Employee Database Management

Objective: Employee database is been setup which helps the organization to track employee information providing required business solutions using SQL.

Overview:

- Analyzing data to get employee information such as pay, designation, title.
- Exploring different factors that describe employees performance.

Dataset: Values have been entered using SQL query as per business requirements which can be found below.

Schema:

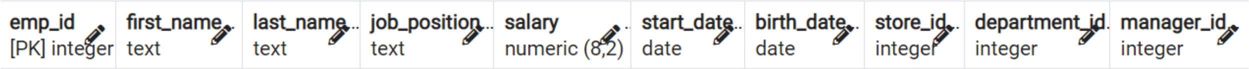
Business problems and solutions:

Task 1

Task 1.1

In your company there hasn't been a database table with all the employee information yet.

You need to set up the table called employees in the following way:



emp_id	first_name	last_name	job_position	salary	start_date	birth_date	store_id	department_id	manager_id
[PK] integer	text	text	text	numeric (8,2)	date	date	integer	integer	integer

There should be NOT NULL constraints for the following columns:

first_name,

last_name ,

job_position,

start_date DATE,

birth_date DATE

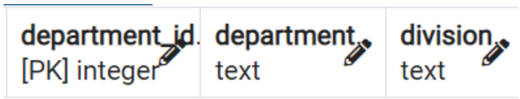
Solution:

CREATE TABLE employees (

```
emp_id SERIAL PRIMARY KEY,

first_name TEXT NOT NULL,

last_name TEXT NOT NULL,

job_position TEXT NOT NULL,

salary decimal(8,2),

start_date DATE NOT NULL,

birth_date DATE NOT NULL,

store_id INT REFERENCES store(store_id),

department_id INT,

manager_id INT

);
```

Task 1.2

Set up an additional table called departments in the following way:



department_id.	department	division
[PK] integer	text	text

Additionally no column should allow nulls.

Solution:

```
CREATE TABLE departments (

department_id SERIAL PRIMARY KEY,

department TEXT NOT NULL,

division TEXT NOT NULL);
```

Task

Alter the employees table in the following way:

- Set the column department_id to not null.

- Add a default value of CURRENT_DATE to the column start_date.

- Add the column end_date with an appropriate data type (one that you think makes sense).

- Add a constraint called birth_check that doesn't allow birth dates that are in the future.

- Rename the column job_position to position_title.

Solution:

```
ALTER TABLE employees

ALTER COLUMN department_id SET NOT NULL,

ALTER COLUMN start_date SET DEFAULT CURRENT_DATE,

ADD COLUMN end_date DATE,

ADD CONSTRAINT birth_check CHECK(birth_date < CURRENT_DATE);

ALTER TABLE employees

RENAME job_position TO position_title;
```

Task 3

Task 3.1

Insert the following values into the *employees* table.

There will be most likely an error when you try to insert the values.

So, try to insert the values and then fix the error.

Columns:

```
(emp_id,first_name,last_name,position_title,salary,start_date,birth_date,store_id,department_id,
manager_id,end_date)
```

Values:

