. Create a view to display the employee details who is working in finance or administration

department.

Query: Create view Emp_Finance as Select * from Employee where Dept_no in (Select Dept_No from Deptt where Dept_Name = 'Finance' or Dept_Name = 'Administration');

select * from emp_finance;

Output:

```
SQL> Create view Emp_Finance as Select * from Employee where Dept_no in (Select Dept_No from Deptt where Dept_Name = 'Finance' or Dept_Name = 'Administration');
View created.
SQL> select * from emp_finance;
```

FIRST_NAME	MI	LAST_NAME	SSN_NUMBE	BIRTHDAY	ADDRESS
Franklin	T	Wong	125	08-DEC-72	Delhi
Jennifer	S	Wallace	564	20-JUN-83	Chennai
Ramesh	K	Narayan	234	15-SEP-85	Bangalore
Rahul	S	Ram	555	10-OCT-73	Vellore

```
SQL> |
```

10.Create a view to display all employees whose salary is greater than “Jennifer”.

Query: Create view Salary_of_Jennifer as Select * from Employee where Salary > (Select Salary from Employee where First_Name = 'Jennifer');

Select * from Salary_of_Jennifer

Output:

```
SQL> select * from salary_of_jennifer;
```

FIRST_NAME	MI	LAST_NAME	SSN_NUMBE	BIRTHDAY	ADDRESS
Doug	E	Gilbert	123	09-JUN-68	Chennai
Joyce		PAN	124	07-FEB-73	Vellore
Rahul	S	Ram	555	10-OCT-73	Vellore

```
SQL> |
```

11.Create a view to display all employees in descending order of employee first name

Query: Create view First_Desc_Name as select * from Employee order by First_Name desc;

Select * from First_Desc_Name;

Output:

```
SQL> Create view First_Desc_Name as select * from Employee order by First_Name desc;
View created.
SQL> select * from First_Desc_Name;
```

FIRST_NAME	MI	LAST_NAME	SSN_NUMBE	BIRTHDAY	ADDRESS
Ramesh	K	Narayan	234	15-SEP-85	Bangalore
Rahul	S	Ram	555	10-OCT-73	Vellore
Joyce		PAN	124	07-FEB-73	Vellore
John	B	Smith	678	09-JAN-87	Madurai
Jennifer	S	Wallace	564	20-JUN-83	Chennai
Franklin	T	Wong	125	08-DEC-72	Delhi
Doug	E	Gilbert	123	09-JUN-68	Chennai

```
7 rows selected.
SQL>
```

13. Create a view to display all employees whose salary is greater than the average salary of the employees.

Query: Create view Salary_Greater_Avg as select * from Employee where Salary > (Select avg(Salary) from Employee);

select * from salary_greater_avg;

Output:

```
SQL> Create view Salary_Greater_Avg as select * from Employee where Salary > (Select avg(Salary) from Employee);
View created.
SQL> select * from salary_greater_avg;
```

FIRST_NAME	MI	LAST_NAME	SSN_NUMBE	BIRTHDAY	ADDRESS
Doug	E	Gilbert	123	09-JUN-68	Chennai
Joyce		PAN	124	07-FEB-73	Vellore

```
SQL>
```

14. Create a view to display all employees whose salary is lower than all the employees of the department 1.

Query: Create view Salary_Lowerthan_Dep1 as select * from Employee where Salary < (Select min(Salary) from Employee where Dept_No = 1);

Select * from salary_lowerthan_dep1;

Output:

```
SQL> Create view Salary_Lowerthan_Dep1 as select * from Employee where Salary < (Select min(Sa
View created.
SQL> Select * from salary_lowerthan_Dep1;
no rows selected
SQL> |
```

15. Display all male employees whose salary is greater than the maximum salary of the female

Employees

Query: Select First_Name, Salary from Employee where Sex = 'M' and salary > (Select max(Salary) from Employee where Sex = 'F');

Output:

```
SQL> Select First_Name, Salary from Employee where Sex = 'M' and salary > (Select max(Salary)
FIRST_NAME          SALARY
-----
Doug                80000
SQL>
```

JOINS

Aim: To understand how to relate and access data from multiple tables.

