**ASSIGNMENT-1**

1. *Write an SQL query to fetch “FIRST\_NAME” from the Worker table using the alias name as <WORKER\_NAME>.*
2. *Write an SQL query to fetch “FIRST\_NAME” from the Worker table in upper case.*
3. *Write an SQL query to fetch unique values of DEPARTMENT from Worker table.*
4. *Write an SQL query to find the position of the alphabet (‘a’) in the first name column ‘Amitabh’ from the Worker table.*

*Notes:*

* *The INSTR method is case-sensitive by default.*
* *Using a Binary operator will make INSTR work as the case-sensitive function.*

1. *Write an SQL query to print the FIRST\_NAME from the Worker table after removing white spaces from the right side.*
2. *Write an SQL query to print the DEPARTMENT from the Worker table after removing white spaces from the left side.*
3. *Write an SQL query to print the FIRST\_NAME from the Worker table after replacing ‘a’ with ‘A’.*
4. *Write an SQL query to print the FIRST\_NAME and LAST\_NAME from the Worker table into a single column COMPLETE\_NAME. A space char should separate them.*
5. *Write an SQL query to print all Worker details from the Worker table order by FIRST\_NAME Ascending.*
6. *Write an SQL query to print all Worker details from the Worker table order by FIRST\_NAME Ascending and DEPARTMENT Descending.*
7. *Write an SQL query to print details of the Workers whose FIRST\_NAME contains ‘a’.*
8. *Write an SQL query to fetch worker names with salaries >= 50000 and <= 100000.*
9. *Write an SQL query to fetch the first 50% records from a table.*
10. *Write an SQL query to show the last record from a table.*
11. *Swapping the Values of first name and last name Columns in a worker table*

postgres=# CREATE DATABASE company;

CREATE DATABASE

postgres=# \c company;

You are now connected to database "company" as user "postgres".

company=# CREATE TABLE worker(WORKER\_ID INT PRIMARY KEY,FIRST\_NAME VARCHAR(50),LAST\_NAME VARCHAR(50), SALARY INT,JOINING\_DATE DATE,DEPARTMENT VARCHAR(15));

CREATE TABLE

company=# INSERT INTO worker(WORKER\_ID,FIRST\_NAME,LAST\_NAME,SALARY,JOINING\_DATE,DEPARTMENT)

VALUES(001,'Monika','Arora',100000,'2019-06-08','HR');

INSERT 0 1

company=# INSERT INTO worker(WORKER\_ID,FIRST\_NAME,LAST\_NAME,SALARY,JOINING\_DATE,DEPARTMENT)

VALUES(002,'Niharika','Verma',80000,'2019-06-02','Admin');

INSERT 0 1

company=# INSERT INTO worker(WORKER\_ID,FIRST\_NAME,LAST\_NAME,SALARY,JOINING\_DATE,DEPARTMENT)VALUES(003,'Vishal','Singhal',300000,'2019-06-03','HR');

INSERT 0 1

company=# INSERT INTO worker(WORKER\_ID,FIRST\_NAME,LAST\_NAME,SALARY,JOINING\_DATE,DEPARTMENT)VALUES(004,'Amithab','Singh',500000,'2019-06-04','Admin');

INSERT 0 1

company=# INSERT INTO worker(WORKER\_ID,FIRST\_NAME,LAST\_NAME,SALARY,JOINING\_DATE,DEPARTMENT)VALUES(005,'Vivek','Bhati',500000,'2019-06-05','Admin');

INSERT 0 1

company=# INSERT INTO worker(WORKER\_ID,FIRST\_NAME,LAST\_NAME,SALARY,JOINING\_DATE,DEPARTMENT)VALUES(006,'Vipul','Diwan',200000,'2019-06-05','Account');

INSERT 0 1

company=# INSERT INTO worker(WORKER\_ID,FIRST\_NAME,LAST\_NAME,SALARY,JOINING\_DATE,DEPARTMENT)VALUES(007,'Satish','Kumar',75000,'2019-06-14','Account');

INSERT 0 1

company=# INSERT INTO worker(WORKER\_ID,FIRST\_NAME,LAST\_NAME,SALARY,JOINING\_DATE,DEPARTMENT)VALUES(008,'Deepika','Chauhan',90000,'2019-06-21','Admin');

