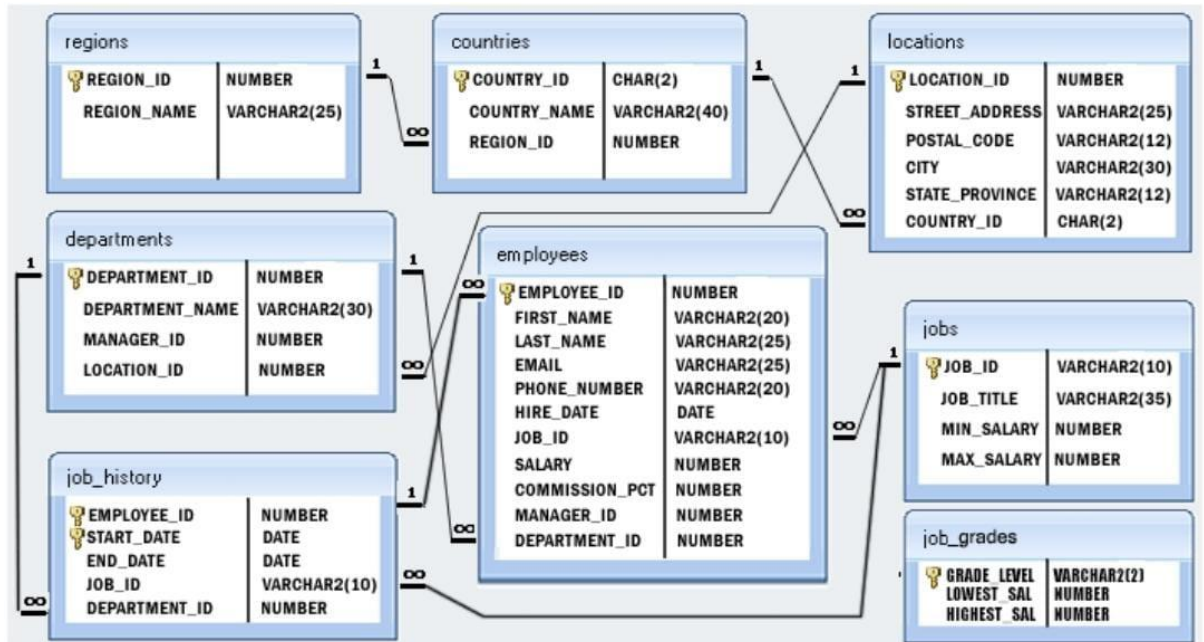


ASSIGNMENT OF POSTGRE SQL

ER DAIGRAM



SQL QUERY

- a. Write a SQL query to find those employees whose salaries are less than 6000. Return full name (first and last name), and salary.

```
SQL Shell (psql)
hr=# SELECT CONCAT(first_name,' ',last_name) AS Full_name,salary FROM employees WHERE salary<6000;
Full_name | salary
-----+-----
David Austin | 4800.00
Valli Pataballa | 4800.00
Diana Lorentz | 4200.00
Alexander Khoo | 3100.00
Shelli Baida | 2900.00
Sigal Tobias | 2800.00
Guy Himuro | 2600.00
Karen Colmenares | 2500.00
Kevin Mourgos | 5800.00
Julia Nayer | 3200.00
Irene Mikellineni | 2700.00
James Landry | 2400.00
Steven Markle | 2200.00
Laura Bissot | 3300.00
Mozhe Atkinson | 2800.00
James Marlow | 2500.00
TJ Olson | 2100.00
Jason Mallin | 3300.00
Michael Rogers | 2900.00
Ki Gee | 2400.00
Hazel Philtanker | 2200.00
Renske Ladwig | 3600.00
Stephen Stiles | 3200.00
John Seo | 2700.00
Joshua Patel | 2500.00
Trenna Rajes | 3500.00
Curtis Davies | 3100.00
Randall Matos | 2600.00
Peter Vargas | 2500.00
Winston Taylor | 3200.00
Jean Fleaur | 3100.00
Martha Sullivan | 2500.00
Girard Geoni | 2800.00
Nandita Sarchand | 4200.00
Alexis Bull | 4100.00
Julia Dellinger | 3400.00
Anthony Cabrio | 3800.00
Kelly Chung | 3800.00
Jennifer Dilly | 3600.00
Timothy Gates | 2900.00
Randall Perkins | 2500.00
```

