

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. Запрос на применение агрегирующих оконных функций	4
2.2. Запрос на применение ранжирующих оконных функций.	5
2.3. Пары атрибутов одной сущности при определенном условии.	6
2.4. Пары атрибутов одной сущности при определенном условии с устранением избыточности.	8
2.5. Однотабличный запрос, использующий подзапрос в условии отбора строк поле фразы WHERE.	9
2.6. Многотабличный запрос, использующий подзапрос в условии отбора строк поле фразы WHERE.	10
2.7. Запрос, использующий подзапрос с агрегатной функцией в условии отбора строк поле фразы WHERE.	11
2.8. Запрос на использование подзапросов, которые выдают много строк с помощью оператора IN.	11
2.9. Запрос, использующий подзапрос в предложении HAVING.	12
2.10. Запрос, использующий подзапрос в предложении FROM.	13
2.11. Запрос на использование соотнесенного подзапроса, который выдает много строк с помощью оператора IN.	14
2.12. Запрос на сравнение таблицы с собой.	14

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.

Задания на создание и выполнения запросов: № Тип запроса

Оконные функции	
1.	Придумать запрос на применение агрегирующих оконных функций
2.	Придумать запрос на применение ранжирующих оконных функций.
Объединение таблиц самих с собой	
3.	Вывести пары атрибутов одной сущности при определенном условии.
4.	То же, что и в пункте 3 с устранением избыточности без оператора DISTINCT.
Подзапросы	
5.	Придумать однотабличный запрос, использующий подзапрос в условии отбора строк поле фразы WHERE.
6.	Придумать многотабличный запрос, использующий подзапрос в условии отбора строк поле фразы WHERE.
7.	Придумать запрос, использующий подзапрос с агрегатной функцией в условии отбора строк поле фразы WHERE.
8.	Придумать запрос на использование подзапросов, которые выдают много строк с помощью оператора IN
9.	Придумать запрос, использующий подзапрос в предложении HAVING
10	Придумать запрос, использующий подзапрос в предложении FROM.
Соотнесенные подзапросы	
11	Придумать запрос на использование соотнесенного подзапроса, который выдает много строк с помощью оператора IN.
12	Придумать запрос на сравнение таблицы с собой

2. РЕШЕНИЕ

2.1. Запрос на применение агрегирующих оконных функций

Средняя стоимость проектов разделённых по типам. SQL запрос:

```
1 SELECT
2     p.title AS "Title",
3     t.type_name AS "Type",
4     p.cost_rub AS "Cost",
5     AVG(p.cost_rub) OVER( PARTITION BY p.project_type_id ) AS "AVG"
6 FROM projects AS p
7 LEFT JOIN project_types AS t
8     ON t.project_type_id = p.project_type_id;
```

Таблица 2 – Результат.

Title	Type	Cost	Avg
Проект1	Технология	1200	1112.5
Проект3	Технология	950	1112.5
Проект6	Технология	1100	1112.5
Проект Ы	Технология	1200	1112.5
Проект А	НИР	200	666.6666666666666
Проект2	НИР	800	666.6666666666666
Проект4	НИР	1000	666.6666666666666
Проект5	ОКР	900	900
Проект Ы		1200	700
Проект А		200	700

2.2. Запрос на применение ранжирующих оконных функций.

Проекты по возрастанию даты.

```
1 SELECT
2     title,
3     start_date,
4     row_number() OVER date_grades
5 FROM projects
6 WINDOW date_grades AS (order by start_date)
```

sql

Таблица 3 – Результат.

title	start_date	row_number
Проект1	2004-02-03	1
Проект2	2005-01-15	2
Проект3	2005-02-02	3
Проект5	2005-02-14	4
Проект4	2005-03-02	5
Проект6	2005-04-04	6
Проект Ы	2024-02-03	7
Проект А	2024-05-03	8
Проект А	2024-05-03	9
Проект Ы	2026-02-03	10

2.3. Пары атрибутов одной сущности при определенном условии.

