

ЛАБОРАТОРНАЯ РАБОТА №2

«Язык SQL. Манипулирование базой данных. Запросы на основе нескольких таблиц»

2.1 Цель работы

Изучить основы реляционной алгебры, как базового средства манипулирования. Выработать у обучающихся практические навыки по работе с реляционными базами данных и представлению запросов как на языке реляционной алгебре, так и SQL.

2.2 Индивидуальный вариант

На рисунке 2.1 показана схема базы данных туристического агентства, построенная в соответствии с вариантом 9.

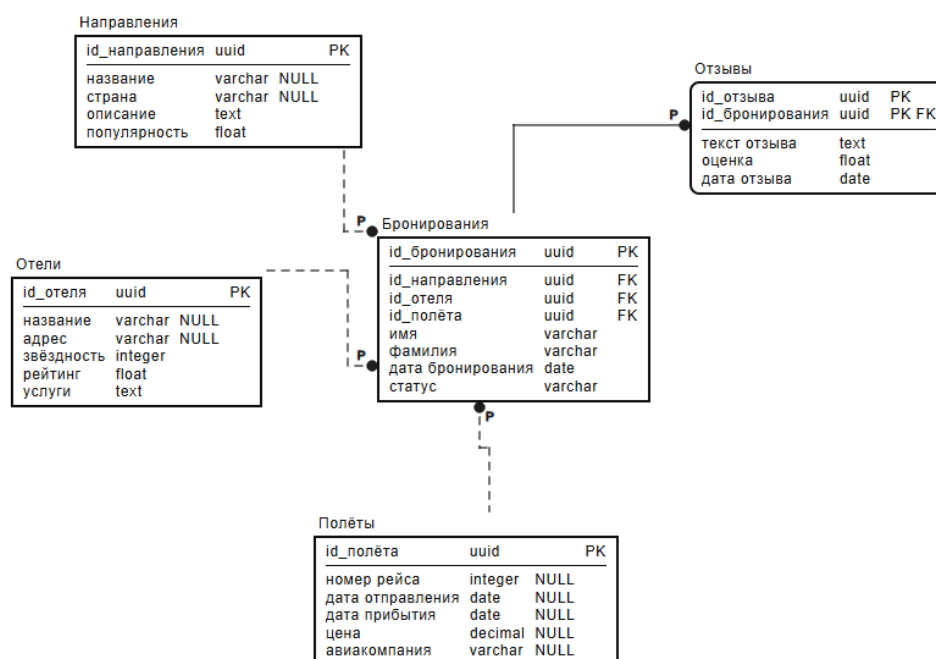


Рисунок 2.1 – Схема базы данных туристического агентства

2.3 Ход выполнения работы

2.3.1 После изучения методических указаний были придуманы и написаны запросы, основывающиеся на реляционной алгебре. Запрос, отражающий операции селекции и соединения в одном запросе: необходимо вывести всю информацию о направлениях, которые бронировали люди с именем John. Запрос, представленный в форме PA:

$R = \sigma (\text{имя} = \text{John} (\text{bookings})) \bowtie \text{directions}$
 $\text{direction_id} = \text{directions.id}$

Текст запроса показан в листинге 2.1, результат выполнения на рисунке 2.2.

Листинг 2.1 – Запрос, содержащий операции селекции и соединения

```
select directions.* from bookings join directions on
bookings.direction_id = directions.id where bookings.name = 'John';
```

	id	name	country	description	popularity
1	75fb288b-368b-461b-afe5-6318c5576834	Paris	France	City of Light	9.8

Рисунок 2.2 – Результат вывода запроса, содержащего операции селекции и соединения

Запрос, использующий операции проекции и деления в одном запросе: выбрать названия отелей, которые есть во всех возможных бронированиях. Запрос, представленный в форме PA:

$R1 = \pi_{\text{name}} (\text{hotels})$
 $R2 = \pi_{\text{hotel_id}} (\text{bookings})$
 $R = R1/R2$

Текст запроса показан в листинге 2.2, результат выполнения на рисунке 2.3.

Листинг 2.2 – Запрос, использующий операции проекции и деления

```
select distinct name from hotels h1 where not exists
  (select hotel_id from bookings where not exists
    (select id from hotels h2
     where h2.id = h1.id AND bookings.hotel_id = h2.id))
```

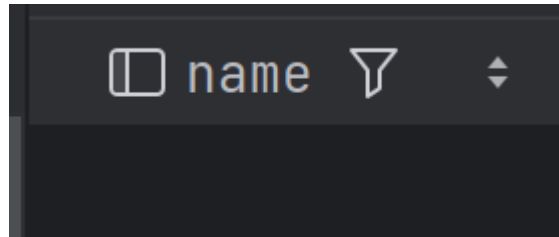


Рисунок 2.3 – Результат вывода запроса, использующего операции проекции и деления

Запрос, использующий операции проекции, объединения и конъюнкции в одном запросе: выбрать идентификаторы отелей либо из бронирования с направлением во францию либо с количеством звёзд больше

4. Запрос, представленный в форме РА:

$R1 = \pi_{id} \sigma_{stars > 4} (hotels)$

$R2 = \pi_{hotel_id} (bookings)$

$R3 = \pi_{id} \sigma_{country = France} (directions)$

$R = R1 \cup R3 \cup R2$

Текст запроса показан в листинге 2.3, результат выполнения на рисунке 2.4.