INSERT 0 1

company=# SELECT \* FROM worker;

worker\_id | first\_name | last\_name | salary | joining\_date | department

-----------+------------+-----------+--------+--------------+------------

1 | Monika | Arora | 100000 | 2019-06-08 | HR

2 | Niharika | Verma | 80000 | 2019-06-02 | Admin

3 | Vishal | Singhal | 300000 | 2019-06-03 | HR

4 | Amithab | Singh | 500000 | 2019-06-04 | Admin

5 | Vivek | Bhati | 500000 | 2019-06-05 | Admin

6 | Vipul | Diwan | 200000 | 2019-06-05 | Account

7 | Satish | Kumar | 75000 | 2019-06-14 | Account

8 | Deepika | Chauhan | 90000 | 2019-06-21 | Admin

(8 rows)

company=# CREATE TABLE bonus(WORKER\_REF\_ID int,BONUS\_DATE DATE,BONUS\_AMOUNT INT);

CREATE TABLE

company=# INSERT INTO bonus(WORKER\_REF\_ID,BONUS\_DATE,BONUS\_AMOUNT) VALUES (1,'2020-06-06',5000),(2,'2020-06-03',3000),(3,'2020-06-04',4000),(1,'2020-06-05',4500),(2,'2020-06-05',3500);

INSERT 0 5

company=# CREATE TABLE Title(WORKER\_REF\_ID INT,WORKER\_TITLE VARCHAR(50),AFFECTED\_FROM DATE)

;

CREATE TABLE

company=# INSERT INTO Title(WORKER\_REF\_ID,WORKER\_TITLE,AFFECTED\_FROM)VALUES(1,'Manager','20

19-06-08'),(2,'Executive','2019-06-02'),(8,'Executive','2019-06-03'),(5,'Manager','2019-06-8'),(4,'Asst.Manager','2019-06-02'),(7,'Executive','2019-06-03');

INSERT 0 6

company=# SELECT \* FROM worker;

worker\_id | first\_name | last\_name | salary | joining\_date | department

-----------+------------+-----------+--------+--------------+------------

1 | Monika | Arora | 100000 | 2019-06-08 | HR

2 | Niharika | Verma | 80000 | 2019-06-02 | Admin

3 | Vishal | Singhal | 300000 | 2019-06-03 | HR

4 | Amithab | Singh | 500000 | 2019-06-04 | Admin

5 | Vivek | Bhati | 500000 | 2019-06-05 | Admin

6 | Vipul | Diwan | 200000 | 2019-06-05 | Account

7 | Satish | Kumar | 75000 | 2019-06-14 | Account

8 | Deepika | Chauhan | 90000 | 2019-06-21 | Admin

(8 rows)

company=# SELECT \* FROM Bonus;

worker\_ref\_id | bonus\_date | bonus\_amount

---------------+------------+--------------

1 | 2020-06-06 | 5000

2 | 2020-06-03 | 3000

3 | 2020-06-04 | 4000

1 | 2020-06-05 | 4500

2 | 2020-06-05 | 3500

(5 rows)

company=# SELECT \* FROM Title;;

worker\_ref\_id | worker\_title | affected\_from

---------------+--------------+---------------

1 | Manager | 2019-06-08

2 | Executive | 2019-06-02

8 | Executive | 2019-06-03

5 | Manager | 2019-06-08

4 | Asst.Manager | 2019-06-02

7 | Executive | 2019-06-03

(6 rows)

company=# SELECT FIRST\_NAME FROM worker AS worker\_name;

first\_name

------------

Monika

Niharika

Vishal

Amithab

Vivek

Vipul

Satish

Deepika

(8 rows)

company=# SELECT UPPER(FIRST\_NAME) AS WORKER\_NAME FROM worker;

worker\_name

-------------

MONIKA

NIHARIKA

VISHAL

AMITHAB

VIVEK

VIPUL

SATISH

DEEPIKA

(8 rows)

company=# SELECT DISTINCT department FROM worker;

department

------------

Admin

Account

HR

(3 rows)

company=# SELECT POSITION('a' IN FIRST\_NAME) AS position\_a FROM worker WHERE FIRST\_NAME = '

