

Министерство науки и высшего образования Российской Федерации
федеральное государственное автономное образовательное учреждение высшего образования
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»

Отчет

по лабораторной работе №2

«Запросы на выборку и модификацию данных, представления и индексы в PostgreSQL»

по дисциплине «Проектирование и реализация баз данных»

Автор: Макунина А.А.

Факультет: ИКТ

Группа: К32421

Преподаватель: Говорова М.М.



Санкт-Петербург 2023

Цель работы: овладеть практическими навыками создания представлений и запросов на выборку данных к базе данных PostgreSQL, использования подзапросов при модификации данных и индексов.

Оборудование: компьютерный класс.

Программное обеспечение: СУБД PostgreSQL, pgAdmin 4.

Практическое задание:

1. Создать запросы и представления на выборку данных к базе данных PostgreSQL (согласно индивидуальному заданию, часть 2 и 3).
2. Составить 3 запроса на модификацию данных (INSERT, UPDATE, DELETE) с использованием подзапросов.
3. Изучить графическое представление запросов и просмотреть историю запросов.
4. Создать простой и составной индексы для двух произвольных запросов и сравнить время выполнения запросов без индексов и с индексами. Для получения плана запроса использовать команду EXPLAIN.

Индивидуальное задание:

Вариант 8. БД «Аэропорт»

Описание предметной области: необходимо обеспечить продажу билетов на нужный рейс, при отсутствии билетов (необходимого количества билетов) предложить билет на ближайший рейс.

Рейсы выполняются по расписанию. Но есть рейсы, назначаемые на определенный период или разовые.

Рейс может иметь несколько транзитных посадок.

Билет может быть приобретен в кассе или онлайн. К базовой стоимости билета может быть дополнительная плата за выбор места, страховку багажа и т.п. Если билет приобретен в кассе, необходимо знать, в какой. Для каждой кассы известны номер и адрес. Кассы могут располагаться в различных населенных пунктах.

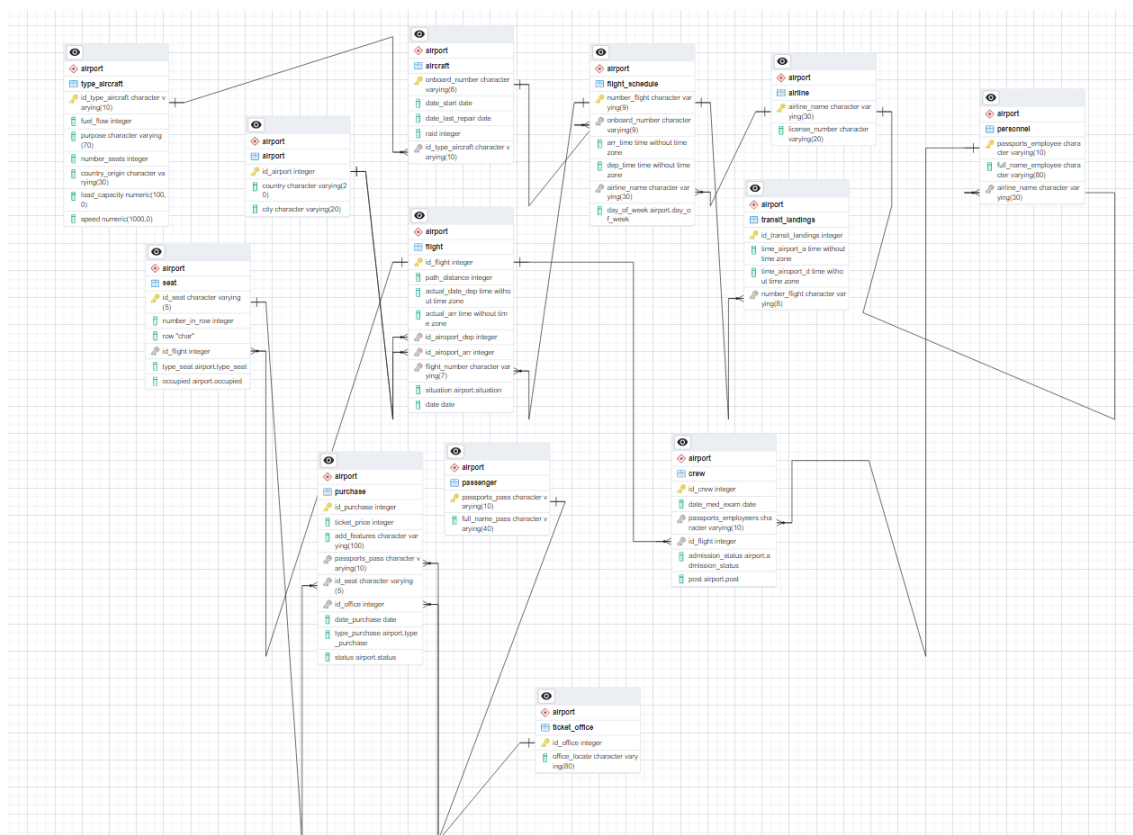
БД должна содержать следующий минимальный набор сведений: Бортовой номер самолета. Тип самолета. Количество мест. Страна. Производитель. Грузоподъемность. Скорость. Дата выпуска. Налёт в часах. Дата последнего ремонта. Назначение самолета. Расход топлива. Код экипажа. Паспортные данные членов экипажа. Номер рейса. Дата вылета. Время вылета. Аэропорт вылета. Аэропорт назначения. Расстояние. Транзитные посадки (прилет, вылет, аэропорт, время в аэропорту). ФИО пассажира. Паспортные данные. Номер места. Тип места. Цена билета. Касса продажи билета (возможен электронный билет) (номер и адрес).