Листинг 2.3 – Запрос, использующий операции проекции, объединения и конъюнкции

```
select bookings.hotel_id from bookings where direction_id in (select id
from directions where country = 'France')
union select id from hotels where stars > 4;
```

	<input type="checkbox"/> id <input type="text"/>
1	595aa6e3-3b09-416f-9843-fe1ca93ca926
2	7f23a9c2-fd42-41df-b753-1232ab102a98
3	5c6e4eae-d0c1-41f6-8007-64d8dec28f3d
4	40170125-0dbe-4ae8-9dcd-edadcff25d8d
5	c2dfe10f-c3f3-4fa4-be97-fd767f66f04c

Рисунок 2.4 – Результат вывода запроса, использующего операции проекции, объединения и конъюнкции

Запрос, использующий операции соединения и деления в одном запросе: выбрать названия всех отелей в бронированиях, для которых проводились все возможные полёты. Запрос, представленный в форме PA:

R1=hotels ► ◄bookings

hotels.id = bookings.hotel_id

R2= π_{id} (flights)

R= **R1****R2**

Текст запроса показан в листинге 2.4, результат выполнения на рисунке 2.5.

Листинг 2.4 – Запрос, использующий операции соединения и деления

```
select hotels.name from hotels join bookings b1 on hotels.id =
b1.hotel_id
where not exists (select id from flights where not exists
(select id from bookings b2 where b2.id = b1.id and b2.flight_id =
flights.id))
```

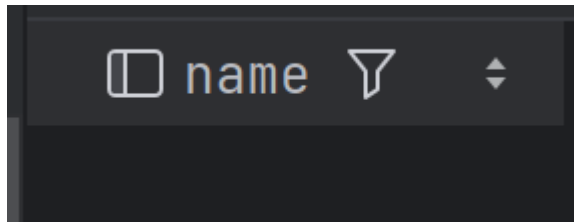


Рисунок 2.5 – Результат вывода запроса, использующего операции соединения и деления

Запрос, использующий операции вычитания и дизъюнкции в одном запросе: выбрать идентификаторы всех полётов, которые не бронировали с 2024-09-04 до 2024-10-30 или их стоимость выше 1000. Запрос, представленный в форме PA:

$R1 = \pi_{\text{flight_id}}(\text{bookings})$

$R2 = \pi_{\text{id}} \sigma_{\text{cost} > 1000}(\text{flights})$

$R3 = \pi_{\text{id}} \sigma_{\text{booking_date} > 2024-09-04 \wedge \text{booking_date} < 2024-10-30}(\text{bookings})$

$R = (R1 - R3) \cup R2$

Текст запроса показан в листинге 2.5, результат выполнения на рисунке 2.6.

Листинг 2.5 – Запрос, использующий операции вычитания и дизъюнкции

```
flight_id from bookings b1 where not exists (select id from
bookings b2 where b1.id = b2.id and booking_date between
'2024-09-04' and '2024-10-30')
union select id from flights where cost > 1000;
```

	flight_id
1	9b8b7cd9-af03-44b3-8aca-883775a6fc3a
2	720004bf-cbe9-4cc2-a5a6-7b1954d1c660
3	a35b451e-8c80-44fc-84c7-016943e46cf2
4	10fa75f0-8dfe-4e8e-bcb4-8f6af92dac75
5	6c3ba82a-5982-4ad6-b675-f313e47ed0be

Рисунок 2.6 – Результат вывода запроса, использующего операции вычитания и дизъюнкции

Запрос, не реализующийся на РА: вычислить среднюю стоимость полёта в тенге. Текст запроса показан в листинге 2.7, результат выполнения на рисунке 2.8.

Листинг 2.7 – Запрос, не реализующийся на РА

```
select avg(cost)*4.94 as avg_cost from flights;
```

	avg_cost
1	3705

Рисунок 2.8 – Результат вывода запроса, не реализующегося на РА

Выводы

В начале выполнения лабораторной работы были изучены методические указания. Были рассмотрены примеры операций реляционной алгебры, а именно: проекция, селекция, объединение, пересечение, вычитание, декартово произведение, соединение и деление. Далее с помощью полученных знаний были придуманы запросы на естественном языке, также они были представлены в форме реляционной алгебры. Затем эти запросы были написаны с помощью PostgreSQL.