Amitabh';

position\_a

------------

(0 rows)

company=# SELECT POSITION('a' IN FIRST\_NAME) AS position\_a FROM worker WHERE FIRST\_NAME = 'Amithab';

position\_a

------------

6

(1 row)

company=# SELECT RTRIM(FIRST\_NAME) AS trimmed\_name FROM worker;

trimmed\_name

--------------

Monika

Niharika

Vishal

Amithab

Vivek

Vipul

Satish

Deepika

(8 rows)

company=# SELECT REPLACE(FIRST\_NAME,'a','A') AS modified\_first\_name FROM worker;

modified\_first\_name

---------------------

MonikA

NihArikA

VishAl

AmithAb

Vivek

Vipul

SAtish

DeepikA

(8 rows)

company=# SELECT FIRST\_NAME||' '|| LAST\_NAME AS complete\_name FROM worker;

complete\_name

-----------------

Monika Arora

Niharika Verma

Vishal Singhal

Amithab Singh

Vivek Bhati

Vipul Diwan

Satish Kumar

Deepika Chauhan

(8 rows)

company=# SELECT \* FROM worker ORDER BY FIRST\_NAME ASC;

worker\_id | first\_name | last\_name | salary | joining\_date | department

-----------+------------+-----------+--------+--------------+------------

4 | Amithab | Singh | 500000 | 2019-06-04 | Admin

8 | Deepika | Chauhan | 90000 | 2019-06-21 | Admin

1 | Monika | Arora | 100000 | 2019-06-08 | HR

2 | Niharika | Verma | 80000 | 2019-06-02 | Admin

7 | Satish | Kumar | 75000 | 2019-06-14 | Account

6 | Vipul | Diwan | 200000 | 2019-06-05 | Account

3 | Vishal | Singhal | 300000 | 2019-06-03 | HR

5 | Vivek | Bhati | 500000 | 2019-06-05 | Admin

(8 rows)

company=# SELECT \* FROM worker ORDER BY FIRST\_NAME DESC;

worker\_id | first\_name | last\_name | salary | joining\_date | department

-----------+------------+-----------+--------+--------------+------------

5 | Vivek | Bhati | 500000 | 2019-06-05 | Admin

3 | Vishal | Singhal | 300000 | 2019-06-03 | HR

6 | Vipul | Diwan | 200000 | 2019-06-05 | Account

7 | Satish | Kumar | 75000 | 2019-06-14 | Account

2 | Niharika | Verma | 80000 | 2019-06-02 | Admin

1 | Monika | Arora | 100000 | 2019-06-08 | HR

8 | Deepika | Chauhan | 90000 | 2019-06-21 | Admin

4 | Amithab | Singh | 500000 | 2019-06-04 | Admin

(8 rows)

company=# SELECT \* FROM worker ORDER BY FIRST\_NAME ASC,DEPARTMENT DESC;

worker\_id | first\_name | last\_name | salary | joining\_date | department

-----------+------------+-----------+--------+--------------+------------

4 | Amithab | Singh | 500000 | 2019-06-04 | Admin

8 | Deepika | Chauhan | 90000 | 2019-06-21 | Admin

1 | Monika | Arora | 100000 | 2019-06-08 | HR

2 | Niharika | Verma | 80000 | 2019-06-02 | Admin

7 | Satish | Kumar | 75000 | 2019-06-14 | Account

6 | Vipul | Diwan | 200000 | 2019-06-05 | Account

3 | Vishal | Singhal | 300000 | 2019-06-03 | HR

5 | Vivek | Bhati | 500000 | 2019-06-05 | Admin

(8 rows)

company=# SELECT \* FROM worker

company-# ORDER BY FIRST\_NAME ASC,DEPARTMENT DESC;

worker\_id | first\_name | last\_name | salary | joining\_date | department

-----------+------------+-----------+--------+--------------+------------

4 | Amithab | Singh | 500000 | 2019-06-04 | Admin

8 | Deepika | Chauhan | 90000 | 2019-06-21 | Admin

1 | Monika | Arora | 100000 | 2019-06-08 | HR

2 | Niharika | Verma | 80000 | 2019-06-02 | Admin

7 | Satish | Kumar | 75000 | 2019-06-14 | Account

6 | Vipul | Diwan | 200000 | 2019-06-05 | Account

3 | Vishal | Singhal | 300000 | 2019-06-03 | HR

5 | Vivek | Bhati | 500000 | 2019-06-05 | Admin

(8 rows)

company=# SELECT \* FROM worker WHERE FIRST\_NAME ILIKE '%A';

worker\_id | first\_name | last\_name | salary | joining\_date | department

-----------+------------+-----------+--------+--------------+------------

1 | Monika | Arora | 100000 | 2019-06-08 | HR

2 | Niharika | Verma | 80000 | 2019-06-02 | Admin

8 | Deepika | Chauhan | 90000 | 2019-06-21 | Admin

(3 rows)

company=# SELECT \* FROM worker WHERE FIRST\_NAME ILIKE '%A%';

