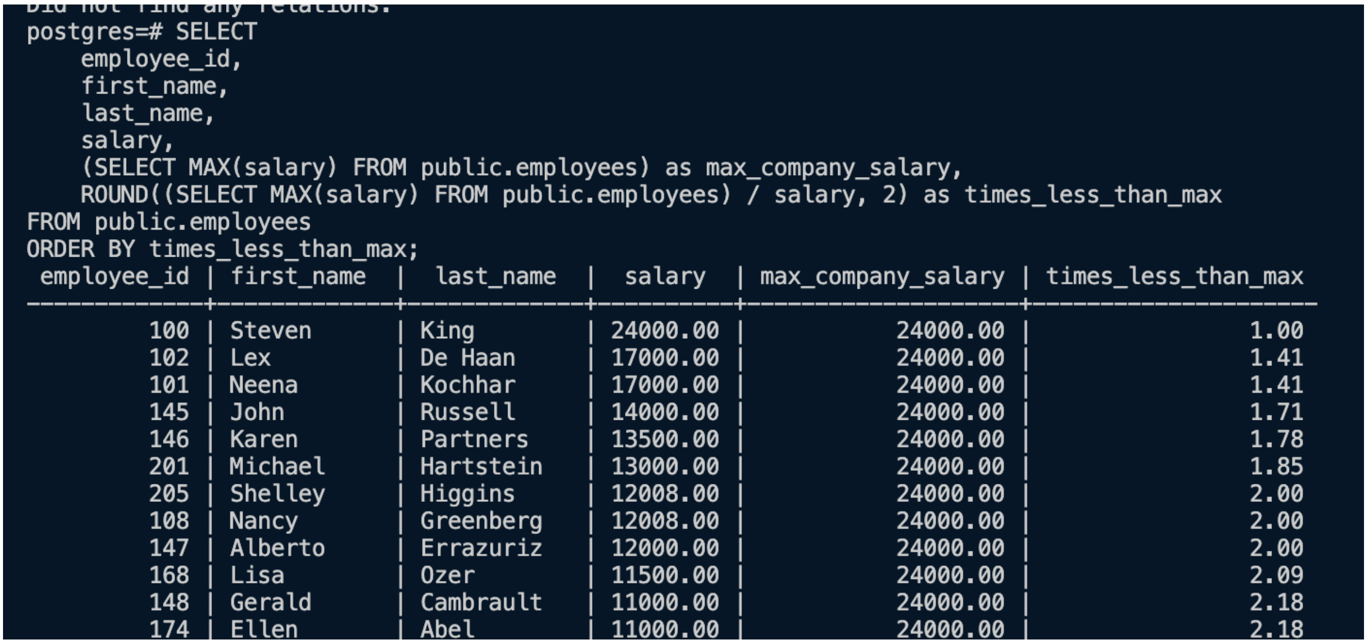
ДЗ 7

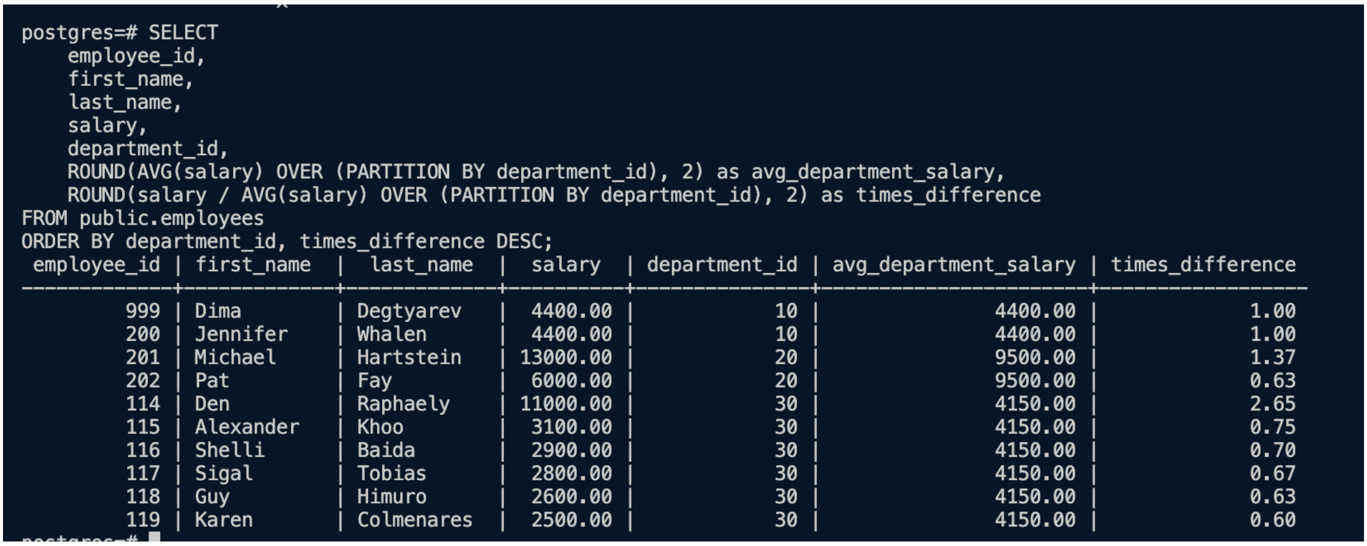
```
SELECT
    employee_id,
    first_name,
    last_name,
    salary,
    (SELECT MAX(salary) FROM public.employees) as max_company_salary,
    ROUND((SELECT MAX(salary) FROM public.employees) / salary, 2) as
times_less_than_max
FROM public.employees
ORDER BY times_less_than_max;
```



```
Did not find any relations.
