

Common Table Expressions (CTEs):

Question 1: Calculate the average salary by department for all Analysts.

```
22
23 -- Common Table Expressions (CTEs):
24 -- Question 1: Calculate the average salary by department for all Analysts.
25 With analyst_salary AS (
26     SELECT e.unit, e.salary
27     FROM employeeRecords e
28     WHERE e.designation = 'Analyst'
29 )
30 SELECT d.unit, ROUND(AVG(d.salary)) AS avg_salary
31 FROM analyst_salary d
32 GROUP BY d.unit
33
34 -- Question 2: List all employees who have used more than 10 leaves.
35
```

	unit	avg_salary
	character varying (100)	numeric
1	Operations	45187
2	Finance	44915
3	Web	45200
4	Management	44975
5	IT	44798
6	Marketing	45054

Question 2: List all employees who have used more than 10 leaves.

```
33
34 -- Question 2: List all employees who have used more than 10 leaves.
35 WITH leaver AS(
36     SELECT e.firstname, e.lastname, e.leavesused
37     FROM employeeRecords e
38     WHERE e.leavesused > 10
39 )
40 SELECT *
41 FROM leaver
42
43 -- Views:
44 -- Question 3: Create a view to show the details of all Senior Analysts.
45 -- Materialized View:
```

	firstname	lastname	leavesused
	character varying (50)	character varying (50)	integer
1	TOMASA	ARMEN	24
2	OLIVE	ANCY	23
3	CHERRY	AQUILAR	22
4	LEON	ABOULAHOU	27
5	VICTORIA	[null]	20
6	ELLIOT	AGULAR	19
7	JACQUES	AKMAL	29
8	KATHY	ALSOP	20
9	LILIAN	APELA	15
10	BELLE	ARDS	22
11	WELDON	AIVAO	15

Views:

Question 3: Create a view to show the details of all Senior Analysts.

```
42
43 -- Views:
44 -- Question 3: Create a view to show the details of all Senior Analysts.
45 CREATE VIEW senior_analyst AS
46     SELECT *
47     FROM employeeRecords e
48     WHERE e.designation = 'Senior Analyst'
49 -- Materialized Views:
50 -- Question 4: Create a materialized view to store the count of employees by department.
51
52 -- Procedures (Stored Procedures):
53 -- Question 5: Create a procedure to update an employee's salary by their first name and
```

Data Output Explain Messages Notifications

CREATE VIEW

Query returned successfully in 196 msec.

```
1 SELECT * FROM public.senior_analyst
2
```

	id	firstname	lastname	sex	doj	currentdate	designation	age	salary	unit
	integer	character varying (50)	character varying (50)	character (1)	date	date	character varying (100)	integer	numeric (10,2)	character varying (50)
1	9	KATHY	ALSOP	F	2014-06...	2016-01-07	Senior Analyst	28	63478.00	Operations
2	29	SEYMOUR	ALBEN	M	2014-12...	2016-01-07	Senior Analyst	25	57488.00	Operations
3	33	FOSTER	ALDERMAN	M	2014-05...	2016-01-07	Senior Analyst	26	68295.00	Operations
4	54	CARI	ARENALES	F	2014-04...	2016-01-07	Senior Analyst	28	66338.00	Web
5	58	PAULINE	ALTSHULER	F	2014-12...	2016-01-07	Senior Analyst	28	61647.00	Finance
6	69	RILEY	AIKINS	M	2013-06...	2016-01-07	Senior Analyst	25	60712.00	Finance
7	73	MARYJANE	ARES	F	2012-08...	2016-01-07	Senior Analyst	25	65212.00	Manageme
8	76	MARY	ALMESTICA	F	2013-10...	2016-01-07	Senior Analyst	27	53339.00	Finance
9	83	WILMER	AKIONA	M	2014-05...	2016-01-07	Senior Analyst	25	50739.00	IT
10	91	ELOISA	ARGIE	F	2013-08...	2016-01-07	Senior Analyst	28	52690.00	Marketing
11	93	KATELYN	APPENZELLER	F	2014-11...	2016-01-07	Senior Analyst	28	56314.00	Finance
12	103	ELIZA	ANSBACHER	F	2013-09...	2016-01-07	Senior Analyst	25	50813.00	Marketing
13	105	CARI	ARENALES	F	2014-04...	2016-01-07	Senior Analyst	28	66338.00	Web
14	116	ELEANORE	ARENA	F	2013-03...	2016-01-07	Senior Analyst	26	62537.00	Manageme
15	121	KENDRICK	AHIMUD	M	2013-04...	2016-01-07	Senior Analyst	26	68245.00	Finance
16	125	TERESITA	ARENTZ	F	2013-11...	2016-01-07	Senior Analyst	27	53275.00	Finance
17	138	JON	ABOLT	M	2013-02...	2016-01-07	Senior Analyst	25	50130.00	Marketing
18	139	JORDON	ALLARA	M	2014-12...	2016-01-07	Senior Analyst	25	55472.00	Finance
19	143	DEIDRA	ARCHANGEL	F	2012-10...	2016-01-07	Senior Analyst	28	69316.00	Operations