worker\_id | first\_name | last\_name | salary | joining\_date | department

-----------+------------+-----------+--------+--------------+------------

1 | Monika | Arora | 100000 | 2019-06-08 | HR

2 | Niharika | Verma | 80000 | 2019-06-02 | Admin

3 | Vishal | Singhal | 300000 | 2019-06-03 | HR

4 | Amithab | Singh | 500000 | 2019-06-04 | Admin

7 | Satish | Kumar | 75000 | 2019-06-14 | Account

8 | Deepika | Chauhan | 90000 | 2019-06-21 | Admin

(6 rows)

company=# SELECT \* FROM worker WHERE FIRST\_NAME LIKE '%a%';

worker\_id | first\_name | last\_name | salary | joining\_date | department

-----------+------------+-----------+--------+--------------+------------

1 | Monika | Arora | 100000 | 2019-06-08 | HR

2 | Niharika | Verma | 80000 | 2019-06-02 | Admin

3 | Vishal | Singhal | 300000 | 2019-06-03 | HR

4 | Amithab | Singh | 500000 | 2019-06-04 | Admin

7 | Satish | Kumar | 75000 | 2019-06-14 | Account

8 | Deepika | Chauhan | 90000 | 2019-06-21 | Admin

(6 rows)

company=# SELECT FIRST\_NAME,SALARY FROM worker WHERE SALARY>=50000 AND SALARY <=100000;

first\_name | salary

------------+--------

Monika | 100000

Niharika | 80000

Satish | 75000

Deepika | 90000

(4 rows)

company=# WITH totalCount AS(SELECT COUNT(\*) AS total\_rows FROM worker),

company-# HalfCount AS(SELECT total\_rows/2 AS half\_rows FROM totalCount)

company-# SELECT \* FROM worker LIMIT(SELECT half\_rows FROM HalfCount);

worker\_id | first\_name | last\_name | salary | joining\_date | department

-----------+------------+-----------+--------+--------------+------------

1 | Monika | Arora | 100000 | 2019-06-08 | HR

2 | Niharika | Verma | 80000 | 2019-06-02 | Admin

3 | Vishal | Singhal | 300000 | 2019-06-03 | HR

4 | Amithab | Singh | 500000 | 2019-06-04 | Admin

(4 rows)

company=# SELECT \* FROM worker ORDER BY WORKER\_ID DESC LIMIT 1;

worker\_id | first\_name | last\_name | salary | joining\_date | department

-----------+------------+-----------+--------+--------------+------------

8 | Deepika | Chauhan | 90000 | 2019-06-21 | Admin

(1 row)

company=# UPDATE worker SET FIRST\_NAME = LAST\_NAME,

company-# LAST\_NAME=FIRST\_NAME;

UPDATE 8

company=# SELECT\*FROM worker;

worker\_id | first\_name | last\_name | salary | joining\_date | department

-----------+------------+-----------+--------+--------------+------------

1 | Arora | Monika | 100000 | 2019-06-08 | HR

2 | Verma | Niharika | 80000 | 2019-06-02 | Admin

3 | Singhal | Vishal | 300000 | 2019-06-03 | HR

4 | Singh | Amithab | 500000 | 2019-06-04 | Admin

5 | Bhati | Vivek | 500000 | 2019-06-05 | Admin

6 | Diwan | Vipul | 200000 | 2019-06-05 | Account

7 | Kumar | Satish | 75000 | 2019-06-14 | Account

8 | Chauhan | Deepika | 90000 | 2019-06-21 | Admin

(8 rows)

**ASSIGNMENT 2**

1. Write a query to find the third highest  salary from the EmployeeInfo table ?
2. Write a query to find the third highest  salary from the table without using TOP/LIMIT keyword ?
3. Write a query to find the duplicate row in  a table ?
4. Write  a query to calculate the even and odd records from a table ?
5. Write a query to display the first and last record from the EmployeeInfo table ?
6. How do you copy all rows of a table using query ?
7. Write  a query to retrieve the list of employees working in the same department ?
8. Write a query to retrieve the last 3 records from the EmployeeInfo table ?
9. Write a query to fetch details of an employee whose EmpLname ends with an alphabet ‘A’ and contains five alphabets ?

company=# CREATE TABLE employeeInfo (EMPID INT PRIMARY KEY,EMPFNAME VARCHAR(50),EMPLNAME VARCHAR(50),DEPARTMENT VARCHAR(25),SALARY INT);