postgres=# SELECT
    employee_id,
    first_name,
    last_name,
    salary,
    (SELECT MAX(salary) FROM public.employees) as max_company_salary,
    ROUND((SELECT MAX(salary) FROM public.employees) / salary, 2) as times_less_than_max
FROM public.employees
ORDER BY times_less_than_max;
 employee_id | first_name  |  last_name  |  salary  | max_company_salary | times_less_than_max
-------------+-------------+-------------+----------+--------------------+---------------------
         100 | Steven      | King        | 24000.00 |           24000.00 |                1.00
         102 | Lex         | De Haan     | 17000.00 |           24000.00 |                1.41
         101 | Neena       | Kochhar     | 17000.00 |           24000.00 |                1.41
         145 | John        | Russell     | 14000.00 |           24000.00 |                1.71
         146 | Karen       | Partners    | 13500.00 |           24000.00 |                1.78
         201 | Michael     | Hartstein   | 13000.00 |           24000.00 |                1.85
         205 | Shelley     | Higgins     | 12008.00 |           24000.00 |                2.00
         108 | Nancy       | Greenberg   | 12008.00 |           24000.00 |                2.00
         147 | Alberto     | Errazuriz   | 12000.00 |           24000.00 |                2.00
         168 | Lisa        | Ozer        | 11500.00 |           24000.00 |                2.09
         148 | Gerald      | Cambrault   | 11000.00 |           24000.00 |                2.18
         174 | Ellen       | Abel        | 11000.00 |           24000.00 |                2.18
```

2

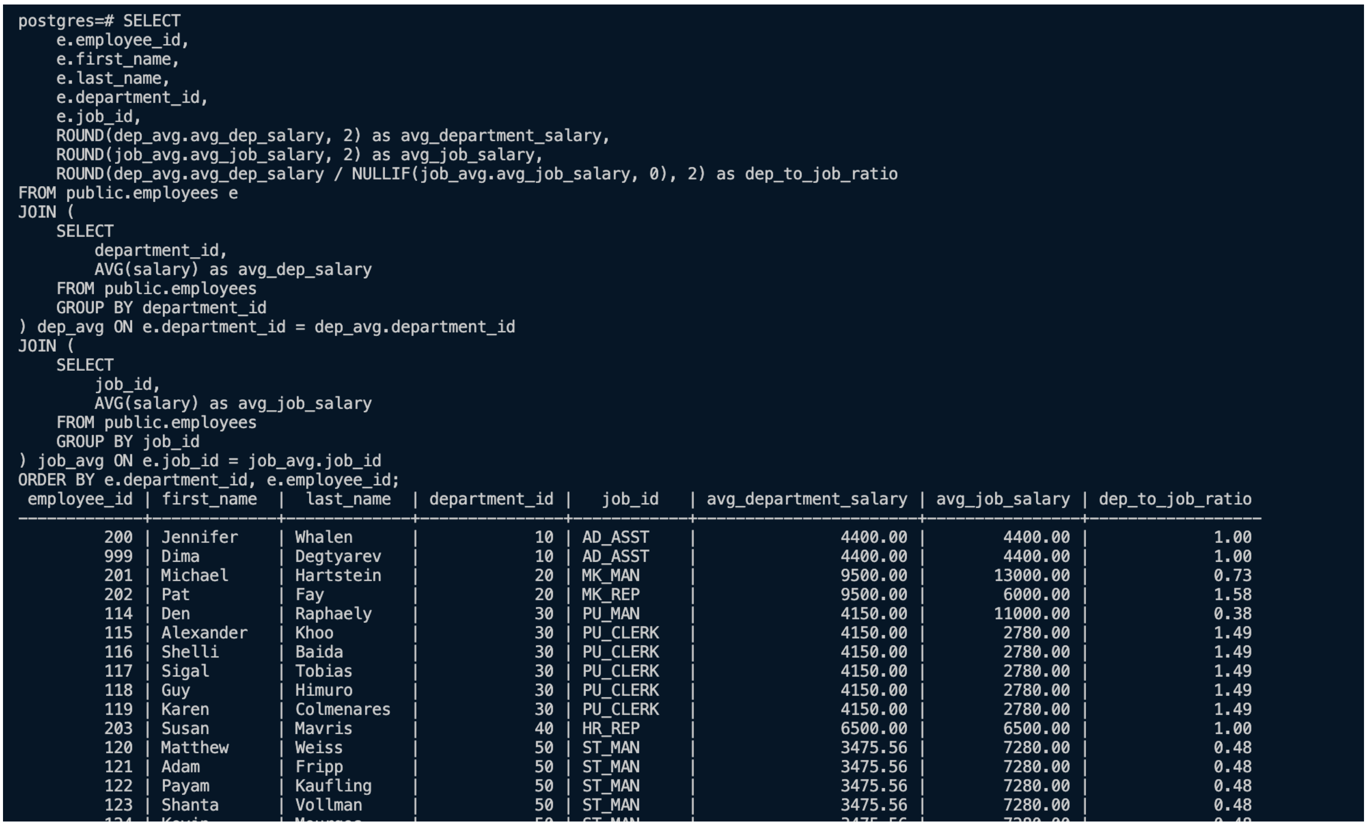
```
SELECT
    employee_id,
    first_name,
    last_name,
    salary,
    department_id,
    ROUND(AVG(salary) OVER (PARTITION BY department_id), 2) as
avg_department_salary,
    ROUND(salary / AVG(salary) OVER (PARTITION BY department_id), 2) as
times_difference
FROM public.employees
ORDER BY department_id, times_difference DESC;
```



```
postgres=# SELECT
    employee_id,
    first_name,
    last_name,
    salary,
    department_id,
    ROUND(AVG(salary) OVER (PARTITION BY department_id), 2) as avg_department_salary,
    ROUND(salary / AVG(salary) OVER (PARTITION BY department_id), 2) as times_difference
FROM public.employees
ORDER BY department_id, times_difference DESC;
 employee_id | first_name  |  last_name  |  salary  | department_id | avg_department_salary | times_difference
-------------+-------------+-------------+----------+---------------+-----------------------+------------------
         999 | Dima        | Degtyarev   |  4400.00 |            10 |               4400.00 |             1.00
         200 | Jennifer    | Whalen      |  4400.00 |            10 |               4400.00 |             1.00
         201 | Michael     | Hartstein   | 13000.00 |            20 |               9500.00 |             1.37