Пары проектов с одинаковой продолжительностью.

```
1 SELECT
2     p1.duration_years,
3     p1.project_id as "id1", p1.title,
4     p2.project_id as "id2", p2.title
5 FROM projects as p1, projects as p2
6 WHERE
7     p1.duration_years = p2.duration_years AND
8     p1.project_id != p2.project_id;
```

Таблица 4 – Результат.

duration_years	id1	title	id2	title
5	1	Проект1	9	Проект Ы
5	1	Проект1	7	Проект Ы
5	1	Проект1	6	Проект6
5	1	Проект1	5	Проект5
5	1	Проект1	4	Проект4
3	3	Проект3	10	Проект А
3	3	Проект3	8	Проект А
5	4	Проект4	9	Проект Ы
5	4	Проект4	7	Проект Ы
5	4	Проект4	6	Проект6
5	4	Проект4	5	Проект5
5	4	Проект4	1	Проект1
5	5	Проект5	9	Проект Ы
5	5	Проект5	7	Проект Ы
5	5	Проект5	6	Проект6
5	5	Проект5	4	Проект4
5	5	Проект5	1	Проект1
5	6	Проект6	9	Проект Ы
5	6	Проект6	7	Проект Ы
5	6	Проект6	5	Проект5
5	6	Проект6	4	Проект4
5	6	Проект6	1	Проект1
5	7	Проект Ы	9	Проект Ы
5	7	Проект Ы	6	Проект6

5	7	Проект Ы	5	Проект5
5	7	Проект Ы	4	Проект4
5	7	Проект Ы	1	Проект1
3	8	Проект А	10	Проект А
3	8	Проект А	3	Проект3
5	9	Проект Ы	7	Проект Ы
5	9	Проект Ы	6	Проект6
5	9	Проект Ы	5	Проект5
5	9	Проект Ы	4	Проект4
5	9	Проект Ы	1	Проект1
3	10	Проект А	8	Проект А
3	10	Проект А	3	Проект3

2.4. Пары атрибутов одной сущности при определенном условии с устранением избыточности.

Пары проектов с одинаковой продолжительностью.

```
1 SELECT
2     p1.duration_years,
3     p1.project_id as "id1", p1.title,
4     p2.project_id as "id2", p2.title
5 FROM projects as p1, projects as p2
6 WHERE
7     p1.duration_years = p2.duration_years AND
8     p1.project_id > p2.project_id;
```

sql

Таблица 5 – Результат.

duration_years	id1	title	id2	title
5	4	Проект4	1	Проект1
5	5	Проект5	4	Проект4
5	5	Проект5	1	Проект1
5	6	Проект6	5	Проект5
5	6	Проект6	4	Проект4
5	6	Проект6	1	Проект1
5	7	Проект Ы	6	Проект6
5	7	Проект Ы	5	Проект5
5	7	Проект Ы	4	Проект4
5	7	Проект Ы	1	Проект1
3	8	Проект А	3	Проект3
5	9	Проект Ы	7	Проект Ы
5	9	Проект Ы	6	Проект6
5	9	Проект Ы	5	Проект5
5	9	Проект Ы	4	Проект4
5	9	Проект Ы	1	Проект1
3	10	Проект А	8	Проект А
3	10	Проект А	3	Проект3

2.5. Однотабличный запрос, использующий подзапрос в условии отбора строк поле фразы WHERE.

Проекты дороже средней стоимости.

```
1 SELECT p.title, p.cost_rub, (SELECT AVG(_p.cost_rub) FROM projects AS _p)
2 FROM projects AS p
3 WHERE p.cost_rub > (SELECT AVG(_p.cost_rub) FROM projects AS _p);
```

sql

Таблица 6 – Результат.

title	cost_rub	avg
Проект1	1200	875
Проект3	950	875
Проект4	1000	875
Проект5	900	875
Проект6	1100	875
Проект Ы	1200	875
Проект Ы	1200	875

2.6. Многотабличный запрос, использующий подзапрос в условии отбора строк поле фразы WHERE.