- b. write a SQL query to identify employees who do not have a department number. Return employee_id, first_name, last_name, email, phone_number, hire_date, job_id, salary,commission_pct, manager_id and department_id

```
SQL Shell (psql)
hr=# SELECT * FROM employees WHERE employees.department_id IS NULL;
employee_id | first_name | last_name | email | phone_number | hire_date | job_id | salary | commission_pct | manager_id | department_id
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----
(1 row)

178 | Kimberly | Grant | KGRANT | 011.44.1644.429263 | 1999-05-24 00:00:00 | SA_REP | 7000.00 | 0.15 | 149 |
```

- c. write a SQL query to find the details of 'Marketing' department. Return all fields

```
SQL Shell (psql)
hr=# SELECT * FROM departments where departments.department_name = 'Marketing';
 department_id | department_name | manager_id | location_id
-----
(1 row)
20 | Marketing | 201 | 1800

hr=#
```

- d. write a SQL query to find those employees whose first name does not contain the letter 'M'. Sort the result-set in ascending order by department ID. Return full name (first and last name together), hire_date, salary and department_id

```
SQL Shell (psql)
hr=# SELECT CONCAT(first_name, ' ', last_name), hire_date, salary, department_id FROM employees WHERE first_name NOT LIKE 'M%' ORDER BY employees.department_id;
concat | hire_date | salary | department_id
-----
Jennifer Whalen | 1987-09-17 00:00:00 | 4400.00 | 10
Pat Fay | 1997-08-17 00:00:00 | 6000.00 | 20
Karen Colmenares | 1999-08-10 00:00:00 | 2500.00 | 30
Shelli Baida | 1997-12-24 00:00:00 | 2900.00 | 30
Alexander Khoo | 1995-05-18 00:00:00 | 3100.00 | 30
Den Raphaely | 1994-12-07 00:00:00 | 11000.00 | 30
Sigal Tobias | 1997-07-24 00:00:00 | 2800.00 | 30
Guy Himuro | 1998-11-15 00:00:00 | 2600.00 | 30
Susan Mavris | 1994-06-07 00:00:00 | 6500.00 | 40
Steven Markle | 2000-03-08 00:00:00 | 2200.00 | 50
Laura Bissot | 1997-08-20 00:00:00 | 3300.00 | 50
James Marlow | 1997-02-16 00:00:00 | 2500.00 | 50
TJ Olson | 1999-04-10 00:00:00 | 2100.00 | 50
Jason Mallin | 1996-06-14 00:00:00 | 3300.00 | 50
Ki Gee | 1999-12-12 00:00:00 | 2400.00 | 50
Vance Jones | 1999-03-17 00:00:00 | 2800.00 | 50
Hazel Philtanker | 2000-02-06 00:00:00 | 2200.00 | 50
Winston Taylor | 1998-01-24 00:00:00 | 3200.00 | 50
Jean Fleaur | 1998-02-23 00:00:00 | 3100.00 | 50
Girard Geoni | 2000-02-03 00:00:00 | 2800.00 | 50
Nandita Sarchand | 1996-01-27 00:00:00 | 4200.00 | 50
Alexis Bull | 1997-02-20 00:00:00 | 4100.00 | 50
Julia Dellinger | 1998-06-24 00:00:00 | 3400.00 | 50
Anthony Cabrio | 1999-02-07 00:00:00 | 3000.00 | 50
Kelly Chung | 1997-06-14 00:00:00 | 3800.00 | 50
Jennifer Dilly | 1997-08-13 00:00:00 | 3600.00 | 50
Timothy Gates | 1998-07-11 00:00:00 | 2900.00 | 50
Randall Perkins | 1999-12-19 00:00:00 | 2500.00 | 50
Sarah Bell | 1996-02-04 00:00:00 | 4000.00 | 50
Britney Everett | 1997-03-03 00:00:00 | 3900.00 | 50
Samuel McCain | 1998-07-01 00:00:00 | 3200.00 | 50
Renske Ladwig | 1995-07-14 00:00:00 | 3600.00 | 50
Stephen Stiles | 1997-10-26 00:00:00 | 3200.00 | 50
Joshua Patel | 1998-04-06 00:00:00 | 2500.00 | 50
Trenna Rajs | 1995-10-17 00:00:00 | 3500.00 | 50
Curtis Davies | 1997-01-29 00:00:00 | 3100.00 | 50
Randall Matos | 1998-03-15 00:00:00 | 2600.00 | 50
Peter Vargas | 1998-07-09 00:00:00 | 2500.00 | 50
John Seo | 1998-02-12 00:00:00 | 2700.00 | 50
Adam Fripp | 1997-04-10 00:00:00 | 8200.00 | 50
Payam Kaufling | 1995-05-01 00:00:00 | 7900.00 | 50
```