         202 | Pat         | Fay         |  6000.00 |            20 |               9500.00 |             0.63
         114 | Den         | Raphaely    | 11000.00 |            30 |               4150.00 |             2.65
         115 | Alexander   | Khoo        |  3100.00 |            30 |               4150.00 |             0.75
         116 | Shelli      | Baida       |  2900.00 |            30 |               4150.00 |             0.70
         117 | Sigal       | Tobias      |  2800.00 |            30 |               4150.00 |             0.67
         118 | Guy         | Himuro      |  2600.00 |            30 |               4150.00 |             0.63
         119 | Karen       | Colmenares  |  2500.00 |            30 |               4150.00 |             0.60
postgres=#
```

3

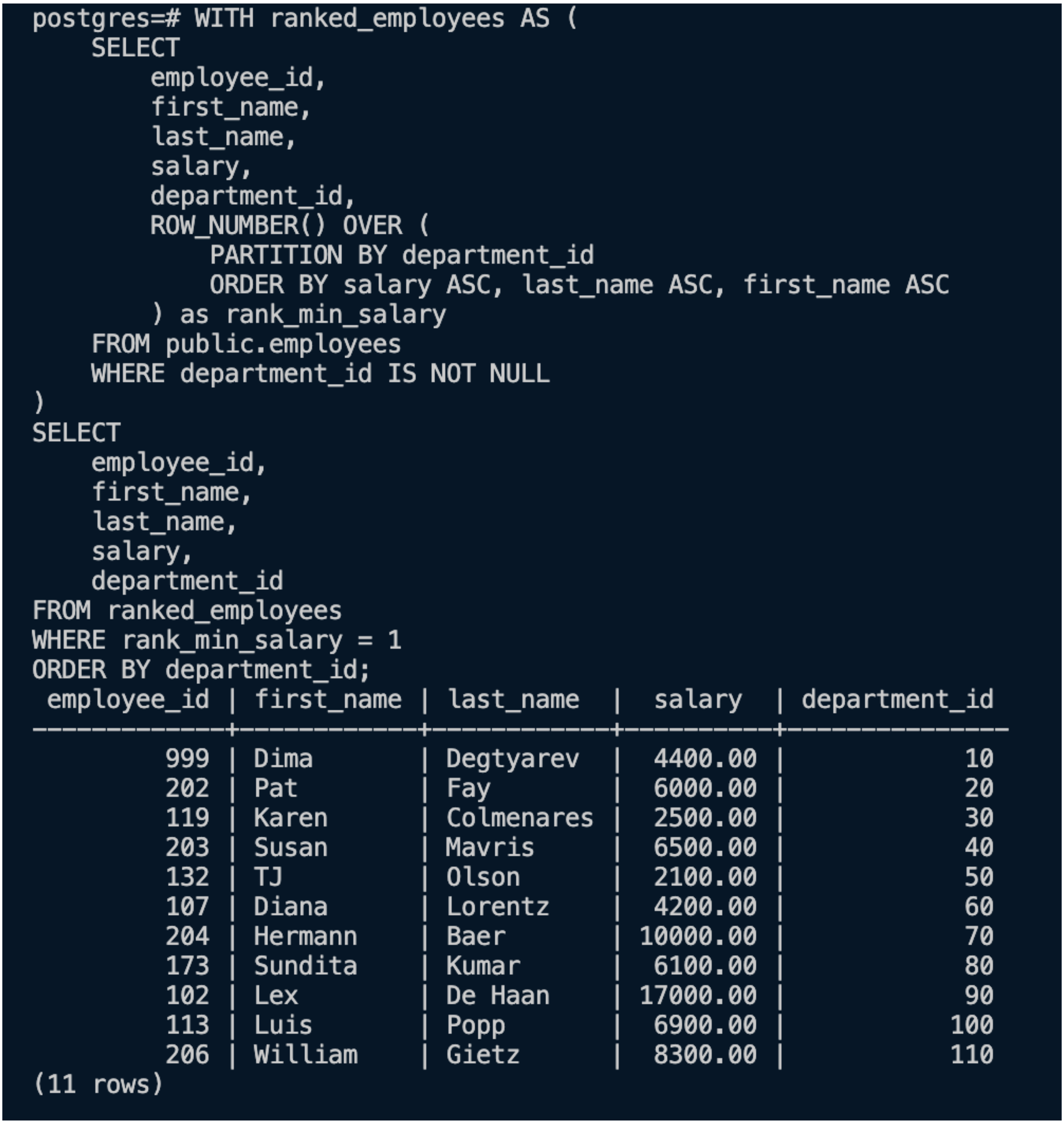
```sql
SELECT
    e.employee_id,
    e.first_name,
    e.last_name,
    e.department_id,
    e.job_id,
    ROUND(dep_avg.avg_dep_salary, 2) as avg_department_salary,
    ROUND(job_avg.avg_job_salary, 2) as avg_job_salary,
    ROUND(dep_avg.avg_dep_salary / NULLIF(job_avg.avg_job_salary, 0), 2)
as dep_to_job_ratio
FROM public.employees e
JOIN (
    SELECT
        department_id,
        AVG(salary) as avg_dep_salary
    FROM public.employees
    GROUP BY department_id
) dep_avg ON e.department_id = dep_avg.department_id
JOIN (
    SELECT
        job_id,
        AVG(salary) as avg_job_salary
    FROM public.employees
    GROUP BY job_id
) job_avg ON e.job_id = job_avg.job_id
ORDER BY e.department_id, e.employee_id;
```



```
postgres=# SELECT
    e.employee_id,
    e.first_name,
    e.last_name,
    e.department_id,
    e.job_id,
    ROUND(dep_avg.avg_dep_salary, 2) as avg_department_salary,
    ROUND(job_avg.avg_job_salary, 2) as avg_job_salary,
    ROUND(dep_avg.avg_dep_salary / NULLIF(job_avg.avg_job_salary, 0), 2) as dep_to_job_ratio
FROM public.employees e
JOIN (
    SELECT
        department_id,
        AVG(salary) as avg_dep_salary
    FROM public.employees
    GROUP BY department_id
) dep_avg ON e.department_id = dep_avg.department_id
JOIN (
    SELECT
        job_id,
        AVG(salary) as avg_job_salary
    FROM public.employees
    GROUP BY job_id
) job_avg ON e.job_id = job_avg.job_id
ORDER BY e.department_id, e.employee_id;
 employee_id | first_name  |  last_name  | department_id |   job_id   | avg_department_salary | avg_job_salary | dep_to_job_ratio
-------------+-------------+-------------+---------------+------------+-----------------------+----------------+------------------
         200 | Jennifer    | Whalen      |            10 | AD_ASST    |               4400.00 |        4400.00 |             1.00
         999 | Dima        | Degtyarev   |            10 | AD_ASST    |               4400.00 |        4400.00 |             1.00
         201 | Michael     | Hartstein   |            20 | MK_MAN     |               9500.00 |       13000.00 |             0.73
