

Министерство образования Республики Беларусь

Учреждение образования
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра электронных вычислительных машин

Дисциплина: Базы данных

ОТЧЕТ

по лабораторной работе № 5

на тему

Реализация SQL-запросов на выборку данных с использованием
подзапросов, агрегатных функций, группировки и операций над
множествами,
вариант № 16 –Кинотеатр

Студент:

А.В. Крачковский

Проверила:

А.О. Игнатович

МИНСК 2023

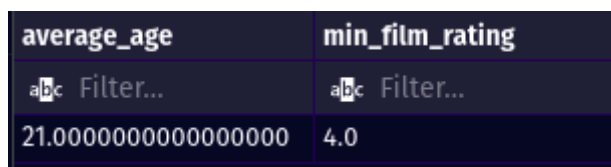
1. Описание

Ниже будут приведены скриншоты результатов запроса. Запросы разделены по таблицам, таблицы в свою очередь являются разделами. Название файла скрипта является подразделом, в нем же и находится скриншот. Описание запроса написаны в комментариях к самому запросу.

2. Таблица actors

2.1 actorAgeRate.sql

```
-- Вывести средний возраст актеров и минимальный рейтинг фильма, в
-- котором они участвовали.
SELECT AVG(a.age) AS average_age,
MIN(f.stars) AS min_film_rating
FROM actors a
JOIN film_actor fa ON a.full_name = fa.actor_name
JOIN film f ON fa.specific_film_name = f.film_name;
```



average_age	min_film_rating
21.000000000000000	4.0

Рисунок 2.1 – результат запроса №1

2.2 actorFilm.sql

```
-- Вывести имена актеров и количество фильмов,
-- в которых они участвовали, упорядоченные по убыванию количества
-- фильмов.
SELECT a.full_name,
COUNT(fa.specific_film_name) AS film_count
FROM actors a
LEFT JOIN film_actor fa ON a.full_name = fa.actor_name
GROUP BY a.full_name
ORDER BY film_count DESC;
```



full_name	film_count
 Filter...	 Filter...
Дикаприо Анна Олегов...	1
Гослинг Анна Олеговна	1
Депп Анна Олеговна	1
Стэтхам Анна Олеговна	1
Питт Анна Олеговна	1
Круз Анна Олеговна	1

Рисунок 2.2 – результат запроса №2

2.3 actorFilmRate.sql

```
-- Вывести имена актеров и их роли в фильмах с рейтингом выше 4.2.
SELECT actor_name,
actor_role
FROM film_actor
WHERE specific_film_name IN (
SELECT film_name
FROM film
WHERE stars > 4.2
);
```



actor_name	actor_role
 Filter...	 Filter...
Гослинг Анна Олеговна	Главная роль
Стэтхам Анна Олеговна	Киборг-убийца
Питт Анна Олеговна	Главная роль
Дикаприо Анна Олегов...	Вторая личность главного героя

Рисунок 2.3 – результат запроса №3

2.4 nameRateGenre.sql

```
-- Вывести актеров, чей рейтинг выше 4.5 и возраст равен 21.
SELECT full_name
FROM actors
WHERE rating > 4.5
AND age = 21;
```

full_name	average_rating
abc Filter...	abc Filter...
Гослинг Анна Олеговна	4.5000000000000000
Дикаприо Анна Олегов...	4.3000000000000000
Стэтхэм Анна Олеговна	4.3000000000000000

Рисунок 2.4 – результат запроса №4

2.5 namesWithAvgRate.sql

-- Вывести имена актеров и средний рейтинг фильмов, в которых они снимались.

```
SELECT a.full_name,
AVG(f.stars) AS average_rating
FROM actors a
JOIN film_actor fa ON a.full_name = fa.actor_name
JOIN film f ON fa.specific_film_name = f.film_name
GROUP BY a.full_name;
```

full_name	average_rating
abc Filter...	abc Filter...
Дикаприо Анна Олегов...	4.3000000000000000
Гослинг Анна Олеговна	4.5000000000000000
Депп Анна Олеговна	4.1000000000000000
Стэтхэм Анна Олеговна	4.3000000000000000
Питт Анна Олеговна	4.5000000000000000
Круз Анна Олеговна	4.0000000000000000