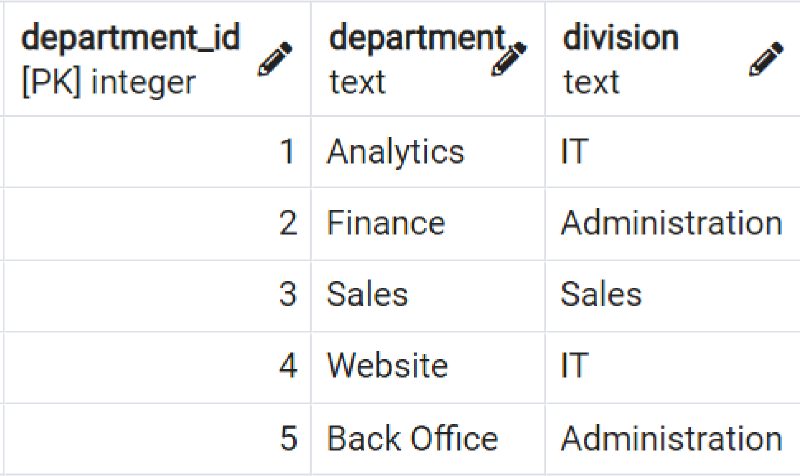
```
(1,'Morrie','Conaboy','CTO',21268.94,'2005-04-30','1983-07-10',1,1,NULL,NULL),
(2,'Miller','McQuarter','Head of BI',14614.00,'2019-07-23','1978-11-09',1,1,1,NULL),
(3,'Christalle','McKenny','Head of Sales',12587.00,'1999-02-05','1973-01-09',2,3,1,NULL),
(4,'Sumner','Seares','SQL Analyst',9515.00,'2006-05-31','1976-08-03',2,1,6,NULL),
(5,'Romain','Hacard','BI Consultant',7107.00,'2012-09-24','1984-07-14',1,1,6,NULL),
(6,'Ely','Luscombe','Team Lead Analytics',12564.00,'2002-06-12','1974-08-01',1,1,2,NULL),
(7,'Clywd','Filyashin','Senior SQL Analyst',10510.00,'2010-04-05','1989-07-23',2,1,2,NULL),
(8,'Christopher','Blague','SQL Analyst',9428.00,'2007-09-30','1990-12-07',2,2,6,NULL),
(9,'Roddie','Izen','Software Engineer',4937.00,'2019-03-22','2008-08-30',1,4,6,NULL),
(10,'Ammamaria','Izhak','Customer Support',2355.00,'2005-03-17','1974-07-27',2,5,3,2013-04-14),
(11,'Carlyn','Stripp','Customer Support',3060.00,'2013-09-06','1981-09-05',1,5,3,NULL),
(12,'Reuben','McRorie','Software Engineer',7119.00,'1995-12-31','1958-08-15',1,5,6,NULL),
(13,'Gates','Raison','Marketing Specialist',3910.00,'2013-07-18','1986-06-24',1,3,3,NULL),
(14,'Jordanna','Raitt','Marketing Specialist',5844.00,'2011-10-23','1993-03-16',2,3,3,NULL),
(15,'Guendolen','Motton','BI Consultant',8330.00,'2011-01-10','1980-10-22',2,3,6,NULL),
(16,'Doria','Turbat','Senior SQL Analyst',9278.00,'2010-08-15','1983-01-11',1,1,6,NULL),
(17,'Cort','Bewlie','Project Manager',5463.00,'2013-05-26','1986-10-05',1,5,3,NULL),
(18,'Margarita','Eaden','SQL Analyst',5977.00,'2014-09-24','1978-10-08',2,1,6,2020-03-16),
(19,'Hetty','Kingaby','SQL Analyst',7541.00,'2009-08-17','1999-04-25',1,2,6,'NULL'),
(20,'Lief','Robardley','SQL Analyst',8981.00,'2002-10-23','1971-01-25',2,3,6,2016-07-01),
(21,'Zaneta','Carlozzi','Working Student',1525.00,'2006-08-29','1995-04-16',1,3,6,2012-02-19),
(22,'Giana','Matz','Working Student',1036.00,'2016-03-18','1987-09-25',1,3,6,NULL),
(23,'Hamil','Evershed','Web Developper',3088.00,'2022-02-03','2012-03-30',1,4,2,NULL),
(24,'Lowe','Diamant','Web Developper',6418.00,'2018-12-31','2002-09-07',1,4,2,NULL),
(25,'Jack','Franklin','SQL Analyst',6771.00,'2013-05-18','2005-10-04',1,2,2,NULL),
(26,'Jessica','Brown','SQL Analyst',8566.00,'2003-10-23','1965-01-29',1,1,2,NULL)
```