Задание 2. Создать запросы:

- Определить расчетное время полета по всем маршрутам.
- Определить расход топлива по всем маршрутам.
- Вывести данные о том, сколько свободных мест оставалось в самолетах, совершавших полет по заданному из рейсов за вчерашний день.
- Рассчитать убытки компании за счет непроданных билетов за вчерашний день.
- Определить, какой тип самолетов чаще всего летал в заданный аэропорт назначения.
- Вывести список самолетов, “возраст” которых превышает средний “возраст” самолетов этого типа.
- Определить тип самолетов, летающих во все аэропорты назначения.

Задание 3. Создать представление:

- для пассажиров авиакомпании о рейсах в Москву на ближайшую неделю;
- количество самолетов каждого типа, летавшими за последний месяц.



Ход работы:

1. Создание запросов

- Определить расчетное время полета по всем маршрутам:

```
SELECT FLIGHT.ID_FLIGHT,
FLIGHT_SCHEDULE.DEP_TIME - FLIGHT_SCHEDULE.ARR_TIME AS
FLIGHT_DURATION
```

FROM airport.FLIGHT

```
JOIN airport.FLIGHT_SCHEDULE ON FLIGHT.flight_number =
FLIGHT_SCHEDULE.NUMBER_FLIGHT;
```

ЗапросИстория запросов

1 SELECT FLIGHT.ID_FLIGHT,
2 FLIGHT_SCHEDULE.DEP_TIME - FLIGHT_SCHEDULE.ARR_TIME AS FLIGHT_DURATION
3 FROM AIRPORT.FLIGHT
4 JOIN AIRPORT.FLIGHT_SCHEDULE ON FLIGHT.FLIGHT_NUMBER = FLIGHT_SCHEDULE.NUMBER_FLIGHT;

Data OutputСообщенияNotifications

+

📄

▼

📋

🗑️

🔄

⬇️

📶

	id_flight [PK] integer	flight_duration interval
1	2	03:20:00
2	3	04:08:00
3	1	00:50:00
4	8	03:20:00
5	9	00:59:00
6	10	01:00:00
7	11	00:39:00

- Определить расход топлива по всем маршрутам:
SELECT flight.flight_number,
type_aircraft.fuel_flow AS fuel_consumption
FROM airport.flight
JOIN airport.flight_schedule ON flight.flight_number = flight_schedule.number_flight
JOIN airport.aircraft ON flight_schedule.onboard_number = aircraft.onboard_number
JOIN airport.type_aircraft ON aircraft.id_type_aircraft = type_aircraft.id_type_aircraft
GROUP BY flight.flight_number, type_aircraft.fuel_flow;

запрос история запросов

```

1 SELECT flight.flight_number,
2       type_aircraft.fuel_flow AS fuel_consumption
3 FROM airport.flight
4 JOIN airport.flight_schedule ON flight.flight_number = flight_schedule.number_flight
5 JOIN airport.aircraft ON flight_schedule.onboard_number = aircraft.onboard_number
6 JOIN airport.type_aircraft ON aircraft.id_type_aircraft = type_aircraft.id_type_aircraft
7 GROUP BY flight.flight_number, type_aircraft.fuel_flow;

```

Data Output Сообщения Notifications

	flight_number character varying (7)	fuel_consumption integer
1	CX890	2774
2	AA123	2521
3	CZ567	2521
4	AF890	2196
5	AC234	2796
6	BA567	2267
7	AY456	2796

- Вывести данные о том, сколько свободных мест оставалось в самолетах, совершавших полет по заданному из рейсов за вчерашний день:

```

SELECT COUNT(*) AS free_seats
FROM airport.seat
WHERE id_flight = 1 AND occupied = 'Свободно'
AND id_flight IN (SELECT id_flight FROM airport.flight WHERE DATE(flight.date)
= DATE(NOW() - INTERVAL '1 DAY'));

```

Запрос История запросов

```

1 SELECT COUNT(*) AS free_seats
2 FROM airport.seat
3 WHERE id_flight = 1 AND occupied = 'Свободно'
4 AND id_flight IN (SELECT id_flight FROM airport.flight WHERE DATE(flight.date) = DATE(NOW() - INTERVAL '1 DAY'));
5

```

Data Output Сообщения Notifications

	free_seats bigint
1	14

- Рассчитать убытки компании за счет непроданных билетов за вчерашний день (так как у меня нет изначальной цены за место, то все, кто приобрели билет – заняли свои места. Поэтому подсчитала доход):

```

SELECT SUM(TICKET_PRICE) AS ДОХОД
FROM AIRPORT.PURCHASE
WHERE ID_SEAT IN
      (SELECT ID_SEAT
       FROM AIRPORT.SEAT