CREATE TABLE

company=# INSERT INTO employeeInfo(EMPID,EMPFNAME,EMPLNAME,DEPARTMENT,SALARY)VALUES(1,'Karan','mehta','HR',300000),(2,'Rohit','Sharma','Admin',75000),(3,'Ankush','Rajput','Account',60000),(4,'Priyadarshini','Sharma','HR',500000),(5,'Sanket','Gupta','Developer',100000),(6,'Shruthi','Varyar','Admin',80000),(7,'Rohit','Sharma','Admin',75000)

company-# ;

INSERT 0 7

company=# SELECT \* FROM employeeInfo;

empid | empfname | emplname | department | salary

-------+---------------+----------+------------+--------

1 | Karan | mehta | HR | 300000

2 | Rohit | Sharma | Admin | 75000

3 | Ankush | Rajput | Account | 60000

4 | Priyadarshini | Sharma | HR | 500000

5 | Sanket | Gupta | Developer | 100000

6 | Shruthi | Varyar | Admin | 80000

7 | Rohit | Sharma | Admin | 75000

(7 rows)

**A.**

company=# SELECT DISTINCT SALARY FROM employeeInfo ORDER BY SALARY DESC OFFSET 2 LIMIT 1;

salary

--------

100000

(1 row)

**B.**

company=# SELECT MAX(SALARY) AS THIRD\_SALARY FROM(SELECT DISTINCT SALARY FROM employeeInfo ORDER BY SALARY DESC OFFSET 2);

third\_salary

--------------

100000

(1 row)

**C.**

company=# SELECT empfname, emplname, department, COUNT(\*)

company-# FROM employeeInfo

company-# GROUP BY empfname, emplname, department

company-# HAVING COUNT(\*) > 1;

empfname | emplname | department | count

----------+----------+------------+-------

Rohit | Sharma | Admin | 2

(1 row)

**D.**

company=# SELECT COUNT(\*) FILTER (WHERE EMPID%2 =0) AS even\_records,

company-# COUNT(\*) FILTER (WHERE EMPID%2=1) AS odd\_recorde FROM employeeInfo;

even\_records | odd\_recorde

--------------+-------------

3 | 4

(1 row)

**E.**

company=# (SELECT \* FROM employeeInfo ORDER BY EMPID ASC LIMIT 1) UNION ALL (SELECT \* FROM employeeInfo ORDER BY EMPID DESC LIMIT 1);

empid | empfname | emplname | department | salary

-------+----------+----------+------------+--------

1 | Karan | mehta | HR | 300000

7 | Rohit | Sharma | Admin | 75000

(2 rows)

**F.**

company=# CREATE TABLE employeeInfo\_backup (

company(# empid INT PRIMARY KEY,

company(# empfname VARCHAR(50),

company(# emplname VARCHAR(50),

company(# department VARCHAR(50),

company(# salary DECIMAL

company(# );

CREATE TABLE

company=# INSERT INTO employeeInfo\_backup

company-# SELECT \* FROM employeeInfo;

INSERT 0 7

**G.**

company=# SELECT \* FROM employeeInfo WHERE DEPARTMENT IN(SELECT DEPARTMENT FROM employeeInfo GROUP BY DEPARTMENT HAVING COUNT(\*)>1)ORDER BY DEPARTMENT, EMPID;

empid | empfname | emplname | department | salary

-------+---------------+----------+------------+--------

2 | Rohit | Sharma | Admin | 75000

6 | Shruthi | Varyar | Admin | 80000

7 | Rohit | Sharma | Admin | 75000

1 | Karan | mehta | HR | 300000

4 | Priyadarshini | Sharma | HR | 500000

(5 rows)

**H.**

company=# SELECT \* FROM employeeInfo ORDER BY EMPID DESC LIMIT 3;

empid | empfname | emplname | department | salary

-------+----------+----------+------------+--------

7 | Rohit | Sharma | Admin | 75000

6 | Shruthi | Varyar | Admin | 80000

5 | Sanket | Gupta | Developer | 100000

(3 rows)

**I.**

company=# SELECT \*

company-# FROM employeeInfo

company-# WHERE emplname LIKE '\_\_\_\_a' AND LENGTH(emplname) = 5;

empid | empfname | emplname | department | salary

-------+----------+----------+------------+--------

1 | Karan | mehta | HR | 300000

5 | Sanket | Gupta | Developer | 100000

(2 rows)