Рисунок 2.5 – результат запроса №5

3. Таблица cinema_hall

3.1 averageSits.sql

```
-- Вывести названия залов и среднее количество мест в каждом зале.  
SELECT hall_name,  
AVG(total_number_of_seats) AS average_seats  
FROM cinema_hall  
GROUP BY hall_name;
```

hall_name	average_seats
 Filter...	 Filter...
VIP Hall	100.00000000000000...
Hall B	120.00000000000000...
Hall A	150.00000000000000...

Рисунок 3.1 – результат запроса №1

3.2 hallReservation.sql

```
-- Вывести названия залов и количество бронирований в каждом зале,  
-- где количество бронирований больше 0.  
SELECT ch.hall_name,  
COUNT(r.identification_number) AS reservation_count  
FROM cinema_hall ch  
LEFT JOIN reservation r ON ch.hall_name = r.cinema_hall_name  
GROUP BY ch.hall_name  
HAVING COUNT(r.identification_number) > 0;
```


hall_name	reservation_count
 Filter...	 Filter...
VIP Hall	1
Hall B	1
Hall A	1

Рисунок 3.2 – результат запроса №2

3.3 hallsAgeActors.sql

```
-- Вывести названия залов и средний возраст актеров, снимавшихся в  
фильмах,  
-- показываемых в каждом зале.  
SELECT ch.hall_name,
```

```

AVG(a.age) AS average_actor_age
FROM cinema_hall ch
JOIN reservation r ON ch.hall_name = r.cinema_hall_name
JOIN film_actor fa ON r.playing_film_name = fa.specific_film_name
JOIN actors a ON fa.actor_name = a.full_name
GROUP BY ch.hall_name;

```

hall_name	average_actor_age
abc Filter...	abc Filter...
VIP Hall	21.0000000000000000
Hall B	21.0000000000000000
Hall A	21.0000000000000000

Рисунок 3.3 – результат запроса №3

3.4 hallsWithEmployees.sql

```

-- Вывести все залы кинотеатра и количество сотрудников,
-- работающих в каждом зале.
SELECT ch.hall_name,
COUNT(e.full_name) AS employee_count
FROM cinema_hall ch
LEFT JOIN employee e ON ch.assigned_employee = e.full_name
GROUP BY ch.hall_name;

```

hall_name	employee_count
abc Filter...	abc Filter...
VIP Hall	1
Hall B	1
Hall A	1

Рисунок 3.4 – результат запроса №4

3.5 hallsWithReservations.sql

```

-- Вывести названия залов и количество бронирований, сделанных в
-- каждом зале.
SELECT ch.hall_name,
COUNT(r.identification_number) AS reservation_count
FROM cinema_hall ch
LEFT JOIN reservation r ON ch.hall_name = r.cinema_hall_name
GROUP BY ch.hall_name;

```

hall_name	reservation_count
abc Filter...	abc Filter...
VIP Hall	1
Hall B	1
Hall A	1

Рисунок 3.5 – результат запроса №5

4. Таблица client

4.1 averageTicketNum.sql

-- Вывести имена клиентов и средний номер билета для каждого клиента.

```
SELECT full_name,  
AVG(ticket_number) AS average_ticket_number  
FROM client  
GROUP BY full_name;
```

full_name	average_ticket_nu...
abc Filter...	abc Filter...
Игнатович Анна Олеговна	10.0000000000000000
Крачковский Александр Витальевич	100.0000000000000000
Жук Тимур Сергеевич	1000.00000000000000...