```

```

WHERE OCCUPIED = 'Занято'
AND ID_FLIGHT = 1 AND ID_FLIGHT IN
(SELECT ID_FLIGHT
FROM AIRPORT.FLIGHT
WHERE DATE(FLIGHT.DATE) =
DATE(NOW()) - INTERVAL '1 DAY')));

```

Запрос История запросов

```

1 SELECT SUM(TICKET_PRICE) AS доход
2 FROM AIRPORT.PURCHASE
3 WHERE ID_SEAT IN
4     (SELECT ID_SEAT
5      FROM AIRPORT.SEAT
6      WHERE OCCUPIED = 'Занято'
7      AND ID_FLIGHT = 1 AND ID_FLIGHT IN
8          (SELECT ID_FLIGHT
9           FROM AIRPORT.FLIGHT
10          WHERE DATE(FLIGHT.DATE) = DATE(NOW() - INTERVAL '1 DAY')));

```

Data Output Сообщения План выполнения x Notifications

	доход bigint
1	221167

- Определить, какой тип самолетов чаще всего летал в заданный аэропорт назначения:

```

SELECT id_type_aircraft, COUNT(*) AS Количество_рейсов
FROM airport.flight
JOIN airport.flight_schedule ON flight.flight_number = flight_schedule.number_flight
JOIN airport.aircraft ON flight_schedule.onboard_number = aircraft.onboard_number
WHERE flight.id_airoport_arr = 1
GROUP BY aircraft.id_type_aircraft
ORDER BY Количество_рейсов DESC
LIMIT 1;

```

Запрос История запросов

```

1 SELECT id_type_aircraft, COUNT(*) AS Количество_рейсов
2 FROM airport.flight
3 JOIN airport.flight_schedule ON flight.flight_number = flight_schedule.number_flight
4 JOIN airport.aircraft ON flight_schedule.onboard_number = aircraft.onboard_number
5 WHERE flight.id_airoport_arr = 1
6 GROUP BY aircraft.id_type_aircraft
7 ORDER BY Количество_рейсов DESC
8 LIMIT 1;

```

Data Output Сообщения План выполнения × Notifications

	id_type_aircraft character varying (10)	Количество_рейсов bigint
1	B737	1

- Вывести список самолетов, “возраст” которых превышает средний “возраст” самолетов этого типа:

```

SELECT a.onboard_number,
       EXTRACT(YEAR FROM AGE(CURRENT_DATE, a.date_start)) AS age,
       a.id_type_aircraft,
       t.avg_age
FROM airport.aircraft AS a
JOIN (
  SELECT id_type_aircraft, AVG(EXTRACT(YEAR FROM AGE(CURRENT_DATE,
date_start)))) AS avg_age
FROM airport.aircraft
GROUP BY id_type_aircraft
) AS t ON a.id_type_aircraft = t.id_type_aircraft
WHERE EXTRACT(YEAR FROM AGE(CURRENT_DATE, a.date_start)) > t.avg_age
ORDER BY age DESC;

```

Запрос История запросов				
<pre> 1 SELECT a.onboard_number, 2 EXTRACT(YEAR FROM AGE(CURRENT_DATE, a.date_start)) AS age, 3 a.id_type_aircraft, 4 t.avg_age 5 FROM airport.aircraft AS a 6 JOIN (7 SELECT id_type_aircraft, AVG(EXTRACT(YEAR FROM AGE(CURRENT_DATE, date_st 8 FROM airport.aircraft 9 GROUP BY id_type_aircraft 10) AS t ON a.id_type_aircraft = t.id_type_aircraft 11 WHERE EXTRACT(YEAR FROM AGE(CURRENT_DATE, a.date_start)) > t.avg_age 12 ORDER BY age DESC; 13 </pre>				
Data Output Сообщения Notifications				
	onboard_number [PK] character varying (6)	age double precision	id_type_aircraft character varying (10)	avg_age double precision
1	EY-456		3 G550	1.3
2	AY-234		2 L1011	1.1111111111111112
3	PY-164		2 C130	1.1333333333333333
4	OM-260		2 F16	1.0909090909090908
5	PU-267		2 C130	1.1333333333333333
6	VT-177		2 L1011	1.1111111111111112
7	WJ-479		2 A320	1.25
8	YG-256		2 F16	1.0909090909090908
9	IE-954		2 B737	1
10	OC-235		2 A320	1.25
11	GG-405		2 KC-135	1
12	WH-518		2 A320	1.25
13	GT-473		2 DHC6	1.125
14	DR-780		2 DHC6	1.125
15	FZ-132		2 A320	1.25
16	JM-113		2 G550	1.3
17	RT-895		2 E170	1
18	KP-335		2 L1011	1.1111111111111112