         202 | Pat         | Fay         |            20 | MK_REP     |               9500.00 |        6000.00 |             1.58
         114 | Den         | Raphaely    |            30 | PU_MAN     |               4150.00 |       11000.00 |             0.38
         115 | Alexander   | Khoo        |            30 | PU_CLERK   |               4150.00 |        2780.00 |             1.49
         116 | Shelli      | Baida       |            30 | PU_CLERK   |               4150.00 |        2780.00 |             1.49
         117 | Sigal       | Tobias      |            30 | PU_CLERK   |               4150.00 |        2780.00 |             1.49
         118 | Guy         | Himuro      |            30 | PU_CLERK   |               4150.00 |        2780.00 |             1.49
         119 | Karen       | Colmenares  |            30 | PU_CLERK   |               4150.00 |        2780.00 |             1.49
         203 | Susan       | Mavris      |            40 | HR_REP     |               6500.00 |        6500.00 |             1.00
         120 | Matthew     | Weiss       |            50 | ST_MAN     |               3475.56 |        7280.00 |             0.48
         121 | Adam        | Fripp       |            50 | ST_MAN     |               3475.56 |        7280.00 |             0.48
         122 | Payam       | Kaufling    |            50 | ST_MAN     |               3475.56 |        7280.00 |             0.48
         123 | Shanta      | Vollman     |            50 | ST_MAN     |               3475.56 |        7280.00 |             0.48
         124 | Kevin       | Mourgos     |            50 | ST_MAN     |               3475.56 |        7280.00 |             0.48
```

4

```
WITH ranked_employees AS (
    SELECT
        employee_id,
        first_name,
        last_name,
        salary,
        department_id,
        ROW_NUMBER() OVER (
            PARTITION BY department_id
            ORDER BY salary ASC, last_name ASC, first_name ASC
        ) as rank_min_salary
    FROM public.employees
    WHERE department_id IS NOT NULL
)
SELECT
    employee_id,
    first_name,
    last_name,
    salary,
    department_id
FROM ranked_employees
WHERE rank_min_salary = 1
ORDER BY department_id;
```



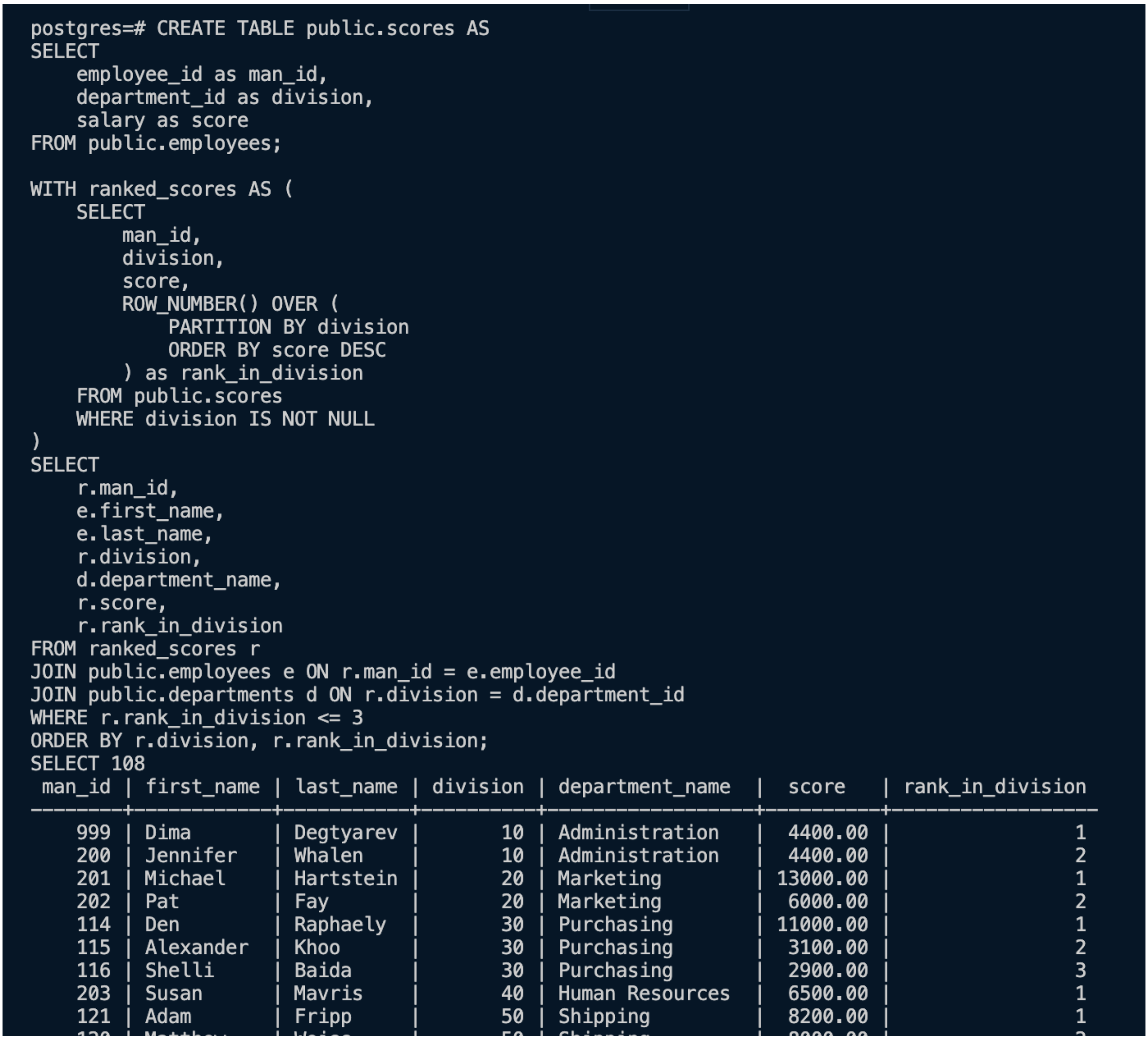
```
postgres=# WITH ranked_employees AS (
    SELECT
        employee_id,
        first_name,
        last_name,
        salary,
        department_id,
        ROW_NUMBER() OVER (
            PARTITION BY department_id
            ORDER BY salary ASC, last_name ASC, first_name ASC
        ) as rank_min_salary
    FROM public.employees
    WHERE department_id IS NOT NULL
)
SELECT
    employee_id,
    first_name,
    last_name,
    salary,
    department_id
FROM ranked_employees
WHERE rank_min_salary = 1
ORDER BY department_id;
 employee_id | first_name | last_name  |  salary  | department_id
-------------+------------+------------+----------+---------------
         999 | Dima       | Degtyarev  |  4400.00 |            10