Рисунок 4.1 – результат запроса №1

4.2 namesAmountReservations.sql

-- Вывести имена клиентов и количество бронирований, упорядоченные по убыванию количества бронирований.

```
SELECT c.full_name,  
COUNT(r.identification_number) AS reservation_count  
FROM client c  
LEFT JOIN reservation r ON c.full_name = r.client_full_name  
GROUP BY c.full_name  
ORDER BY reservation_count DESC;
```

full_name	reservation_count
abc Filter...	abc Filter...
Игнатович Анна Олеговна	1
Крачковский Александр Витальевич	1
Жук Тимур Сергеевич	1

Рисунок 4.2 – результат запроса №2

4.3 nameSumTickets.sql

-- Вывести имена клиентов и общую сумму билетов, которые они приобрели.

```
SELECT c.full_name,  
SUM(f.stars) AS total_ticket_cost
```



```

FROM client c
JOIN reservation r ON c.full_name = r.client_full_name
JOIN film f ON r.playing_film_name = f.film_name
GROUP BY c.full_name;

```

full_name	total_ticket_cost
abc Filter...	abc Filter...
Игнатович Анна Олеговна	4.3
Крачковский Александр Витальевич	4.5
Жук Тимур Сергеевич	4.0

Рисунок 4.3 – результат запроса №3

4.4 namesWithReservations.sql

```

-- Вывести имена клиентов и количество бронирований,
-- сделанных каждым клиентом.
SELECT c.full_name,
COUNT(r.identification_number) AS reservation_count
FROM client c
LEFT JOIN reservation r ON c.full_name = r.client_full_name
GROUP BY c.full_name;

```

full_name	reservation_count
abc Filter...	abc Filter...
Игнатович Анна Олеговна	1
Крачковский Александр Витальевич	1
Жук Тимур Сергеевич	1

Рисунок 4.4 – результат запроса №4

4.5 reservationBy.sql

```

-- Вывести имена клиентов и общее количество бронирований
-- в залах с номером меньше 5.
SELECT c.full_name,
COUNT(r.identification_number) AS reservation_count
FROM client c
JOIN reservation r ON c.full_name = r.client_full_name
JOIN cinema_hall ch ON r.cinema_hall_name = ch.hall_name
WHERE ch.hall_number < 2
GROUP BY c.full_name;

```

full_name	reservation_count
<input type="text" value="abc Filter..."/>	<input type="text" value="abc Filter..."/>
Крачковский Александр Витальевич	1

Рисунок 4.5 – результат запроса №5

5. Таблица employee

5.1 hardWorkingEmployee.sql

```
-- Вывести имена сотрудников и общее количество бронирований в
фильмах,
-- где они работали.
SELECT e.full_name,
COUNT(r.identification_number) AS reservation_count
FROM employee e
JOIN cinema_hall ch ON e.full_name = ch.assigned_employee
JOIN reservation r ON ch.hall_name = r.cinema_hall_name
GROUP BY e.full_name;
```

full_name	reservation_count
abc Filter...	abc Filter...
Иван Иванов	1
Наполеон Бонапарт	1
Паша техник	1

Рисунок 5.1 – результат запроса №1

5.2 nameWithAvg.sql

```
-- Вывести имена сотрудников и среднюю зарплату для каждого
сотрудника.
SELECT e.full_name,
AVG(e.salary) AS average_salary
FROM employee e
GROUP BY e.full_name;
```

full_name	average_salary
abc Filter...	abc Filter...
Иван Иванов	1400.0000000000000000...
Алексей Павлов	2100.0000000000000000
Наполеон Бонапарт	1400.0000000000000000...
Мария Козлова	1000.0000000000000000...
Анна Петрова	1300.0000000000000000...
Паша техник	1400.0000000000000000...
Сергей Смирнов	2320.0000000000000000

Рисунок 5.2 – результат запроса №2

5.3 nameWithHallsAmount.sql

-- Вывести имена сотрудников и количество залов, в которых они работают.

```
SELECT e.full_name,  
COUNT(ch.hall_name) AS hall_count  
FROM employee e  
LEFT JOIN cinema_hall ch ON e.full_name = ch.assigned_employee  
GROUP BY e.full_name;
```

full_name	hall_count
Иван Иванов	1
Алексей Павлов	0
Наполеон Бонапарт	1
Мария Козлова	0
Анна Петрова	0
Паша техник	1
Сергей Смирнов	0