Solution:

```
INSERT INTO employees
VALUES
(1,'Morrie','Conaboy','CTO',21268.94,'2005-04-30','1983-07-10',1,1,NULL,NULL),
(2,'Miller','McQuarter','Head of BI',14614.00,'2019-07-23','1978-11-09',1,1,1,NULL),
(3,'Christalle','McKenny','Head of Sales',12587.00,'1999-02-05','1973-01-09',2,3,1,NULL),
(4,'Sumner','Seares','SQL Analyst',9515.00,'2006-05-31','1976-08-03',2,1,6,NULL),
(5,'Romain','Hacard','BI Consultant',7107.00,'2012-09-24','1984-07-14',1,1,6,NULL),
(6,'Ely','Luscombe','Team Lead Analytics',12564.00,'2002-06-12','1974-08-01',1,1,2,NULL),
(7,'Clywd','Filyashin','Senior SQL Analyst',10510.00,'2010-04-05','1989-07-23',2,1,2,NULL),
(8,'Christopher','Blague','SQL Analyst',9428.00,'2007-09-30','1990-12-07',2,2,6,NULL),
(9,'Roddie','Izen','Software Engineer',4937.00,'2019-03-22','2008-08-30',1,4,6,NULL),
(10,'Ammamaria','Izhak','Customer Support',2355.00,'2005-03-17','1974-07-27',2,5,3,'2013-04-14'),
(11,'Carlyn','Stripp','Customer Support',3060.00,'2013-09-06','1981-09-05',1,5,3,NULL),
(12,'Reuben','McRorie','Software Engineer',7119.00,'1995-12-31','1958-08-15',1,5,6,NULL),
(13,'Gates','Raison','Marketing Specialist',3910.00,'2013-07-18','1986-06-24',1,3,3,NULL),
(14,'Jordanna','Raitt','Marketing Specialist',5844.00,'2011-10-23','1993-03-16',2,3,3,NULL),
(15,'Guendolen','Motton','BI Consultant',8330.00,'2011-01-10','1980-10-22',2,3,6,NULL),
(16,'Doria','Turbat','Senior SQL Analyst',9278.00,'2010-08-15','1983-01-11',1,1,6,NULL),
(17,'Cort','Bewlie','Project Manager',5463.00,'2013-05-26','1986-10-05',1,5,3,NULL),
(18,'Margarita','Eaden','SQL Analyst',5977.00,'2014-09-24','1978-10-08',2,1,6,'2020-03-16'),
(19,'Hetty','Kingaby','SQL Analyst',7541.00,'2009-08-17','1999-04-25',1,2,6,NULL),
(20,'Lief','Robardley','SQL Analyst',8981.00,'2002-10-23','1971-01-25',2,3,6,'2016-07-01'),
(21,'Zaneta','Carlozzi','Working Student',1525.00,'2006-08-29','1995-04-16',1,3,6,'2012-02-19'),
(22,'Giana','Matz','Working Student',1036.00,'2016-03-18','1987-09-25',1,3,6,NULL),
(23,'Hamil','Evershed','Web Developper',3088.00,'2022-02-03','2012-03-30',1,4,2,NULL),
(24,'Lowe','Diamant','Web Developper',6418.00,'2018-12-31','2002-09-07',1,4,2,NULL),
(25,'Jack','Franklin','SQL Analyst',6771.00,'2013-05-18','2005-10-04',1,2,2,NULL),
(26,'Jessica','Brown','SQL Analyst',8566.00,'2003-10-23','1965-01-29',1,1,2,NULL);
```

Task 3.2

Insert the following values into the departments table.



department_id [PK] integer	department text	division text
1	Analytics	IT
2	Finance	Administration
3	Sales	Sales
4	Website	IT
5	Back Office	Administration

Solution:

```
INSERT INTO departments
VALUES (1, 'Analytics','IT'),
(2, 'Finance','Administration'),
(3, 'Sales','Sales'),
(4, 'Website','IT'),
(5, 'Back Office','Administration')
```

Task 4

Task 4.1

Jack Franklin gets promoted to 'Senior SQL Analyst' and the salary raises to 7200.