         202 | Pat        | Fay        |  6000.00 |            20
         119 | Karen      | Colmenares |  2500.00 |            30
         203 | Susan      | Mavris     |  6500.00 |            40
         132 | TJ         | Olson      |  2100.00 |            50
         107 | Diana      | Lorentz    |  4200.00 |            60
         204 | Hermann    | Baer       | 10000.00 |            70
         173 | Sundita    | Kumar      |  6100.00 |            80
         102 | Lex        | De Haan    | 17000.00 |            90
         113 | Luis       | Popp       |  6900.00 |           100
         206 | William    | Gietz      |  8300.00 |           110
(11 rows)
```

5

```
CREATE TABLE public.scores AS
SELECT
    employee_id as man_id,
    department_id as division,
    salary as score
FROM public.employees;

WITH ranked_scores AS (
    SELECT
        man_id,
        division,
```

```sql
        score,
        ROW_NUMBER() OVER (
            PARTITION BY division
            ORDER BY score DESC
        ) as rank_in_division
    FROM public.scores
    WHERE division IS NOT NULL
)
SELECT
    r.man_id,
    e.first_name,
    e.last_name,
    r.division,
    d.department_name,
    r.score,
    r.rank_in_division
FROM ranked_scores r
JOIN public.employees e ON r.man_id = e.employee_id
JOIN public.departments d ON r.division = d.department_id
WHERE r.rank_in_division <= 3
ORDER BY r.division, r.rank_in_division;
```



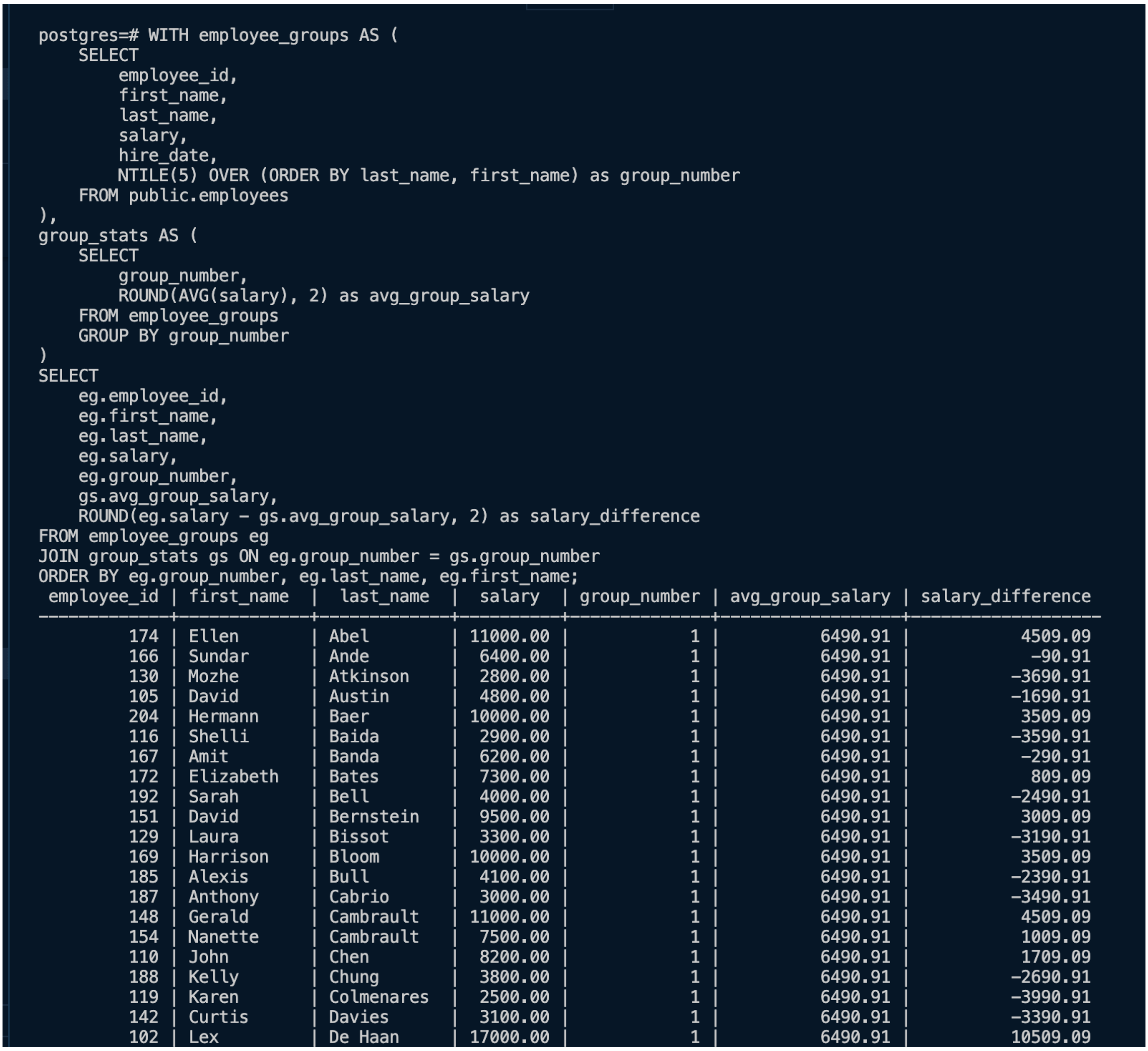
```
postgres=# CREATE TABLE public.scores AS
SELECT
    employee_id as man_id,
    department_id as division,
    salary as score
FROM public.employees;

WITH ranked_scores AS (
    SELECT
        man_id,
        division,
        score,
        ROW_NUMBER() OVER (
            PARTITION BY division
            ORDER BY score DESC
        ) as rank_in_division
    FROM public.scores
    WHERE division IS NOT NULL
)
SELECT
    r.man_id,
    e.first_name,
    e.last_name,
    r.division,
    d.department_name,
    r.score,
    r.rank_in_division
FROM ranked_scores r
JOIN public.employees e ON r.man_id = e.employee_id
JOIN public.departments d ON r.division = d.department_id
WHERE r.rank_in_division <= 3
ORDER BY r.division, r.rank_in_division;
SELECT 108
 man_id | first_name | last_name | division | department_name |  score   | rank_in_division
--------+------------+-----------+----------+-----------------+----------+------------------
    999 | Dima       | Degtyarev |       10 | Administration  |  4400.00 |                1
    200 | Jennifer   | Whalen    |       10 | Administration  |  4400.00 |                2
    201 | Michael    | Hartstein |       20 | Marketing       | 13000.00 |                1
    202 | Pat        | Fay       |       20 | Marketing       |  6000.00 |                2
    114 | Den        | Raphaely  |       30 | Purchasing      | 11000.00 |                1
    115 | Alexander  | Khoo      |       30 | Purchasing      |  3100.00 |                2
