

Министерство науки и высшего образования Российской  
Федерации  
Российский химико-технологический университет имени Д.  
И. Менделеева

---

Факультет цифровых технологий и химического  
инжиниринга  
Кафедра информационных компьютерных технологий

Мохов Марк Геннадьевич КС-34

**Лабораторная работа №3 по предмету управление  
данными:  
«Функции группировки и агрегирования данных».**

**ВАРИАНТ №11**

Научный руководитель:

Семёнов Геннадий Николаевич

Москва – 2024

# СОДЕРЖАНИЕ

|  |    |
|--|----|
| 1. Задание .....   | 3  |
| 2. Решение .....   | 4  |
| 2.1. Сформулировать на русском языке (в виде текста) осмысленные запросы. .... | 4  |
| 3. Вывод .....   | 10 |

# 1. ЗАДАНИЕ

Форма оформления отчета в текстовом редакторе WORD.

1. Сформулировать на русском языке (в виде текста) осмысленные запросы.
2. Для сформулированного запроса написать код на языке SQL.
3. Выполнить запросы в СУБД PostgreSQL, приложить результаты в виде картинки.

Таблица 1 – Задания на создание и выполнения запросов:

| №   | Тип запроса   |
|-----|---|
| 1.  | Создать запрос на простейшее агрегирование с использованием какой-либо функции агрегирования.   |
| 2.  | Создать запрос на агрегирование с группировкой по значению функции (например, на функцию выделения года из данных типа date)  |
| 3.  | Создать запрос на группировку данных по одному полю с использованием функций группировки.   |
| 4.  | Создать запрос на группировку данных по двум полям с использованием функций группировки. И объяснить в чем разница результатов по сравнению с п.3.                    |
| 5.  | Создать запрос на группировку данных по одному полю с использованием функций группировки и условием отбора (HAVING по датам).   |
| 6.  | Создать запрос на группировку данных по одному полю с использованием функций группировки и условием отбора (HAVING) по числовому полю.                                |
| 7.  | Создать запрос на группировку данных по одному полю с использованием функций группировки и условием отбора (HAVING с групповой функцией) по числовому полю.           |
| 8.  | Создать запрос на группировку данных по одному полю с использованием функций группировки с соединением 3-х таблиц.  |
| 9.  | Создать вычисляемый запрос с соединением 3-х таблиц с сортировкой строк по возрастанию в вычисленном поле.  |
| 10. | Создать вычисляемый запрос с соединением 3-х таблиц, с группировкой по одному полю. Сортировку результирующего набора строк выполнить по убыванию в вычисленном поле. |

## 2. РЕШЕНИЕ

### 2.1. Сформулировать на русском языке (в виде текста) осмысленные запросы.

1. Суммарная стоимость всех проектов Сидоровой.

```
1 SELECT SUM(p.cost_rub)
2 FROM authors as a, projects as p
3 WHERE p.author_id = a.author_id AND
4       a.family_name = 'Сидорова';
```

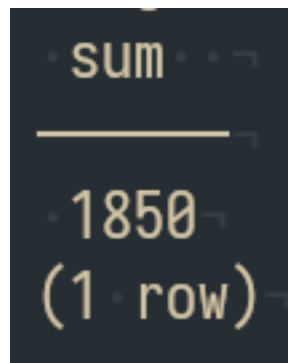


Рисунок 2.1 – Первое задание.

2. Средняя стоимость проектов, по годам.

```
1 SELECT AVG(p.cost_rub) as "avg cost", EXTRACT(YEAR FROM p.start_date)
   as "year"
2 FROM projects as p
3 GROUP BY EXTRACT(YEAR FROM p.start_date);
```

| avg cost           | year |
|--------------------|------|
| 1200               | 2026 |
| 533.33333333333334 | 2024 |
| 950                | 2005 |
| 1200               | 2004 |
| (4 rows)           |      |

Рисунок 2.2 – Второе задание.