Consider the schema given in exercise 2, and execute the following queries

1. Find the names of all the employees who are directly supervised by 'Joyce'.

Query: Select * from Employee where Supervisor_SSN = (Select SSN_Number from Employee where First_Name = 'Joyce');

Output:

```
SQL> Select * from Employee where Supervisor_SSN = (Select SSN_Number from Employee where First
FIRST_NAME          MI LAST_NAME          SSN_NUMBE BIRTHDAY  ADDRESS
-----
John                B  Smith                678        09-JAN-87 Madurai
Ramesh              K  Narayan              234        15-SEP-85 Bangalore
SQL>
```

2. Find the names of all the employees who are working in department 'Headquarter'.

Query: Select * from (Employee natural Deptt) where Dept_Name='Headquarter';

Output:

```
SQL> Select * from (Employee natural join Deptt) where Dept_Name = 'Headquarter';
```

DEPT_NO	FIRST_NAME	MI	LAST_NAME	SSN_NUMBE	BIRTHDAY	ADDRESS
1	Doug	E	Gilbert	123	09-JUN-68	Chennai
1	Joyce		PAN	124	07-FEB-73	Vellore
1	John	B	Smith	678	09-JAN-87	Madurai

```
SQL> |
```

3. List the department names and if has a manager then display the manager name too.

Query: Select Dept_Name, First_Name, SSN_Number from (Deptt natural join Employee) where Employee.SSN_Number = Deptt.ManagerSSN;

Output:

```
SQL> Select Dept_Name, First_Name, SSN_Number from (Deptt natural join Employee) where Employee
```

DEPT_NAME	FIRST_NAME	SSN_NUMBE
Administration	Jennifer	564
Headquater	John	678
Finance	Ramesh	234

```
SQL> |
```

4. Retrieve the names of the departments which have more than 2 employees.

Query:

Select Dept_Name, Dept_No from Deptt where Dept_No in (Select Dept_No from Employee having count(*) > 2 group by Dept_No);

Output:

```
SQL> Select Dept_Name, Dept_No from Deptt where Dept_No in (Select Dept_No from Employee havir
```

DEPT_NAME	DEPT_NO
Headquater	1

```
SQL> |
```

5. Display the employee name along with the deptname and MgrSSN of that department.

Query: Select First_Name, Dept_Name, ManagerSSN from (Employee natural join Deptt);

Output:

```
SQL> Select First_Name, Dept_Name, ManagerSSN from (Employee natural join Deptt);
```

FIRST_NAME	DEPT_NAME	MANAGERSS
Doug	Headquater	678
Joyce	Headquater	678
Franklin	Administration	564
Jennifer	Administration	564
John	Headquater	678
Ramesh	Finance	234
Rahul	Finance	234

7 rows selected.

```
SQL> |
```

6. Find out the ename, deptname, deptno, eno of those who are working under some department

number (1 or 2 or 3)

Query: Select First_Name, SSN_Number, Dept_Name, Dept_No from (Employee natural join Deptt) where Dept_No = 1;

Select First_Name, SSN_Number, Dept_Name, Dept_No from (Employee natural join Deptt) where Dept_No = 2;

Select First_Name, SSN_Number, Dept_Name, Dept_No from (Employee natural join Deptt) where Dept_No = 3;

Output:

```
SQL> Select First_Name, SSN_Number, Dept_Name, Dept_No from (Employee natural join Deptt) wher
```

FIRST_NAME	SSN_NUMBE	DEPT_NAME	DEPT_NO
Doug	123	Headquater	1
Joyce	124	Headquater	1
John	678	Headquater	1

```
SQL> Select First_Name, SSN_Number, Dept_Name, Dept_No from (Employee natural join Deptt) wher
```

FIRST_NAME	SSN_NUMBE	DEPT_NAME	DEPT_NO
Franklin	125	Administration	2
Jennifer	564	Administration	2

```
SQL> Select First_Name, SSN_Number, Dept_Name, Dept_No from (Employee natural join Deptt) wher
```

FIRST_NAME	SSN_NUMBE	DEPT_NAME	DEPT_NO
Ramesh	234	Finance	3
Rahul	555	Finance	3

```
SQL> |
```