**ASSIGNMENT 3**

   3. You are the business owner and would like to obtain a sales report for category items and day of the week.

* Write an SQL query to report how many units in each category have been ordered on each day of the week.
* Return the result table ordered by category.
* The query result format is in the following example:

company=# SELECT \* FROM Orders;

order\_id | customer\_id | order\_date | item\_id | quantity

----------+-------------+------------+---------+----------

1 | 1 | 2020-06-01 | 1 | 10

2 | 1 | 2020-06-08 | 2 | 10

3 | 2 | 2020-06-02 | 1 | 5

4 | 3 | 2020-06-03 | 3 | 5

5 | 4 | 2020-06-04 | 4 | 1

6 | 4 | 2020-06-05 | 5 | 5

7 | 5 | 2020-06-05 | 1 | 10

8 | 5 | 2020-06-14 | 4 | 5

9 | 5 | 2020-06-21 | 3 | 5

(9 rows)

company=# SELECT \* FROM Items;

item\_id | item\_name | item\_category

---------+---------------+---------------

1 | LC Alg. Book | Book

2 | LC DB. Book | Book

3 | LC SmartPhone | Phone

4 | LC Phone 2020 | Phone

5 | LC SmartGlass | Glasses

6 | LC T-Shirt XL | T-Shirt

(6 rows)

^

company=# SELECT

company-# i.item\_category,

company-# COALESCE(SUM(CASE WHEN EXTRACT(DOW FROM o.order\_date) = 0 THEN o.quantity ELSE 0 END), 0) AS Sunday,

company-# COALESCE(SUM(CASE WHEN EXTRACT(DOW FROM o.order\_date) = 1 THEN o.quantity ELSE 0 END), 0) AS Monday,

company-# COALESCE(SUM(CASE WHEN EXTRACT(DOW FROM o.order\_date) = 2 THEN o.quantity ELSE 0 END), 0) AS Tuesday,

company-# COALESCE(SUM(CASE WHEN EXTRACT(DOW FROM o.order\_date) = 3 THEN o.quantity ELSE 0 END), 0) AS Wednesday,

company-# COALESCE(SUM(CASE WHEN EXTRACT(DOW FROM o.order\_date) = 4 THEN o.quantity ELSE 0 END), 0) AS Thursday,

company-# COALESCE(SUM(CASE WHEN EXTRACT(DOW FROM o.order\_date) = 5 THEN o.quantity ELSE 0 END), 0) AS Friday,

company-# COALESCE(SUM(CASE WHEN EXTRACT(DOW FROM o.order\_date) = 6 THEN o.quantity ELSE 0 END), 0) AS Saturday

company-# FROM

company-# Orders o

company-# JOIN

company-# Items i ON o.item\_id = i.item\_id

company-# GROUP BY

company-# i.item\_category;

item\_category | sunday | monday | tuesday | wednesday | thursday | friday | saturday

---------------+--------+--------+---------+-----------+----------+--------+----------

Phone | 10 | 0 | 0 | 5 | 1 | 0 | 0

Glasses | 0 | 0 | 0 | 0 | 0 | 5 | 0

Book | 0 | 20 | 5 | 0 | 0 | 10 | 0

(3 rows)

**ASSIGNMENT 4**

     4. The Employee table holds all employees. Every employee has an Id, and there is also a column for the department Id.

company=# SELECT\*FROM employee;

id | name | salary | department\_id

----+-------+--------+---------------

1 | Joe | 85000 | 1

2 | Henry | 80000 | 2

3 | sam | 60000 | 2

4 | Max | 90000 | 1

5 | Janet | 69000 | 1

6 | Randy | 85000 | 1

7 | Will | 70000 | 1

(7 rows)

company=# SELECT \* FROM department;

id | name

----+-------

1 | IT

2 | Sales

(2 rows)

* **Write a SQL query to find employees who earn the top three salaries in each of the departments. For the above tables, your SQL query should return the following rows (order of rows does not matter).**

company=# SELECT

company-# d.Name AS Department,

company-# e.Name AS Employee,

company-# e.Salary

company-# FROM

company-# Employee e

company-# JOIN

company-# Department d ON e.department\_id = d.id

company-# WHERE

company-# 3 > (SELECT COUNT(DISTINCT e2.Salary)

company(# FROM Employee e2

company(# WHERE e2.department\_id = e.department\_id AND e2.Salary > e.Salary)

company-# ORDER BY

company-# Department, Salary DESC;

department | employee | salary

------------+----------+--------

IT | Max | 90000

IT | Joe | 85000

IT | Randy | 85000

IT | Will | 70000

Sales | Henry | 80000

Sales | sam | 60000

(6 rows)