Update the values accordingly.

Solution:

```
UPDATE employees
SET position_title = 'Senior SQL Analyst'
WHERE emp_id=25;
UPDATE employees
SET salary=7200
WHERE emp_id=25;
```

Task 4.2

The responsible people decided to rename the position_title Customer Support to Customer Specialist.

Update the values accordingly.

Solution:

```
UPDATE employees
SET position_title='Customer Specialist'
WHERE position_title='Customer Support';
```

Task 4.3

All SQL Analysts including Senior SQL Analysts get a raise of 6%.

Upate the salaries accordingly.

Solution:

```
UPDATE employees
SET salary=salary*1.06
WHERE position_title LIKE '%SQL Analyst';
```

Task 4.4

What is the average salary of a SQL Analyst in the company (excluding Senior SQL Analyst)?

Answer:

8834.75

Solution:

```
SELECT ROUND(AVG(salary),2) FROM employees
WHERE position_title='SQL Analyst'
```

Task 5

Task 5.1

Write a query that adds a column called manager that contains first_name and last_name (in one column) in the data output.

Secondly, add a column called is_active with 'false' if the employee has left the company already, otherwise the value is 'true'.

Solution:

```
SELECT
emp.*,
CASE WHEN emp.end_date IS NULL THEN 'true'
ELSE 'false'
END as is_active,
mng.first_name ||' '|| mng.last_name AS manager_name
FROM employees emp
LEFT JOIN employees mng
ON emp.manager_id=mng.emp_id;
```

Task 5.2

Create a view called v_employees_info from that previous query.

Solution:

```
CREATE VIEW v_employees_info
AS
SELECT
```

```
emp.*,

CASE WHEN emp.end_date IS NULL THEN 'true'

ELSE 'false'

END as is_active,

mng.first_name ||' '|| mng.last_name AS manager_name

FROM employees emp

LEFT JOIN employees mng

ON emp.manager_id=mng.emp_id;
```

Task 6

Write a query that returns the average salaries for each positions with appropriate roundings.

Question:

What is the average salary for a Software Engineer in the company.

Answer:

6028.00

Solution:

```
SELECT

position_title,

ROUND(AVG(salary),2)

FROM v_employees_info

GROUP BY position_title

ORDER BY 2;
```

Task 7

Write a query that returns the average salaries per division.

Question:

What is the average salary in the Sales department?

Answer:

6107.41

Solution:

```
SELECT

division,

ROUND(AVG(salary),2)

FROM employees e

LEFT JOIN departments d

ON e.department_id=d.department_id

GROUP BY division

ORDER BY 2
```

Task 8

Task 8.1

Write a query that returns the following:

emp_id,

first_name,

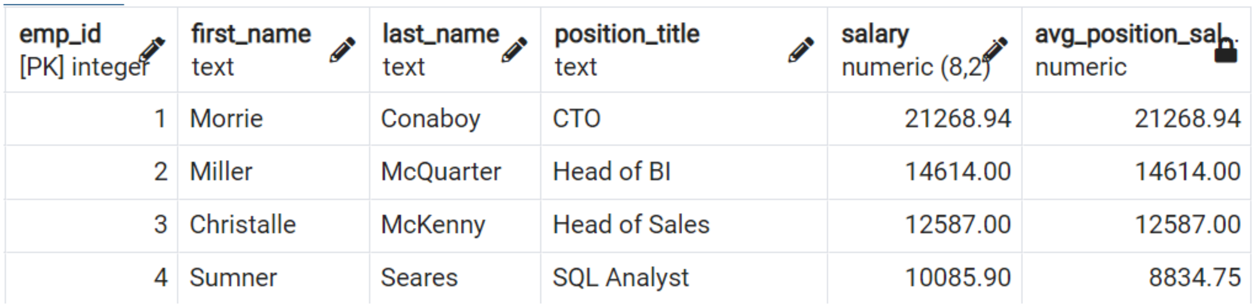
last_name,

position_title,

salary

and a column that returns the average salary for every job_position.