Рисунок 5.3 – результат запроса №3

5.4 nameWithMinSalary.sql

-- Вывести имена сотрудников и минимальную зарплату для каждой должности.

```
SELECT e.full_name,  
MIN(e.salary) AS min_salary  
FROM employee e  
GROUP BY e.full_name;
```

full_name	min_salary
Иван Иванов	1400
Алексей Павлов	2100
Наполеон Бонапарт	1400
Мария Козлова	1000
Анна Петрова	1300
Паша техник	1400
Сергей Смирнов	2320

Рисунок 5.4 – результат запроса №4

5.5 nameWithworkhours.sql

-- Вывести имена сотрудников и общее количество часов работы, упорядоченные по убыванию общего количества часов.

```
SELECT e.full_name,  
SUM(  
EXTRACT(  
HOUR  
FROM (e.end_time - e.start_time)  
)  
AS total_hours  
FROM employee e  
GROUP BY e.full_name  
ORDER BY total_hours DESC;
```

full_name	total_hours
<input type="text" value="Filter..."/>	<input type="text" value="Filter..."/>
Иван Иванов	9
Алексей Павлов	9
Наполеон Бонапарт	9
Мария Козлова	9
Анна Петрова	9
Паша техник	9
Сергей Смирнов	9

Рисунок 5.5 – результат запроса №5

6. Таблица film

6.1 filmAndReservedSeats.sql

-- Вывести названия фильмов и общее количество забронированных мест на этих фильмах.

```
SELECT f.film_name,  
SUM(r.column_number * r.sit_number) AS total_reserved_seats  
FROM film f  
JOIN reservation r ON f.film_name = r.playing_film_name  
GROUP BY f.film_name;
```

film_name	total_reserved_s...
abc Filter...	abc Filter...
Мне нужна только Анна	6
Секреты Анны: Откровения Черных Котов	5
Анна и Сумасшедшая Жизнь	6

Рисунок 6.1 – результат запроса №1

6.2 filmsOneLine.sql

-- Вывести названия фильмов и список жанров в одной строке для каждого фильма.

```
SELECT film_name,  
STRING_AGG(genre, ', ') AS genres  
FROM (  
SELECT f.film_name,  
UNNEST(f.genres) AS genre  
FROM film f  
) AS film_genres  
GROUP BY film_name;
```

film_name	genres
abc Filter...	abc Filter...
Мне нужна только Анна	Драмма, Комедия
Секреты Анны: Откровения Черных Котов	Комедия, Мистика
Анна: Путешествие к Чудовищам	Фэнтези, Приключение
Анна и Путешествие во Времени	Приключение, Научная фантастика
Анна и Сумасшедшая Жизнь	Комедия, Драмма

Рисунок 6.2 – результат запроса №2

6.3 filmWithAmountOfActors.sql

```
-- Вывести названия фильмов и количество актеров,  
-- участвовавших в каждом фильме.  
SELECT f.film_name,  
COUNT(fa.actor_name) AS actor_count  
FROM film f  
LEFT JOIN film_actor fa ON f.film_name = fa.specific_film_name  
GROUP BY f.film_name;
```

film_name	actor_count
abc Filter...	abc Filter...
Анна: Путешествие к Чудовищам	1
Мне нужна только Анна	1
Секреты Анны: Откровения Черных Котов	1
Анна и Путешествие во Времени	1
Анна и Сумасшедшая Жизнь	2

Рисунок 6.3 – результат запроса №3

6.4 filmWithRate.sql

```
-- Вывести названия фильмов и средний рейтинг по каждому фильму.  
SELECT film_name,  
AVG(stars) AS average_rating  
FROM film  
GROUP BY film_name;
```

film_name	average_rating
abc Filter...	abc Filter...
Анна: Путешествие к Чудовищам	4.1000000000000000
Мне нужна только Анна	4.5000000000000000
Секреты Анны: Откровения Черных Котов	4.0000000000000000
Анна и Путешествие во Времени	4.5000000000000000
Анна и Сумасшедшая Жизнь	4.3000000000000000