Materialized Views:

Question 4: Create a materialized view to store the count of employees by department.

```
49
50 -- Materialized Views:
51 -- Question 4: Create a materialized view to store the count of employees by department.
52 CREATE MATERIALIZED VIEW employee_count_by_department AS
53     SELECT e.unit AS Department, COUNT(e.id)
54     FROM employeeRecords e
55     GROUP BY Department
56
57 -- Procedures (Stored Procedures):
58 -- Question 6: Create a procedure to update an employee's salary by their first name and last name.
59 -- Question 7: Create a procedure to calculate the total number of leaves used across all departments.
60
```

Data Output Explain Messages Notifications

SELECT 6

Query returned successfully in 154 msec.

Query Editor Query History

```
1 SELECT * FROM public.employee_count_by_department
2
```

Data Output Explain Messages Notifications

	department character varying (100)	count bigint
1	Operations	438
2	Finance	446
3	Web	431
4	Management	425
5	IT	461
6	Marketing	438

Procedures (Stored Procedures):

Question 6: Create a procedure to update an employee's salary by their first name and last name.

```
56
57 -- Procedures (Stored Procedures):
58 -- Question 6: Create a procedure to update an employee's salary by their first name and last name.
59 CREATE OR REPLACE PROCEDURE update_employee_salary(
60     fname VARCHAR(50),
61     lname VARCHAR(50),
62     new_salary DECIMAL(10,2))
63 LANGUAGE plpgsql
64 AS $$
65 BEGIN
66     UPDATE employeeRecords
67     SET salary = new_salary
68     WHERE firstname = fname AND lastname = lname;
69 END;
70 $$;
71
72 --call the procedure
73
```

Data Output Explain Messages Notifications

CREATE PROCEDURE

Query returned successfully in 101 msec.

```
72 --call the procedure
73 CALL update_employee_salary('TOMASA','ARMEN',10);
74
75 --check the updated table
76 SELECT firstname, lastname, salary
77 FROM employeeRecords
78 ORDER BY id
79
80 -- Question 7: Create a procedure to calculate the total number of leaves used across all departments.
81 CREATE OR REPLACE PROCEDURE display_total_leaves()
82 RETURNS int
83
```

Data Output Explain Messages Notifications

	firstname character varying (50)	lastname character varying (50)	salary numeric (10,2)
1	TOMASA	ARMEN	10.00
2	ANNIE	[null]	89207.00
3	OLIVE	ANCY	40955.00
4	CHERRY	AQUILAR	45550.00
5	LEON	ABOULAHOU	43161.00
6	VICTORIA	[null]	48736.00
7	ELLIOT	AGULAR	40339.00

Successfully run. Total

Question 7: Create a procedure to calculate the total number of leaves used across all departments.

```
79
80 -- Question 7: Create a procedure to calculate the total number of leaves used across all departments.
81 CREATE OR REPLACE PROCEDURE total_leaves()
82 LANGUAGE plpgsql
83 AS $$
84 BEGIN
85     CREATE OR REPLACE VIEW total_leaves AS
86     SELECT SUM(leavesused)
87     FROM employeeRecords;
88 END;
89 $$;
90
91 --call the procedure
92 CALL total_leaves()
93
94 --display results
```

Data Output Explain Messages Notifications

CREATE PROCEDURE

Query returned successfully in 98 msec.

```
90
91 --call the procedure
92 CALL total_leaves()
93
94 --display results
95 SELECT *
96 FROM total_leaves
```

Data Output Explain Messages Notifications

	sum bigint	
1	59314	