3. Средняя стоимость и продолжительность проектов по типам проектов.

```

1 SELECT p.project_type_id, AVG(p.cost_rub), AVG(p.duration_years)
2 FROM projects AS p
3 GROUP BY p.project_type_id;

```

sql

| project_type_id | avg               | avg                |
|-----------------|-------------------|--------------------|
| 3               | 700               | 4.0000000000000000 |
| 2               | 900               | 5.0000000000000000 |
| 1               | 666.6666666666666 | 4.0000000000000000 |
| 1               | 1112.5            | 4.5000000000000000 |
| (4 rows)        |                   |                    |

Рисунок 2.3 – Третье задание.

4. Данные о проектах сгруппированные по авторам и годам.

```

1 SELECT
2     AVG(p.cost_rub) as "avg cost",
3     AVG(p.duration_years) as "avg duration",
4     COUNT(*),
5     EXTRACT(YEAR FROM p.start_date) as "year",
6     a.family_name as "family name"
7 FROM projects AS p
8 LEFT JOIN authors AS a ON a.author_id = p.author_id
9 GROUP BY a.family_name, EXTRACT(YEAR FROM p.start_date);

```

sql

| avg cost | avg duration       | count | year | family name |
|----------|--------------------|-------|------|-------------|
| 1200     | 5.0000000000000000 | 1     | 2024 | Иванов      |
| 200      | 3.0000000000000000 | 1     | 2024 |             |
| 200      | 3.0000000000000000 | 1     | 2024 | Петрова     |
| 925      | 4.0000000000000000 | 2     | 2005 | Сидорова    |
| 1200     | 5.0000000000000000 | 1     | 2026 |             |
| 1000     | 5.0000000000000000 | 1     | 2005 | Иванов      |
| 1200     | 5.0000000000000000 | 1     | 2004 | Иванов      |
| 950      | 4.5000000000000000 | 2     | 2005 | Петрова     |

(8 rows)

Рисунок 2.4 – Четвёртое задание.

5. Данные о проектах сгруппированные по годам в диапазоне с 2004 по 2024.

```

1 SELECT
2     EXTRACT(YEAR FROM p.start_date) as "year",
3     AVG(p.cost_rub) as "avg cost",
4     AVG(p.duration_years) as "avg duration",
5     COUNT(*)
6 FROM projects AS p
7 GROUP BY EXTRACT(YEAR FROM p.start_date)
8 HAVING EXTRACT(YEAR FROM p.start_date) BETWEEN 2005 AND 2024;
```

| year | avg cost          | avg duration       | count |
|------|-------------------|--------------------|-------|
| 2005 | 950               | 4.4000000000000000 | 5     |
| 2024 | 533.3333333333334 | 3.6666666666666667 | 3     |

(2 rows)

Рисунок 2.5 – Пятое задание.

6. Средняя стоимость проектов продолжительностью больше 3 лет.

```

1 SELECT
2     p.duration_years as "duration",
3     AVG(p.cost_rub) as "avg cost",
4     COUNT(*)
5 FROM projects AS p
6 GROUP BY p.duration_years
7 HAVING p.duration_years > 3;
```

| duration | avg cost | count |
|----------|----------|-------|
| 5        | 1100     | 6     |
| 4        | 800      | 1     |
| (2 rows) |          |       |

Рисунок 2.6 – Шестое задание.

7. Данные о проектах сгруппированные по годам со средней стоимостью больше 800.

```

1  SELECT
2      EXTRACT(YEAR FROM p.start_date) as "year",
3      AVG(p.cost_rub) as "avg cost",
4      AVG(p.duration_years) as "avg duration",
5      COUNT(*)
6  FROM projects AS p
7  GROUP BY EXTRACT(YEAR FROM p.start_date)
8  HAVING AVG(p.cost_rub) > 800;

```

| year     | avg cost | avg duration       | count |
|----------|----------|--------------------|-------|
| 2026     | 1200     | 5.0000000000000000 | 1     |
| 2005     | 950      | 4.4000000000000000 | 5     |
| 2004     | 1200     | 5.0000000000000000 | 1     |
| (3 rows) |          |                    |       |

Рисунок 2.7 – Седьмое задание.