Рисунок 6.4 – результат запроса №4

6.5 uniqueGenresFilm.sql

```
-- Вывести названия фильмов и количество уникальных жанров в  
каждом фильме.  
SELECT film_name,  
COUNT(DISTINCT genre) AS unique_genre_count  
FROM (  
SELECT f.film_name,  
UNNEST(f.genres) AS genre  
FROM film f
```

```
) AS film_genres  
GROUP BY film_name;
```

film_name	unique_genre_co...
abc Filter...	abc Filter...
Анна и Путешествие во Времени	2
Анна и Сумасшедшая Жизнь	2
Анна: Путешествие к Чудовищам	2
Мне нужна только Анна	2
Секреты Анны: Откровения Черных Котов	2

Рисунок 6.5 – результат запроса №5

7. Таблица job_title

7.1 jobStats.sql

```
-- Вывести названия всех должностей и общую зарплату для каждой  
-- должности,  
-- упорядоченные по убыванию общей зарплаты.  
SELECT jt.title,  
SUM(e.salary) AS total_salary  
FROM job_title jt  
JOIN employee e ON jt.title = e.job_title  
GROUP BY jt.title  
ORDER BY total_salary DESC;
```

title	total_salary
abc Filter...	abc Filter...
Менеджер кинозала	4200
Кинотехник	2320
Оператор проекционной комна...	2100
Продавец-билетов	1300
Уборщик	1000

Рисунок 7.1 – результат запроса №1

7.2 titlesWithAmountOfEmployee.sql

```
-- Вывести названия всех должностей  
-- и количество сотрудников на каждой должности.  
SELECT j.title,  
COUNT(e.full_name) AS employee_count  
FROM job_title j  
JOIN employee e ON j.title = e.job_title  
GROUP BY j.title;
```

title	employee_count
abc Filter...	abc Filter...
уборщик	1
Менеджер кинозала	3
Оператор проекционной комна...	1
Кинотехник	1
Продавец-билетов	1

Рисунок 7.2 – результат запроса №2

7.3 titlesWithSalary.sql

```
-- Вывести все названия должностей и соответствующие им
минимальные
-- и максимальные зарплаты.
SELECT title,
MIN(salary) AS min_salary,
MAX(salary) AS max_salary
FROM job_title
JOIN employee ON job_title.title = employee.job_title
GROUP BY title;
```

title	min_salary	max_salary
abc Filter...	abc Filter...	abc Filter...
Уборщик	1000	1000
Менеджер кинозала	1400	1400
Оператор проекционной комна...	2100	2100
Кинотехник	2320	2320
Продавец-билетов	1300	1300

Рисунок 7.3 – результат запроса №3

7.4 titleWithAvgSalary.sql

```
-- Вывести названия должностей и среднюю зарплату для каждой
должности.
SELECT title, AVG(salary) AS average_salary
FROM job_title
JOIN employee ON job_title.title = employee.job_title
GROUP BY title;
```

title	average_salary
abc Filter...	abc Filter...
Уборщик	1000.0000000000000000...
Менеджер кинозала	1400.0000000000000000...
Оператор проекционной комна...	2100.0000000000000000
Кинотехник	2320.0000000000000000
Продавец-билетов	1300.0000000000000000...

Рисунок 7.4 – результат запроса №4

7.5 valuableJobs.sql

```
-- Вывести названия должностей и количество сотрудников, у которых
зарплата выше средней зарплаты по всем должностям.
SELECT jt.title,
```

```

COUNT(e.full_name) AS high_salary_employee_count
FROM job_title jt
JOIN employee e ON jt.title = e.job_title
WHERE e.salary > (
SELECT AVG(salary)
FROM employee
)
GROUP BY jt.title;

```

title	high_salary_emp...
abc Filter...	abc Filter...
Оператор проекционной комна...	1
Кинотехник	1

Рисунок 7.5 – результат запроса №5