Order the results by the emp_id.



emp_id [PK] integer	first_name text	last_name text	position_title text	salary numeric (8,2)	avg_position_sal. numeric
1	Morrie	Conaboy	CTO	21268.94	21268.94
2	Miller	McQuarter	Head of BI	14614.00	14614.00
3	Christalle	McKenny	Head of Sales	12587.00	12587.00
4	Sumner	Seares	SQL Analyst	10085.90	8834.75

Solution:

```
SELECT
emp_id,
first_name,
last_name,
position_title,
salary,
ROUND(AVG(salary) OVER(PARTITION BY position_title),2) as avg_position_sal
FROM employees
ORDER BY 1;
```

Task 8.2

How many people earn less than there avg_position_salary?

Write a query that answers that question.

Ideally the output just shows that number directly.

Answer:

9

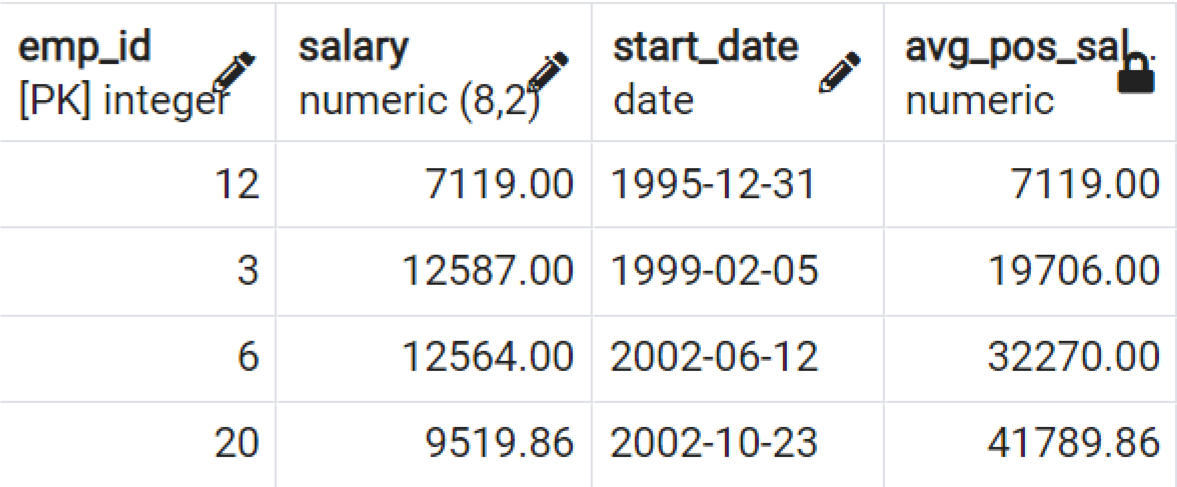
Solution:

```
SELECT

COUNT(*)

FROM (

SELECT

emp_id,

salary,
ROUND(AVG(salary) OVER(PARTITION BY position_title),2) as avg_pos_sal

FROM employees) a

WHERE salary<avg_pos_sal;
```

Task 9:

Write a query that returns a running total of the salary development ordered by the start_date.

In your calculation, disregard that fact that people have left the company (write the query as if they were still working for the company).



emp_id [PK] integer	salary numeric (8,2)	start_date date	avg_pos_sal numeric
12	7119.00	1995-12-31	7119.00
3	12587.00	1999-02-05	19706.00
6	12564.00	2002-06-12	32270.00
20	9519.86	2002-10-23	41789.86

Question:

What was the total salary after 2018-12-31?