    116 | Shelli     | Baida     |       30 | Purchasing      |  2900.00 |                3
    203 | Susan      | Mavris    |       40 | Human Resources |  6500.00 |                1
    121 | Adam       | Fripp     |       50 | Shipping        |  8200.00 |                1
```

6

```
WITH employee_groups AS (
    SELECT
        employee_id,
        first_name,
        last_name,
        salary,
        hire_date,
        NTILE(5) OVER (ORDER BY last_name, first_name) as group_number
    FROM public.employees
),
group_stats AS (
    SELECT
        group_number,
        ROUND(AVG(salary), 2) as avg_group_salary
    FROM employee_groups
    GROUP BY group_number
)
```

```
SELECT
    eg.employee_id,
    eg.first_name,
    eg.last_name,
    eg.salary,
    eg.group_number,
    gs.avg_group_salary,
    ROUND(eg.salary - gs.avg_group_salary, 2) as salary_difference
FROM employee_groups eg
JOIN group_stats gs ON eg.group_number = gs.group_number
ORDER BY eg.group_number, eg.last_name, eg.first_name;
```

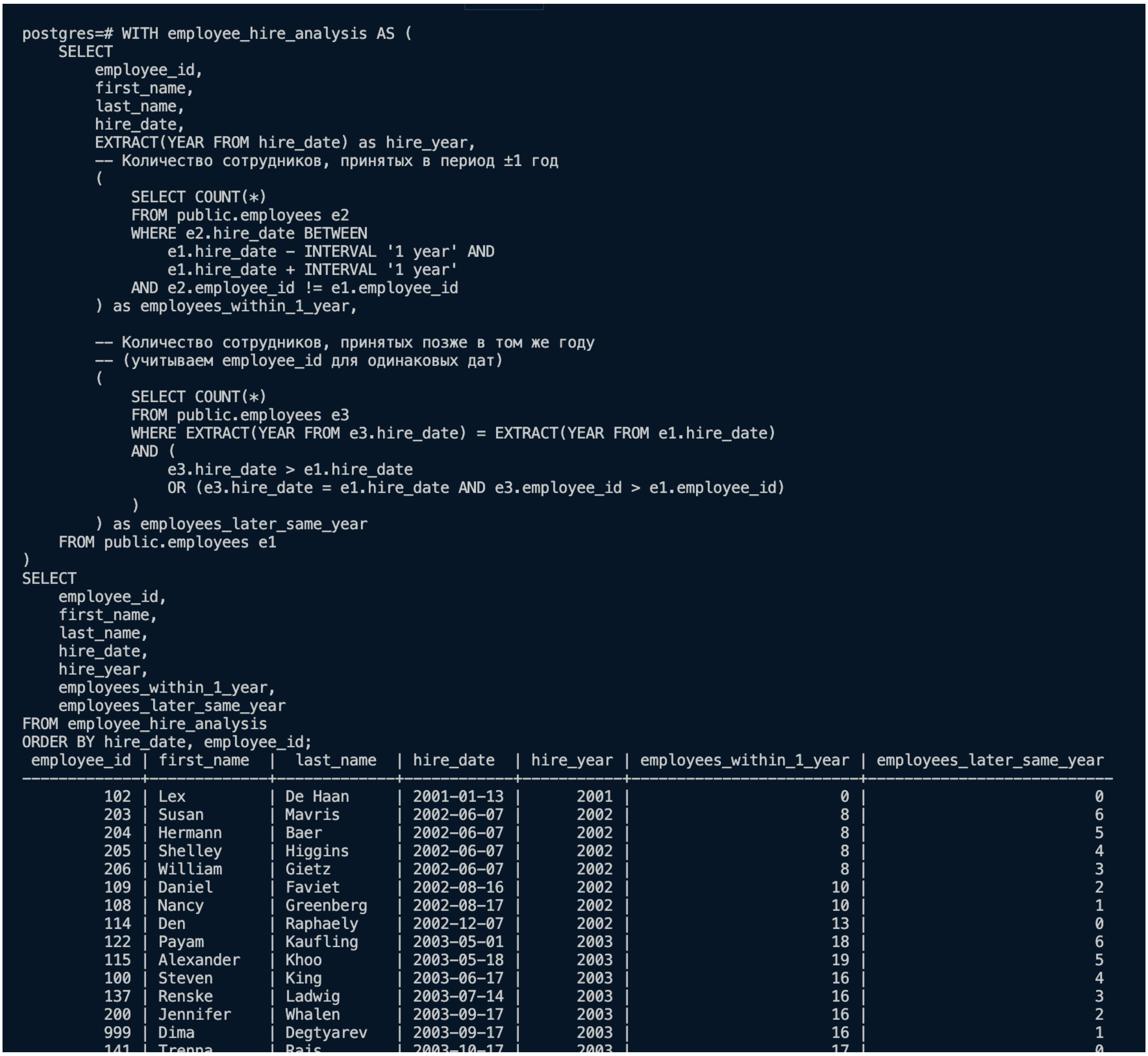


```
postgres=# WITH employee_groups AS (
    SELECT
        employee_id,
        first_name,
        last_name,
        salary,
        hire_date,
        NTILE(5) OVER (ORDER BY last_name, first_name) as group_number
    FROM public.employees
),
group_stats AS (
    SELECT
        group_number,
        ROUND(AVG(salary), 2) as avg_group_salary
    FROM employee_groups
    GROUP BY group_number
)
SELECT
    eg.employee_id,
    eg.first_name,
    eg.last_name,
    eg.salary,
    eg.group_number,
    gs.avg_group_salary,
    ROUND(eg.salary - gs.avg_group_salary, 2) as salary_difference
FROM employee_groups eg
JOIN group_stats gs ON eg.group_number = gs.group_number
ORDER BY eg.group_number, eg.last_name, eg.first_name;
 employee_id | first_name  |  last_name  |  salary  | group_number | avg_group_salary | salary_difference
-------------+-------------+-------------+----------+--------------+------------------+-------------------
         174 | Ellen       | Abel        | 11000.00 |            1 |          6490.91 |           4509.09
         166 | Sundar      | Ande        |  6400.00 |            1 |          6490.91 |            -90.91
         130 | Mozhe       | Atkinson    |  2800.00 |            1 |          6490.91 |          -3690.91
         105 | David       | Austin      |  4800.00 |            1 |          6490.91 |          -1690.91
         204 | Hermann     | Baer        | 10000.00 |            1 |          6490.91 |           3509.09
         116 | Shelli      | Baida       |  2900.00 |            1 |          6490.91 |          -3590.91
         167 | Amit        | Banda       |  6200.00 |            1 |          6490.91 |           -290.91
         172 | Elizabeth   | Bates       |  7300.00 |            1 |          6490.91 |            809.09
         192 | Sarah       | Bell        |  4000.00 |            1 |          6490.91 |          -2490.91