- Определить тип самолетов, летающих во все аэропорты назначения:
SELECT DISTINCT
a.country || ', ' || a.city AS destination,
at.id_type_aircraft
FROM
airport.flight f
JOIN airport.airport a ON f.id_airport_arr = a.id_airport
JOIN airport.flight_schedule fs ON f.flight_number = fs.number_flight
JOIN airport.aircraft ac ON fs.onboard_number = ac.onboard_number
JOIN airport.type_aircraft at ON ac.id_type_aircraft = at.id_type_aircraft
ORDER BY destination ASC;

Запрос

История запросов

```
1 SELECT DISTINCT
2   a.country || ', ' || a.city AS destination,
3   at.id_type_aircraft
4 FROM
5   airport.flight f
6   JOIN airport.airport a ON f.id_airoport_arr = a.id_airport
7   JOIN airport.flight_schedule fs ON f.flight_number = fs.number_flight
8   JOIN airport.aircraft ac ON fs.onboard_number = ac.onboard_number
9   JOIN airport.type_aircraft at ON ac.id_type_aircraft = at.id_type_aircraft
10 ORDER BY destination ASC;
11
```

Data Output

Сообщения

Notifications

	destination text	<div><div></div>id_type_aircraft [PK] character varying (10)</div>
1	Golden Valley, Stepnogorsk	E170
2	Lunaria, Zelenogradsk	C130
3	Lunaria, Zelenogradsk	G550
4	Mountain, Rechnoy	DHC6
5	Oceania, Rainbow	A320
6	Oceania, Rainbow	C130
7	Oceania, Rainbow	E170
8	The Ice Empire, Lesogorsk	L1011
9	The Solar Republic, Urban	F16
10	The Solar Republic, Urban	KC-135
11	Verdantia, Novopolis	B737
12	Verdantia, Novopolis	L1011
13	Wind Republic, Peaceful	A320
14	Wind Republic, Peaceful	B737
15	Wind Republic, Peaceful	KC-135
16	Wind Republic, Peaceful	L1011
17	Zemland, Solnechnogorsk	DHC6
18	Zemland, Solnechnogorsk	G550
19	Zemland, Solnechnogorsk	L1011

2. Запросы на модификацию данных:

INSERT

До:

<pre> 1 SELECT * FROM airport.purchase 2 ORDER BY id_purchase ASC </pre>		Data Output		Сообщения		Notifications	
id_purchase	ticket_price	add_features	passports_pass	id_seat	id_office	date_purchase	type_purchase
321	341	3290	Food	8146372675	4C40	2022-05-25	Электронный
322	342	3738	Food	8198263915	1C33	2022-11-09	В кассе
323	343	5319	With an animal	8212342258	2C37	2022-07-12	Электронный
324	344	3837	With a baby	8236949766	4A3	2022-04-09	В кассе
325	345	9507	Food	8276742919	4B41	2022-05-21	Электронный
326	346	5976	With a baby	8301039088	5B8	2022-08-27	В кассе
327	347	5692	Food	8303880794	1B34	2022-01-03	Электронный
328	348	7252	[null]	8304815812	4C3	2022-05-02	В кассе
329	349	9957	[null]	8347299351	1A34	2022-05-06	Электронный
330	350	9863	With an animal	8348224943	3C36	2022-03-05	Электронный
331	351	4367	With a baby	8403587980	2C32	2022-08-28	Электронный
332	352	6620	[null]	8437237211	1F10	2022-05-28	В кассе
333	353	9846	With a baby	8492202006	2C33	2022-05-01	Электронный
334	354	9083	With a baby	8510128888	5C42	2022-11-07	В кассе
335	355	12889	[null]	8513678387	3F2	2022-03-14	Электронный
336	357	4470	[null]	8589680552	3D34	2022-11-19	В кассе
337	358	7518	[null]	8649323912	1F39	2022-06-08	В кассе
338	359	13679	With a baby	8653562883	4A38	2022-04-23	Электронный
339	360	8000	With a baby	8690996963	5D11	2022-10-05	В кассе
340	361	11835	Food	8723026190	1A8	2022-06-04	Электронный
341	362	11888	Food	8781538923	2D35	2022-03-23	В кассе
342	363	13186	[null]	8783692994	3A11	2022-02-09	Электронный
343	364	5576	Food	8808967742	1B43	2022-08-24	В кассе
344	365	4984	[null]	8814743491	1E36	2022-06-08	В кассе