- e. write a SQL query to find those employees who were hired between November 5th, 2007 and July 5th, 2009. Return full name (first and last), job id and hire date

```
SQL Shell (psql)
hr=# SELECT CONCAT(first_name,' ',last_name) AS Full_name,job_id,hire_date FROM employees WHERE hire_date BETWEEN DATE '2007-11-05' AND DATE '2009-07-05';
Full_name | job_id | hire_date
-----
(0 rows)

hr=#
```

- f. write a SQL query to count the number of employees, the sum of all salary, and difference between the highest salary and lowest salaries by each job id. Return job_id, count, sum, salary_difference.

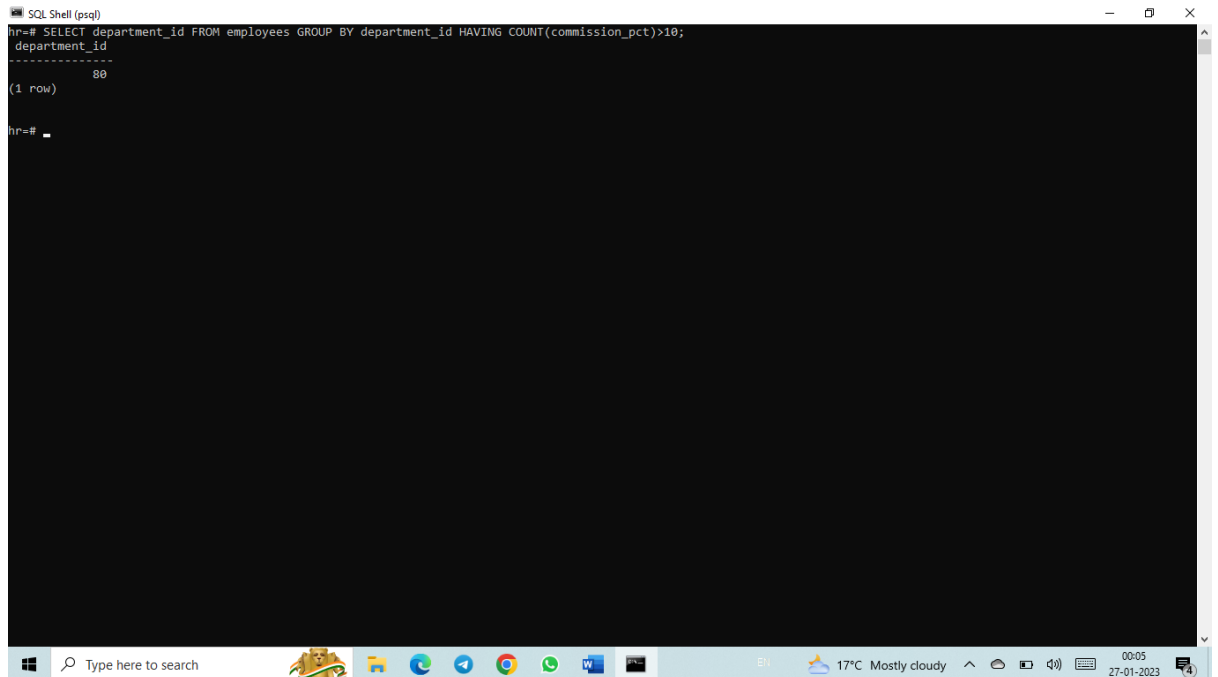
```
SQL Shell (psql)
hr=# SELECT DISTINCT(job_id), COUNT(employee_id),SUM(salary ),MAX(SALARY)-MIN(SALARY) AS salary_difference FROM employees group by job_id;
job_id | count | sum | salary_difference
-----
PR_REP | 1 | 10000.00 | 0.00
IT_PROG | 5 | 28800.00 | 4800.00
MK_MAN | 1 | 13000.00 | 0.00
AC_MGR | 1 | 12000.00 | 0.00
AD_PRES | 1 | 24000.00 | 0.00
ST_MAN | 5 | 36400.00 | 2400.00
SA_MAN | 5 | 61000.00 | 3500.00
MK_REP | 1 | 6000.00 | 0.00
FI_ACCOUNT | 5 | 39600.00 | 2100.00
AD_ASST | 1 | 4400.00 | 0.00
SA_REP | 30 | 250500.00 | 5400.00
PU_MAN | 1 | 11000.00 | 0.00
PU_CLERK | 5 | 13900.00 | 600.00
AC_ACCOUNT | 1 | 8300.00 | 0.00
HR_REP | 1 | 6500.00 | 0.00
AD_VP | 2 | 34000.00 | 0.00
FI_MGR | 1 | 12000.00 | 0.00
ST_CLERK | 20 | 55700.00 | 1500.00
SH_CLERK | 20 | 64300.00 | 1700.00
(19 rows)

hr=#
```