8. Полные данные об проектах, авторах и типах сгруппированные по годам.

```

1  SELECT
2      EXTRACT(YEAR FROM p.start_date) as "year",
3      AVG(p.cost_rub) as "avg cost",
4      AVG(p.duration_years) as "avg duration",
5      AVG(a.age) as "avg age",
6      COUNT(t.type_name) as "diff types count"
7  FROM projects AS p
8  LEFT JOIN authors AS a ON a.author_id = p.author_id
9  LEFT JOIN project_types AS t ON t.project_type_id = p.project_type_id

```

```
10 GROUP BY EXTRACT(YEAR FROM p.start_date);
```

| year | avg cost          | avg duration       | avg age            | diff types count |
|------|-------------------|--------------------|--------------------|------------------|
| 2026 | 1200              | 5.0000000000000000 |                    | 1                |
| 2024 | 533.3333333333334 | 3.6666666666666667 | 29.500000000000000 | 1                |
| 2005 | 950               | 4.4000000000000000 | 34.000000000000000 | 5                |
| 2004 | 1200              | 5.0000000000000000 | 30.000000000000000 | 1                |

(4 rows)

Рисунок 2.8 – Восьмое задание.

9. Полные данные об проектах, авторах и типах сгруппированные по годам в порядке возрастания названий.

```
1  SELECT
2      SUM(p.cost_rub) * 0.8 as "sum_cost",
3      AVG(p.duration_years) as "avg duration",
4      AVG(a.age) as "avg age",
5      EXTRACT(YEAR FROM p.start_date) as "year",
6      p.title
7  FROM projects AS p
8      LEFT JOIN authors AS a ON a.author_id = p.author_id
9      LEFT JOIN project_types AS t ON t.project_type_id = p.project_type_id
10 GROUP BY EXTRACT(YEAR FROM p.start_date)
11 ORDER BY p.title ASC;
```

| sum_cost | avg duration       | avg age            | year |
|----------|--------------------|--------------------|------|
| 960      | 5.0000000000000000 | 30.000000000000000 | 2004 |
| 3800     | 4.4000000000000000 | 34.000000000000000 | 2005 |
| 1280     | 3.6666666666666667 | 29.500000000000000 | 2024 |
| 960      | 5.0000000000000000 |                    | 2026 |

(4 rows)

Рисунок 2.9 – Девятое задание.

10. Полные данные об проектах, авторах и типах сгруппированные по годам в порядке возрастания суммарной стоимости проектов.

```
1  SELECT
2      SUM(p.cost_rub) * 0.8 as "sum_cost",
3      AVG(p.duration_years) as "avg duration",
4      AVG(a.age) as "avg age",
5      EXTRACT(YEAR FROM p.start_date) as "year",
6      p.title
7  FROM projects AS p
```



```

8      LEFT JOIN authors AS a ON a.author_id = p.author_id
9      LEFT JOIN project_types AS t ON t.project_type_id = p.project_type_id
10     GROUP BY EXTRACT(YEAR FROM p.start_date)
11     ORDER BY sum_cost DESC;

```

| sum_cost | avg duration       | avg age            | year |
|----------|--------------------|--------------------|------|
| 3800     | 4.4000000000000000 | 34.000000000000000 | 2005 |
| 1280     | 3.6666666666666667 | 29.500000000000000 | 2024 |
| 960      | 5.0000000000000000 |                    | 2026 |
| 960      | 5.0000000000000000 | 30.000000000000000 | 2004 |
| (4 rows) |                    |                    |      |

Рисунок 2.10 – Десятое задание.

### 3. ВЫВОД

В результате выполнения лабораторной работы научился составлять запросы по группировке данных, использовать функции агрегирования и сортировать результат.