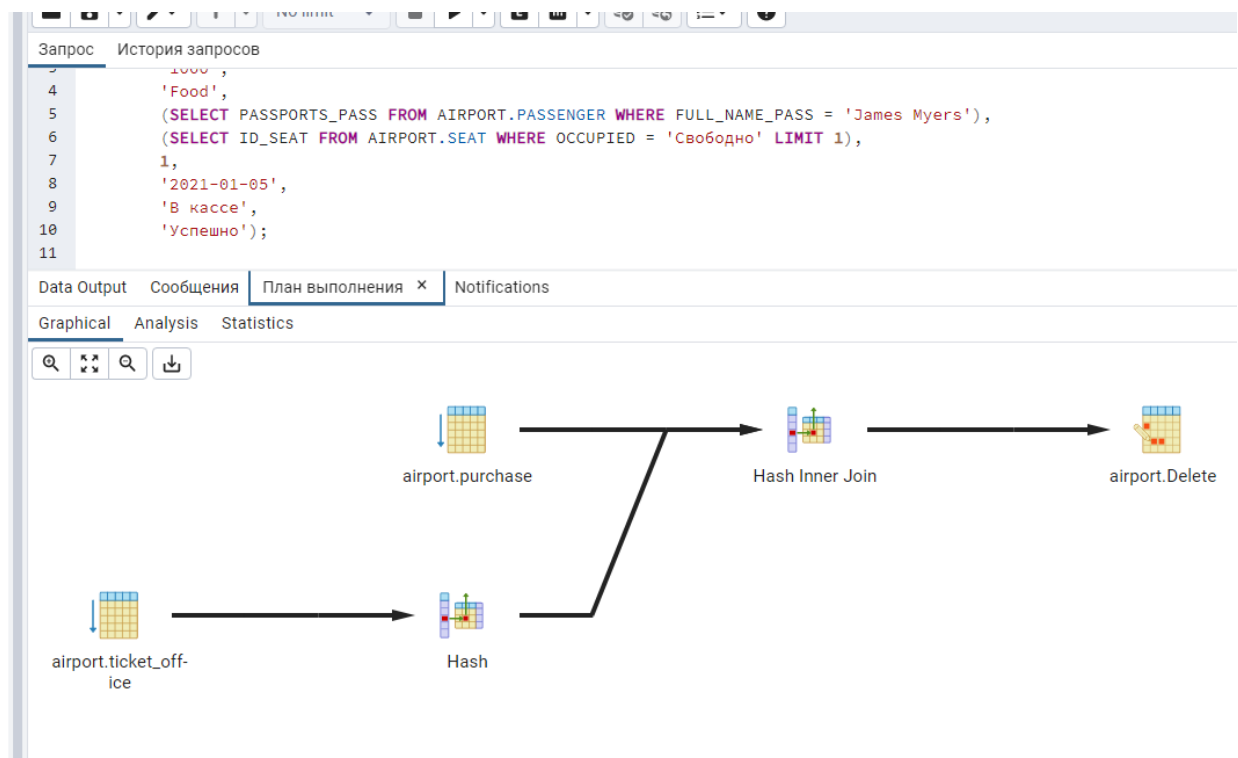
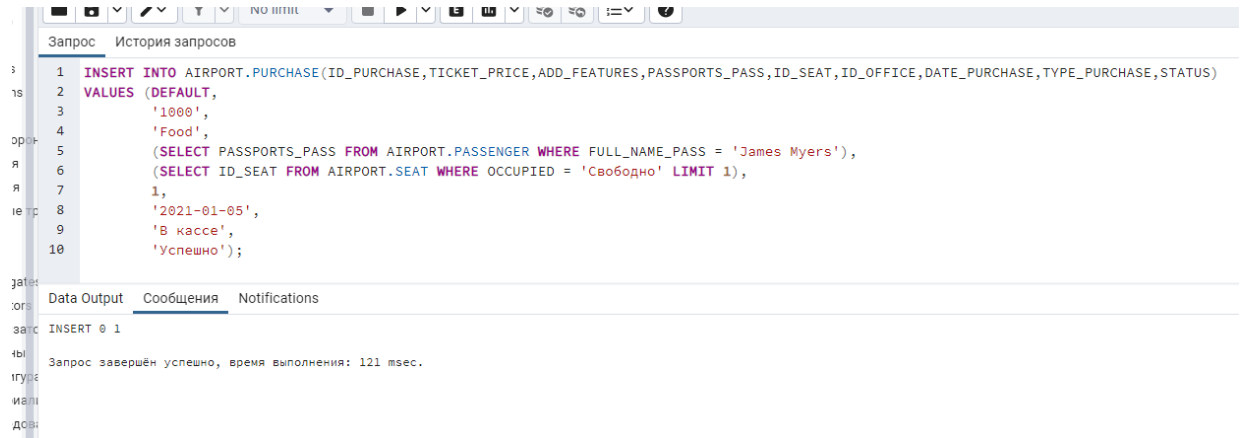
INSERT INTO

AIRPORT.PURCHASE(ID_PURCHASE,TICKET_PRICE,ADD_FEATURES,PASSPORTS_PASS,ID_SEAT,ID_OFFICE,DATE_PURCHASE,TYPE_PURCHASE,STATUS)
VALUES (DEFAULT,

```

        '1000',
        'Food',
        (SELECT PASSPORTS_PASS FROM AIRPORT.PASSENGER WHERE
FULL_NAME_PASS = 'James Myers'),
        (SELECT ID_SEAT FROM AIRPORT.SEAT WHERE OCCUPIED =
'Свободно' LIMIT 1),
        1,
        '2021-01-05',
        'В кассе',
        'Успешно');

```



После:

1	SELECT *	FROM airport.purchase
2	ORDER BY	id_purchase ASC

Data Output	Сообщения	Notifications
-------------	-----------	---------------

	id_purchase (PK) integer	ticket_price integer	add_features character varying (100)	passports_pass character varying (10)	id_seat character varying (5)	id_office integer	date_purchase date	type_purchase airport.type_purchase	status airport.status
323	343	5319	With an animal	8212342258	2C37	6	2022-07-12	Электронный	Ожидает оплаты
324	344	3837	With a baby	8236949766	4A2	11	2022-04-09	В кассе	Возврат
325	345	9507	Food	8276742919	4B41	5	2022-05-21	Электронный	Успешно
326	346	5976	With a baby	8301039088	5B8	12	2022-08-27	В кассе	Неуспешно
327	347	5692	Food	8303880794	1B34	12	2022-01-03	Электронный	Успешно
328	348	7252	[null]	8304815812	4C3	14	2022-05-02	В кассе	Возврат
329	349	9957	[null]	8347299351	1A34	18	2022-05-06	Электронный	Успешно
330	350	9863	With an animal	8348224943	3C36	1	2022-03-05	Электронный	Неуспешно
331	351	4367	With a baby	8403587980	2C32	4	2022-08-28	Электронный	Ожидает оплаты
332	352	6620	[null]	8437237211	1F10	13	2022-05-28	В кассе	Успешно
333	353	9846	With a baby	8492202006	2C33	17	2022-05-01	Электронный	Неуспешно
334	354	9083	With a baby	8510128888	5C42	9	2022-11-07	В кассе	Возврат
335	355	12889	[null]	8513678387	3F2	9	2022-03-14	Электронный	Неуспешно
336	357	4470	[null]	8589680552	3D34	17	2022-11-19	В кассе	Ожидает оплаты
337	358	7518	[null]	8649323912	1F39	15	2022-06-08	В кассе	Успешно
338	359	13079	With a baby	8653582883	4A38	6	2022-04-23	Электронный	Ожидает оплаты
339	360	8000	With a baby	8690996963	5D11	19	2022-10-05	В кассе	Успешно
340	361	11835	Food	8723026190	1A8	19	2022-06-04	Электронный	Успешно
341	362	11888	Food	8781538923	2D35	17	2022-03-23	В кассе	Возврат
342	363	13186	[null]	8783692994	3A11	3	2022-02-09	Электронный	Ожидает оплаты
343	364	5576	Food	8808967742	1B43	2	2022-08-24	В кассе	Возврат
344	365	4984	[null]	8814743491	1E36	7	2022-06-08	В кассе	Успешно
345	367	1000	Food	1455889696	6E37	1	2021-01-05	В кассе	Успешно

Total rows: 345 of 345 Query complete 00:00:00.343

UPDATE

До:

1	SELECT *	FROM airport.flight
2	ORDER BY	id_flight ASC

Data Output	Сообщения	Notifications
-------------	-----------	---------------

	id_flight (PK) integer	path_distance integer	actual_date_dep time without time zone	actual_arr time without time zone	id_airoport_dep integer	id_airoport_arr integer	flight_number character varying (7)	situation airport.situation	date date
1	1	3994	02:05:00	01:15:00	2	1	AA123	Прилетел	2023-03-22
2	2	1690	08:10:00	04:50:00	3	2	AC234	Посадка	2022-03-22
3	3	5635	10:50:00	06:50:00	4	3	AF890	Задерживается	2022-03-22
4	8	1698	18:15:00	14:51:00	4	3	AY456	Запланирован	2022-03-22
5	9	1234	06:20:00	05:25:00	5	4	BA567	Посадка	2022-03-22
6	10	2111	14:00:00	12:55:00	6	5	CX890	Задерживается	2022-03-22
7	11	1341	[null]	[null]	7	6	CZ567	Отменён	2022-03-22
8	32	1021	22:56:00	17:25:00	12	7	DL789	Запланирован	2023-07-08
9	33	2377	08:42:00	07:58:00	8	4	EK456	Вылетел	2023-12-11
10	34	4289	03:50:00	01:30:00	11	9	KE789	Прилетел	2023-10-25
11	35	1560	04:34:00	02:00:00	9	3	KLM567	Отменён	2023-03-17
12	36	2601	04:02:00	02:31:00	3	7	LH123	Запланирован	2023-07-03
13	37	3227	07:00:00	02:51:00	6	5	MU123	Вылетел	2023-09-11
14	38	2975	03:54:00	00:19:00	13	1	OS123	Вылетел	2023-10-21
15	39	1286	12:33:00	11:13:00	1	10	QR234	Задерживается	2023-10-16
16	40	2826	05:46:00	02:03:00	5	2	SA456	Задерживается	2023-06-06
17	41	2723	02:56:00	00:54:00	2	4	SQ234	Посадка	2023-06-08
18	42	4906	07:47:00	03:55:00	4	7	TG890	Посадка	2023-07-14
19	43	3897	11:57:00	10:23:00	10	7	TK789	Прилетел	2023-02-02

UPDATE airport.flight

SET actual_date_dep = actual_date_dep + INTERVAL '1 DAY'

WHERE id_flight = (

SELECT id_flight FROM airport.purchase WHERE passports_pass = '1259298477');

Запрос

История запросов

1

UPDATE airport.flight

2

SET actual_date_dep = actual_date_dep + INTERVAL '1 DAY'

3

WHERE id_flight = (

4

SELECT id_flight FROM airport.purchase WHERE passports_pass = '1259298477');

5

Data Output

Сообщения

Notifications

UPDATE 19

Запрос завершён успешно, время выполнения: 102 мсек.