- g. write a SQL query to find the departments where more than ten employees receive commissions. Return department id

```
SQL Shell (psql)
hr=# SELECT department_id FROM employees GROUP BY department_id HAVING COUNT(commission_pct)>10;
 department_id
-----
          80
(1 row)

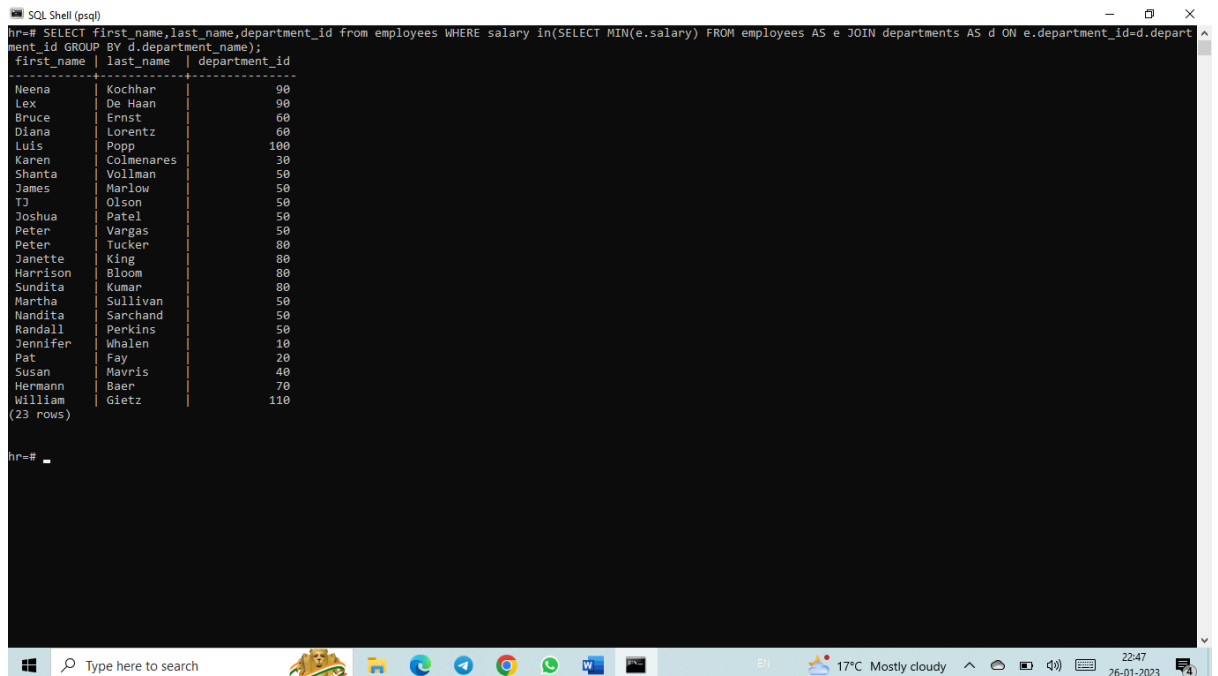
hr=#
```