         151 | David       | Bernstein   |  9500.00 |            1 |          6490.91 |           3009.09
         129 | Laura       | Bissot      |  3300.00 |            1 |          6490.91 |          -3190.91
         169 | Harrison    | Bloom       | 10000.00 |            1 |          6490.91 |           3509.09
         185 | Alexis      | Bull        |  4100.00 |            1 |          6490.91 |          -2390.91
         187 | Anthony     | Cabrio      |  3000.00 |            1 |          6490.91 |          -3490.91
         148 | Gerald      | Cambrault   | 11000.00 |            1 |          6490.91 |           4509.09
         154 | Nanette     | Cambrault   |  7500.00 |            1 |          6490.91 |           1009.09
         110 | John        | Chen        |  8200.00 |            1 |          6490.91 |           1709.09
         188 | Kelly       | Chung       |  3800.00 |            1 |          6490.91 |          -2690.91
         119 | Karen       | Colmenares  |  2500.00 |            1 |          6490.91 |          -3990.91
         142 | Curtis      | Davies      |  3100.00 |            1 |          6490.91 |          -3390.91
         102 | Lex         | De Haan     | 17000.00 |            1 |          6490.91 |          10509.09
```

7

```
WITH employee_hire_analysis AS (
    SELECT
        employee_id,
        first_name,
        last_name,
        hire_date,
        EXTRACT(YEAR FROM hire_date) as hire_year,
        -- Количество сотрудников, принятых в период ±1 год
        (
            SELECT COUNT(*)
            FROM public.employees e2
            WHERE e2.hire_date BETWEEN
                e1.hire_date - INTERVAL '1 year' AND
                e1.hire_date + INTERVAL '1 year'
            AND e2.employee_id != e1.employee_id
        ) as employees_within_1_year,

        -- Количество сотрудников, принятых позже в том же году
        -- (учитываем employee_id для одинаковых дат)
        (
            SELECT COUNT(*)
            FROM public.employees e3
            WHERE EXTRACT(YEAR FROM e3.hire_date) = EXTRACT(YEAR FROM
e1.hire_date)
            AND (
                e3.hire_date > e1.hire_date
                OR (e3.hire_date = e1.hire_date AND e3.employee_id >
e1.employee_id)
            )
        ) as employees_later_same_year
    FROM public.employees e1
)
SELECT
    employee_id,
    first_name,
    last_name,
    hire_date,
    hire_year,
    employees_within_1_year,
    employees_later_same_year
FROM employee_hire_analysis
ORDER BY hire_date, employee_id;
```



```
postgres=# WITH employee_hire_analysis AS (
    SELECT
        employee_id,
        first_name,
        last_name,
        hire_date,
        EXTRACT(YEAR FROM hire_date) as hire_year,
        -- Количество сотрудников, принятых в период ±1 год
        (
            SELECT COUNT(*)
            FROM public.employees e2
            WHERE e2.hire_date BETWEEN