Проекты, авторы которых старше среднего возраста.

```
1 SELECT
2     p.title, a.family_name, a.age,
3     (SELECT AVG(_a.age) FROM authors AS _a)
4 FROM projects as p, authors as a
5 WHERE
6     a.age >= (SELECT AVG(_a.age) FROM authors AS _a) AND
7     a.author_id = p.author_id;
```

Таблица 7 – Результат.

title	family_name	age	avg
Проект3	Сидорова	41	33.33333333333333
Проект5	Сидорова	41	33.33333333333333

2.7. Запрос, использующий подзапрос с агрегатной функцией в условии отбора строк поле фразы WHERE.

Проекты составляющие больше 50% общей стоимости всех проектов этого типа.

```
1 SELECT
2     p.title AS "Title",
3     p.project_type_id AS "Type ID",
4     p.cost_rub AS "Cost"
5 FROM projects AS p
6 WHERE
7     p.cost_rub * 2 >= (SELECT SUM(_p.cost_rub)
8                        FROM projects AS _p
9                        WHERE _p.project_type_id = p.project_type_id);
```

Таблица 8 – Результат.

Title	Type ID	Cost
Проект4	2	1000
Проект5	3	900

2.8. Запрос на использование подзапросов, которые выдают много строк с помощью оператора IN.

Проекты авторов старше 30 лет.

```
1 SELECT p.title, p.cost_rub
2 FROM projects AS p
3 WHERE p.author_id IN (SELECT a.author_id FROM authors AS a WHERE a.age >= 30);
```

Таблица 9 – Результат.

title	cost_rub
Проект1	1200
Проект3	950
Проект4	1000
Проект5	900
Проект Ы	1200

2.9. Запрос, использующий подзапрос в предложении HAVING.

Годы в которых средняя стоимость проектов была больше общей средней стоимости.

```
1 SELECT
2     EXTRACT(YEAR FROM p.start_date) AS year,
3     AVG(p.cost_rub) AS avg_cost
4 FROM projects AS p
5 GROUP BY year
6 HAVING AVG(p.cost_rub) >= (SELECT AVG(p.cost_rub) FROM projects AS p)
7 ORDER BY year;
```

Таблица 10 – Результат.

year	avg_cost
2004	1200
2005	950
2026	1200

2.10. Запрос, использующий подзапрос в предложении FROM.

Названия, стоимость и дата начала всех проектов.

```
1 SELECT *
2 FROM (SELECT title, cost_rub, start_date FROM projects)
3 ORDER BY start_date;
```

sql

Таблица 11 – Результат.

title	cost_rub	start_date
Проект1	1200	2004-02-03
Проект2	800	2005-01-15
Проект3	950	2005-02-02
Проект5	900	2005-02-14
Проект4	1000	2005-03-02
Проект6	1100	2005-04-04
Проект Ы	1200	2024-02-03
Проект А	200	2024-05-03
Проект А	200	2024-05-03
Проект Ы	1200	2026-02-03

2.11. Запрос на использование соотнесенного подзапроса, который выдает много строк с помощью оператора IN.

Типы проектов на которыми работал Иванов.

```
1 SELECT t.project_type_id, t.type_name
2 FROM project_types AS t
3 WHERE 1 IN (SELECT p.author_id
4             FROM projects as p
5             WHERE p.project_type_id = t.project_type_id);
```

Таблица 12 – Результат.

project_type_id	type_name
1	Технология
2	НИР

2.12. Запрос на сравнение таблицы с собой.

Проекты дороже средней стоимости проектов того же типа.

```
1 SELECT
2     p.title AS "Title",
3     p.cost_rub AS "Cost",
4     p.project_type_id AS "Type Id",
5     (SELECT AVG(p3.cost_rub)
6      FROM projects AS p3
7      WHERE p3.project_type_id = p.project_type_id) AS "Type Avg"
8 FROM projects AS p
9 WHERE p.cost_rub > (SELECT AVG(p2.cost_rub)
10                   FROM projects AS p2
11                   WHERE p2.project_type_id = p.project_type_id);
```

Таблица 13 – Результат.

Title	Cost	Type Id	Type Avg
Проект1	1200	1	1112.5
Проект2	800	2	666.6666666666666
Проект4	1000	2	666.6666666666666
Проект Ы	1200	1	1112.5