- h. write a SQL query to find those employees whose salary matches the lowest salary of any of the departments. Return first name, last name and department ID

```
SQL Shell (psql)
hr=# SELECT first_name,last_name,department_id from employees WHERE salary in(SELECT MIN(e.salary) FROM employees AS e JOIN departments AS d ON e.department_id=d.department_id GROUP BY d.department_name);
 first_name | last_name | department_id
-----
 Neena      | Kochhar   | 90
 Lex        | De Haan   | 90
 Bruce      | Ernst     | 60
 Diana      | Lorentz   | 60
 Luis       | Popp      | 100
 Karen      | Colmenares | 30
 Shanta     | Vollman   | 50
 James      | Marlow    | 50
 JD         | Olson     | 50
 Joshua     | Patel     | 50
 Peter      | Vargas    | 50
 Peter      | Tucker   | 80
 Janette    | King      | 80
 Harrison   | Bloom     | 80
 Sundita    | Kumar     | 80
 Martha     | Sullivan  | 50
 Mandita    | Sarchand  | 50
 Randall    | Perkins   | 50
 Jennifer   | Whalen    | 10
 Pat        | Fay       | 20
 Susan      | Mavris    | 40
 Hermann    | Baer      | 70
 William    | Gietz     | 110
(23 rows)

hr=#
```



- i. write a SQL query to find all those employees who work in the Finance department. Return department ID, name (first), job ID and department name

```
SQL Shell (psql)
hr=# SELECT e.department_id,e.first_name,e.job_id,d.department_name FROM employees AS e JOIN departments AS d ON e.department_id=d.department_id WHERE d.department_name='Finance';
 department_id | first_name | job_id | department_name
-----
          100 | Nancy     | FI_MGR | Finance
          100 | Daniel    | FI_ACCOUNT | Finance
          100 | John      | FI_ACCOUNT | Finance
          100 | Ismael    | FI_ACCOUNT | Finance
          100 | Jose Manuel | FI_ACCOUNT | Finance
          100 | Luis      | FI_ACCOUNT | Finance
(6 rows)

hr=#
```

- j. write a SQL query to find those employees who get second-highest salary. Return all the fields of the employees

```
SQL Shell (psql)
hr=# SELECT * FROM employees WHERE salary =(SELECT DISTINCT(salary) FROM employees order by salary desc OFFSET 1 LIMIT 1);
 employee_id | first_name | last_name | email | phone_number | hire_date | job_id | salary | commission_pct | manager_id | department_id
-----
          101 | Neena     | Kochhar   | NKOCHHAR | 515.123.4568 | 1988-09-21 00:00:00 | AD_VP | 17000.00 | 0.00 | 100 | 90
          102 | Lex       | De Haan   | LDEHAAN | 515.123.4569 | 1993-01-13 00:00:00 | AD_VP | 17000.00 | 0.00 | 100 | 90
(2 rows)

hr=#
```

- k. write a SQL query to find those employees who earn more than the average salary and work in the same department as an employee whose first name contains the letter 'J'. Return employee ID, first name and salary

```
SQL Shell (psql)
hr=# SELECT employee_id,first_name,salary FROM employees WHERE salary>(SELECT AVG(salary) FROM employees) AND department_id IN(SELECT department_id FROM employees WHERE
first_name LIKE '%J%');
 employee_id | first_name | salary
-----
108 | Nancy | 12000.00
109 | Daniel | 9000.00
110 | John | 8200.00
111 | Ismael | 7700.00
112 | Jose Manuel | 7800.00
113 | Luis | 6900.00
120 | Matthew | 8000.00
121 | Adam | 8200.00
122 | Payam | 7900.00
123 | Shanta | 6500.00
145 | John | 14000.00
146 | Karen | 13500.00
147 | Alberto | 12000.00
148 | Gerald | 11000.00
149 | Eleni | 10500.00
150 | Peter | 10000.00
151 | David | 9500.00
152 | Peter | 9000.00
153 | Christopher | 8000.00
154 | Nanette | 7500.00
155 | Oliver | 7000.00
156 | Janette | 10000.00
157 | Patrick | 9500.00
158 | Allan | 9000.00
159 | Lindsey | 8000.00
160 | Louise | 7500.00
161 | Sarath | 7000.00
162 | Clara | 10500.00
163 | Danielle | 9500.00
164 | Mattea | 7200.00
165 | David | 6800.00
168 | Lisa | 11500.00
169 | Harrison | 10000.00
170 | Tayler | 9600.00
171 | William | 7400.00
172 | Elizabeth | 7200.00
174 | Ellen | 11000.00
175 | Alyssa | 8800.00
176 | Jonathon | 8600.00
177 | Jack | 8400.00
```