                e1.hire_date - INTERVAL '1 year' AND
                e1.hire_date + INTERVAL '1 year'
            AND e2.employee_id != e1.employee_id
        ) as employees_within_1_year,

        -- Количество сотрудников, принятых позже в том же году
        -- (учитываем employee_id для одинаковых дат)
        (
            SELECT COUNT(*)
            FROM public.employees e3
            WHERE EXTRACT(YEAR FROM e3.hire_date) = EXTRACT(YEAR FROM e1.hire_date)
            AND (
                e3.hire_date > e1.hire_date
                OR (e3.hire_date = e1.hire_date AND e3.employee_id > e1.employee_id)
            )
        ) as employees_later_same_year
    FROM public.employees e1
)
SELECT
    employee_id,
    first_name,
    last_name,
    hire_date,
    hire_year,
    employees_within_1_year,
    employees_later_same_year
FROM employee_hire_analysis
ORDER BY hire_date, employee_id;
 employee_id | first_name  |  last_name  | hire_date  | hire_year | employees_within_1_year | employees_later_same_year
-------------+-------------+-------------+------------+-----------+-------------------------+---------------------------
         102 | Lex         | De Haan     | 2001-01-13 |      2001 |                       0 |                         0
         203 | Susan       | Mavris      | 2002-06-07 |      2002 |                       8 |                         6
         204 | Hermann     | Baer        | 2002-06-07 |      2002 |                       8 |                         5
         205 | Shelley     | Higgins     | 2002-06-07 |      2002 |                       8 |                         4
         206 | William     | Gietz       | 2002-06-07 |      2002 |                       8 |                         3
         109 | Daniel      | Faviet      | 2002-08-16 |      2002 |                      10 |                         2
         108 | Nancy       | Greenberg   | 2002-08-17 |      2002 |                      10 |                         1
         114 | Den         | Raphaely    | 2002-12-07 |      2002 |                      13 |                         0
         122 | Payam       | Kaufling    | 2003-05-01 |      2003 |                      18 |                         6
         115 | Alexander   | Khoo        | 2003-05-18 |      2003 |                      19 |                         5
         100 | Steven      | King        | 2003-06-17 |      2003 |                      16 |                         4
         137 | Renske      | Ladwig      | 2003-07-14 |      2003 |                      16 |                         3
         200 | Jennifer    | Whalen      | 2003-09-17 |      2003 |                      16 |                         2
         999 | Dima        | Degtyarev   | 2003-09-17 |      2003 |                      16 |                         1
         141 | Trenna      | Rais        | 2003-10-17 |      2003 |                      17 |                         0
```