Answer:

180202.70

Solution:

```
SELECT

emp_id,

salary,

start_date,

SUM(salary) OVER(ORDER BY start_date) as avg_pos_sal
```

```
FROM employees;
```

Task 10:

Create the same running total but now also considder the fact that people were leaving the company.

Note:

This challenge is actually very difficult.

Don't worry if you can't solve it you are not expected to do so.

It is possible to solve the challenge even without the hints.

If you want you can try to solve it using the hints and it is still a difficult challenge.

Question:

What was the total salary after 2018-12-31?

Answer:

166802.84

Hint 1:
Use the view v_employees_info.

Hint 2:
Create to separate queries one with all employees and one with the people that have already left

Hint 3:
In the first query use start_date and in the second query use end_date instead of the start_date

Hint 4:
Multiply the salary of the second query with (-1).

Hint 5:
Create a subquery from that UNION and use a window function in that to create the running total.

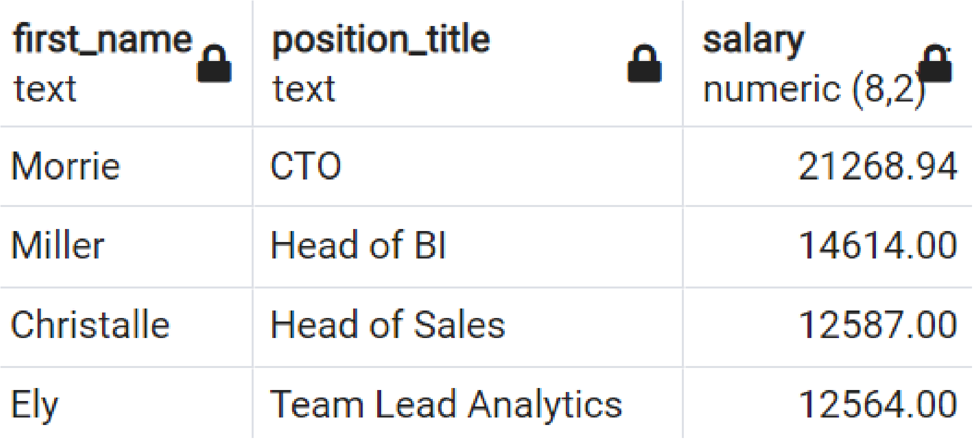
Solution:

```
SELECT

start_date,
SUM(salary) OVER(ORDER BY start_date)
FROM (
SELECT
emp_id,
salary,
start_date
FROM employees
UNION
SELECT
emp_id,
-salary,
end_date
FROM v_employees_info
WHERE is_active ='false'
ORDER BY start_date) a
```

Task 11

Task 11.1

Write a query that outputs only the top earner per position_title including first_name and position_title and their salary.



first_name text	position_title text	salary numeric (8,2)
Morrie	CTO	21268.94
Miller	Head of BI	14614.00
Christalle	Head of Sales	12587.00
Ely	Team Lead Analytics	12564.00

Question:

What is the top earner with the position_title SQL Analyst?

Answer:

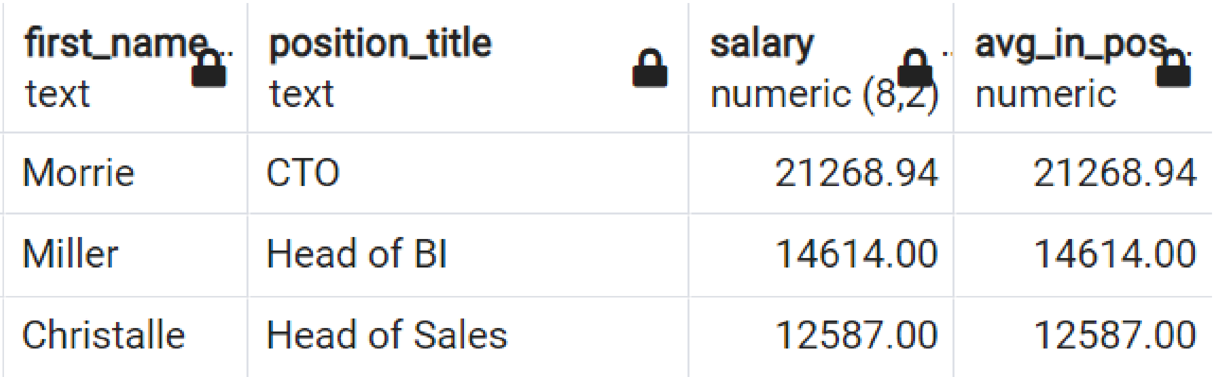
Sumner with 10085.90

Solution:

```
SELECT

first_name,

position_title,

salary

FROM employees e1

WHERE salary = (SELECT MAX(salary)

                      FROM employees e2

                      WHERE e1.position_title=e2.position_title)
```

Task 11.2

Add also the average salary per position_title.



first_name..	position_title	salary	avg_in_pos..
text	text	numeric (8,2)	numeric
Morrie	CTO	21268.94	21268.94
Miller	Head of BI	14614.00	14614.00
Christalle	Head of Sales	12587.00	12587.00

Solution:

```
SELECT
first_name,
position_title,
salary,
(SELECT ROUND(AVG(salary),2) as avg_in_pos FROM employees e3
WHERE e1.position_title=e3.position_title)
FROM employees e1
WHERE salary = (SELECT MAX(salary)
                                   FROM employees e2
                                   WHERE e1.position_title=e2.position_title)
```

Task 11.3

Remove those employees from the output of the previous query that have the same salary as the average of their position_title.

These are the people that are the only ones with their position_title.

Solution:

```
SELECT

first_name,

position_title,

salary,

(SELECT ROUND(AVG(salary),2) as avg_in_pos FROM employees e3

WHERE e1.position_title=e3.position_title)

FROM employees e1

WHERE salary = (SELECT MAX(salary)
                              FROM employees e2

                              WHERE e1.position_title=e2.position_title)

AND salary<>(SELECT ROUND(AVG(salary),2) as avg_in_pos FROM employees e3

WHERE e1.position_title=e3.position_title)
```

Task 12

Write a query that returns all meaningful aggregations of

- sum of salary,

- number of employees,

- average salary

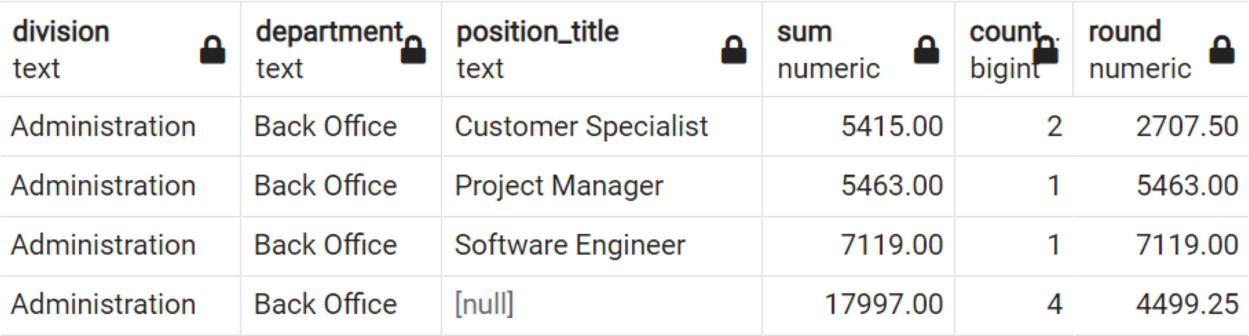
grouped by all meaningful combinations of

- division,

- department,

- position_title.