- l. write a SQL query to find the first name, last name, department, city, and state province for each employee

```
SQL Shell (psql)
hr=# SELECT e.first_name,e.last_name,d.department_name,l.city,l.state_province FROM employees AS e JOIN departments AS d ON e.department_id=d.department_id JOIN locations AS l ON l.location_id=d.location_id;
 first_name | last_name | department_name | city | state_province
-----
Steven | King | Executive | Seattle | Washington
Neena | Kochhar | Executive | Seattle | Washington
Lex | De Haan | Executive | Seattle | Washington
Alexander | Hunold | IT | Southlake | Texas
Bruce | Ernst | IT | Southlake | Texas
David | Austin | IT | Southlake | Texas
Valli | Pataballa | IT | Southlake | Texas
Diana | Lorentz | IT | Southlake | Texas
Nancy | Greenberg | Finance | Seattle | Washington
Daniel | Faviet | Finance | Seattle | Washington
John | Chan | Finance | Seattle | Washington
Ismael | Sciarra | Finance | Seattle | Washington
Jose Manuel | Urman | Finance | Seattle | Washington
Luis | Popp | Finance | Seattle | Washington
Den | Raphaely | Purchasing | Seattle | Washington
Alexander | Khoo | Purchasing | Seattle | Washington
Shelli | Baida | Purchasing | Seattle | Washington
Sigal | Tobias | Purchasing | Seattle | Washington
Guy | Himuro | Purchasing | Seattle | Washington
Karen | Colmenares | Purchasing | Seattle | Washington
Matthew | Weiss | Shipping | South San Francisco | California
Adam | Fripp | Shipping | South San Francisco | California
Payam | Kaufling | Shipping | South San Francisco | California
Shanta | Vollman | Shipping | South San Francisco | California
Kevin | Mourgos | Shipping | South San Francisco | California
Julia | Nayer | Shipping | South San Francisco | California
Irene | Mikkilineni | Shipping | South San Francisco | California
James | Landry | Shipping | South San Francisco | California
Steven | Markle | Shipping | South San Francisco | California
Laura | Bissot | Shipping | South San Francisco | California
Mozhe | Atkinson | Shipping | South San Francisco | California
James | Marlow | Shipping | South San Francisco | California
TJ | Olson | Shipping | South San Francisco | California
Jason | Mallin | Shipping | South San Francisco | California
Michael | Rogers | Shipping | South San Francisco | California
Ki | Green | Shipping | South San Francisco | California
Hazel | Philtanker | Shipping | South San Francisco | California
Renske | Ladwig | Shipping | South San Francisco | California
Stephen | Stiles | Shipping | South San Francisco | California
John | Seo | Shipping | South San Francisco | California
```

- m. write a SQL query to display the department name, city, and state province for each department

```
SQL Shell (psql)
hr=# SELECT d.department_name,l.city,l.state_province FROM departments as d JOIN locations as l ON d.location_id =l.location_id;
 department_name | city | state_province
-----
Administration | Seattle | Washington
Marketing | Toronto | Ontario
Purchasing | Seattle | Washington
Human Resources | London | 
Shipping | South San Francisco | California
IT | Southlake | Texas
Public Relations | Munich | Bavaria
Sales | Oxford | Oxford
Executive | Seattle | Washington
Finance | Seattle | Washington
Accounting | Seattle | Washington
Treasury | Seattle | Washington
Corporate Tax | Seattle | Washington
Control And Credit | Seattle | Washington
Shareholder Services | Seattle | Washington
Benefits | Seattle | Washington
Manufacturing | Seattle | Washington
Construction | Seattle | Washington
Contracting | Seattle | Washington
Operations | Seattle | Washington
IT Support | Seattle | Washington
NOC | Seattle | Washington
IT Helpdesk | Seattle | Washington
Government Sales | Seattle | Washington
Retail Sales | Seattle | Washington
Recruiting | Seattle | Washington
Payroll | Seattle | Washington
(27 rows)

hr=#
```