Запрос

История запросов

1

UPDATE airport.flight

2

SET actual_date_dep = actual_date_dep + INTERVAL '1 DAY'

3

WHERE id_flight = (

4

SELECT id_flight FROM airport.purchase WHERE passports_pass = '1259298477');

5

Data Output

Сообщения

План выполнения ×

Notifications

Graphical

Analysis

Statistics

🔍

↕

🔍

📄

purchase

→

flight

→

Update

После:

Запрос

История запросов

1 SELECT * FROM airport.flight

2 ORDER BY id_flight ASC

Data Output

Сообщения

Notifications

	id_flight [PK] integer	path_distance integer	actual_date_dep time without time zone	actual_arr time without time zone	id_airport_dep integer	id_airport_arr integer	flight_number character varying (7)	situation airport.situation	date date
1	1	3994	02:05:00	01:15:00	2	1	AA123	Прилетел	2023-03-22
2	2	1690	08:10:00	04:50:00	3	2	AC234	Посадка	2022-03-22
3	3	5635	10:50:00	06:50:00	4	3	AF890	Задерживается	2022-03-22
4	8	1698	18:15:00	14:51:00	4	3	AY456	Запланирован	2022-03-22
5	9	1234	06:20:00	05:25:00	5	4	BA567	Посадка	2022-03-22
6	10	2111	14:00:00	12:55:00	6	5	CX890	Задерживается	2022-03-22
7	11	1341	[null]	[null]	7	6	CZ567	Отменён	2022-03-22
8	32	1021	22:56:00	17:25:00	12	7	DL789	Запланирован	2023-07-08
9	33	2377	08:42:00	07:58:00	8	4	EK456	Вылетел	2023-12-11
10	34	4289	03:50:00	01:30:00	11	9	KE789	Прилетел	2023-10-25
11	35	1560	04:34:00	02:00:00	9	3	KLM567	Отменён	2023-03-17
12	36	2601	04:02:00	02:31:00	3	7	LH123	Запланирован	2023-07-03
13	37	3227	07:00:00	02:51:00	6	5	MU123	Вылетел	2023-09-11
14	38	2975	03:54:00	00:19:00	13	1	OS123	Вылетел	2023-10-21
15	39	1286	12:33:00	11:13:00	1	10	QR234	Задерживается	2023-10-16
16	40	2826	05:46:00	02:03:00	5	2	SA456	Задерживается	2023-06-06
17	41	2723	02:56:00	00:54:00	2	4	SQ234	Посадка	2023-06-08
18	42	4906	07:47:00	03:55:00	4	7	TG890	Посадка	2023-07-14
19	43	3897	11:57:00	10:23:00	10	7	TK789	Прилетел	2023-02-02

DELETE

До:

Запрос

История запросов

1

SELECT * FROM airport.purchase

2

ORDER BY id_purchase ASC

Data Output

Сообщения

Notifications

	id_purchase [PK] integer	ticket_price integer	add_features character varying (100)	passport_pass character varying (10)	id_seat character varying (5)	id_office integer	date_purchase date	type_purchase airport.type_purchase	status airport.status
1	1	8000	Food	4319932537	4B1	18	2023-01-01	Электронный	Успешно
2	6	9020	[null]	1042509636	4A37	1	2022-02-18	Электронный	Возврат
3	7	6683	With an animal	1086375627	5B40	8	2022-02-26	В кассе	Возврат
4	8	5817	Food	1144203113	3B3	12	2022-11-15	В кассе	Ожидает оплаты
5	9	13685	Food	1166137371	1A32	8	2022-08-01	В кассе	Успешно
6	10	12941	With a baby	1199069686	4A33	3	2022-04-22	Электронный	Успешно
7	11	7941	[null]	1259298477	3D43	2	2022-09-01	Электронный	Ожидает оплаты
8	12	14140	Food	1270625735	6A35	9	2022-04-25	В кассе	Возврат
9	13	7920	[null]	1290154699	3B2	19	2022-05-19	Электронный	Успешно
10	14	10887	Food	1301646243	2A32	6	2022-11-16	Электронный	Неуспешно
11	15	7857	[null]	1311284000	2C10	13	2022-03-24	В кассе	Неуспешно
12	16	14624	With an animal	1312494123	3E38	19	2022-03-05	Электронный	Ожидает оплаты
13	17	14603	With a baby	1323075339	3D37	14	2022-08-10	Электронный	Успешно
14	18	7204	Food	1365270282	3C38	17	2022-02-12	В кассе	Ожидает оплаты
15	19	10198	Food	1397593406	5A2	9	2022-10-08	В кассе	Успешно
16	21	6326	[null]	1455889696	3D99	2	2022-03-04	В кассе	Неуспешно
17	23	13892	[null]	1502707878	4E1	5	2022-02-20	Электронный	Неуспешно
18	24	6940	[null]	1511588854	3E35	2	2022-08-08	В кассе	Возврат
19	25	6329	[null]	1530634632	1A43	18	2022-12-12	В кассе	Ожидает оплаты
20	26	3967	With a baby	1533937430	6E34	5	2022-05-28	В кассе	Возврат
21	27	8746	[null]	1566392781	6D42	15	2022-06-05	Электронный	Возврат
22	28	13675	Food	1568089007	3F38	10	2022-02-07	Электронный	Неуспешно
23	29	12422	With a baby	1573011360	3F32	6	2022-07-08	Электронный	Неуспешно
24	30	8616	Food	1625996801	1F37				