Consider the levels of hierarchies in a meaningful way.



division	department	position_title	sum	count	round
text	text	text	numeric	bigint	numeric
Administration	Back Office	Customer Specialist	5415.00	2	2707.50
Administration	Back Office	Project Manager	5463.00	1	5463.00
Administration	Back Office	Software Engineer	7119.00	1	7119.00
Administration	Back Office	[null]	17997.00	4	4499.25

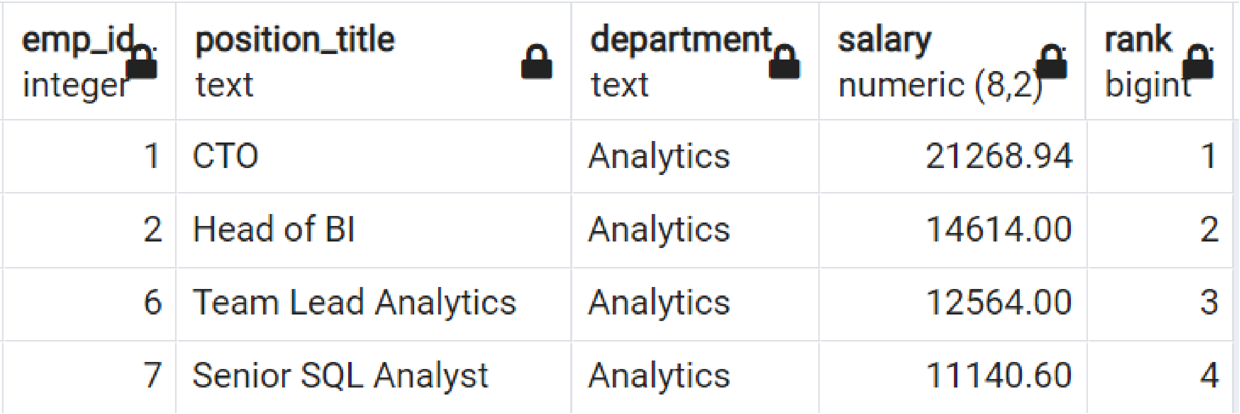
Solution:

```
SELECT

division,

department,
position_title,

SUM(salary),

COUNT(*),

ROUND(AVG(salary),2)

FROM employees

NATURAL JOIN departments

GROUP BY

ROLLUP(

division,

department,

position_title

)

ORDER BY 1,2,3
```

Task 13

Write a query that returns all employees (emp_id) including their position_title, department their salary and the rank of that salary partitioned by department.

The highest salary per division should have rank 1.



emp_id integer	position_title text	department text	salary numeric (8,2)	rank bigint
1	CTO	Analytics	21268.94	1
2	Head of BI	Analytics	14614.00	2
6	Team Lead Analytics	Analytics	12564.00	3
7	Senior SQL Analyst	Analytics	11140.60	4

Question:

Which employee (emp_id) is in rank 7 in the department Analytics?

Answer:

emp_id 26

Solution:

```
SELECT
emp_id,
position_title,
department,
salary,
RANK() OVER(PARTITION BY department ORDER BY salary DESC)
FROM employees
NATURAL LEFT JOIN departments
```

Task 14

Write a query that returns only the top earner of each department including their emp_id, position_title, department and their salary.

Question:

Which employee (emp_id) is the top earner in the department Finance?

Answer:

emp_id 8

Solution:

```
SELECT * FROM
(
SELECT
emp_id,
position_title,
department,

salary,

RANK() OVER(PARTITION BY department ORDER BY salary DESC)

FROM employees

NATURAL LEFT JOIN departments) a

WHERE rank=1
```

Findings:

- Organization is now aware of the employees pay range based on various departments, thus helping organization to understand to revise salary of employees working in different departments based on the performance.Therefore, resulting in promotion of the desired employees.