- n. write a SQL query to find the department name, full name (first and last name) of the manager and their city

```
SQL Shell (psql)
hr=# SELECT d.department_name,CONCAT(e.first_name,' ',e.last_name),l.city FROM departments AS d JOIN employees AS e ON e.manager_id = d.manager_id JOIN locations AS l ON d.location_id=l.location_id;
 department_name | concat | city
-----
Executive | Neena Kochhar | Seattle
Executive | Lex De Haan | Seattle
IT | Bruce Ernst | Southlake
IT | David Austin | Southlake
IT | Valli Pataballa | Southlake
IT | Diana Lorentz | Southlake
Finance | Daniel Faviet | Seattle
Finance | John Chen | Seattle
Finance | Ismael Sciarra | Seattle
Finance | Jose Manuel Urman | Seattle
Finance | Luis Popp | Seattle
Executive | Den Raphaely | Seattle
Purchasing | Alexander Khoo | Seattle
Purchasing | Shelli Baida | Seattle
Purchasing | Sigal Tobias | Seattle
Purchasing | Guy Himuro | Seattle
Purchasing | Karen Colmenares | Seattle
Executive | Matthew Weiss | Seattle
Executive | Adam Fripp | Seattle
Executive | Payam Kaufling | Seattle
Executive | Shanta Vollman | Seattle
Executive | Kevin Mourgos | Seattle
Shipping | Laura Bissot | South San Francisco
Shipping | Mozhe Atkinson | South San Francisco
Shipping | James Marlow | South San Francisco
Shipping | TJ Olson | South San Francisco
Executive | John Russell | Seattle
Executive | Karen Partners | Seattle
Executive | Alberto Errazuriz | Seattle
Executive | Gerald Cambrault | Seattle
Executive | Eleni Zlotkey | Seattle
Sales | Peter Tucker | Oxford
Sales | David Bernstein | Oxford
Sales | Peter Hall | Oxford
Sales | Christopher Olsen | Oxford
Sales | Nanette Cambrault | Oxford
Sales | Oliver Tuvault | Oxford
Shipping | Nandita Sarchand | South San Francisco
Shipping | Alexis Bull | South San Francisco
Shipping | Julia Dellinger | South San Francisco
```


- o. write a SQL query to calculate the number of days worked by employees in a department of ID 80. Return employee ID, job title, number of days worked

```
SQL Shell (psql)
hr=# SELECT e.department_id,j.job_title, (jh.end_date-jh.start_date) as no_of_days_worked FROM employees AS e JOIN jobs as j ON e.job_id=j.job_id JOIN job_history as jh
ON j.job_id=jh.job_id WHERE e.department_id=80;
 department_id | job_title | no_of_days_worked
-----
80 | Sales Manager | 364 days
80 | Sales Manager | 364 days
80 | Sales Manager | 364 days
80 | Sales Manager | 364 days
80 | Sales Manager | 364 days
80 | Sales Representative | 282 days
80 | Sales Representative | 282 days
80 | Sales Representative | 282 days
80 | Sales Representative | 282 days
80 | Sales Representative | 282 days
80 | Sales Representative | 282 days
80 | Sales Representative | 282 days
80 | Sales Representative | 282 days
80 | Sales Representative | 282 days
80 | Sales Representative | 282 days
80 | Sales Representative | 282 days
80 | Sales Representative | 282 days
80 | Sales Representative | 282 days
80 | Sales Representative | 282 days
80 | Sales Representative | 282 days
80 | Sales Representative | 282 days
80 | Sales Representative | 282 days
80 | Sales Representative | 282 days
80 | Sales Representative | 282 days
80 | Sales Representative | 282 days
80 | Sales Representative | 282 days
80 | Sales Representative | 282 days
80 | Sales Representative | 282 days
80 | Sales Representative | 282 days
80 | Sales Representative | 282 days
80 | Sales Representative | 282 days
80 | Sales Representative | 282 days
80 | Sales Representative | 282 days
80 | Sales Representative | 282 days
(34 rows)

hr=#
```

Windows taskbar: 17°C Mostly cloudy, 23:47, 26-01-2023