Total rows: 345 of 345 Query complete 00:00:00.839

✓

Запрос выполнен успешно. Общее время выполнения: 839 мсек. ошибок: 0

DELETE FROM airport.purchase
WHERE id_office IN (
SELECT id_office FROM airport.ticket_office WHERE office_locate = '2727 Willowcrest Lane');

ЗапросИстория запросов

1DELETE FROM airport.purchase

2WHERE id_office IN (

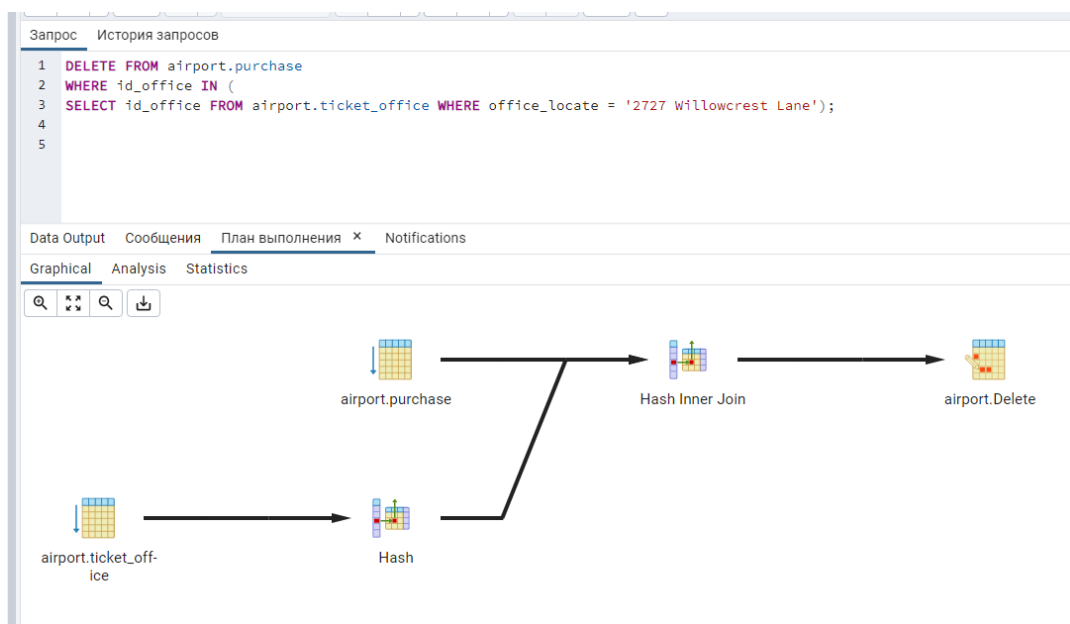
3SELECT id_office FROM airport.ticket_office WHERE office_locate = '2727 Willowcrest Lane')

4

Data OutputСообщенияNotifications

DELETE 17

Запрос завершен успешно, время выполнения: 105 msec.



После:

1	SELECT * FROM airport.purchase
2	ORDER BY id_purchase ASC

Data Output	Сообщения	Notifications
-------------	-----------	---------------

	id_purchase [PK] integer	ticket_price integer	add_features character varying (100)	passports_pass character varying (10)	id_seat character varying (5)	id_office integer	date_purchase date	type_purchase airport_type_purchase	status airport_status
1	1	8000	Food	4319932537	4B1	18	2023-01-01	Электронный	Успешно
2	6	9020	[null]	1042509636	4A37	1	2022-02-18	Электронный	Возврат
3	7	6683	With an animal	1086375627	5B40	8	2022-02-26	В кассе	Возврат
4	8	5817	Food	1144203113	3B3	12	2022-11-15	В кассе	Ожидает оплаты
5	9	13685	Food	1166137371	1A32	8	2022-08-01	В кассе	Успешно
6	10	12941	With a baby	1199069686	4A33	3	2022-04-22	Электронный	Успешно
7	11	7941	[null]	1259298477	3D43	2	2022-09-01	Электронный	Ожидает оплаты
8	12	14140	Food	1270625735	6A35	9	2022-04-25	В кассе	Возврат
9	13	7920	[null]	1290154699	3B2	19	2022-05-19	Электронный	Успешно
10	14	10887	Food	1301646243	2A32	6	2022-11-16	Электронный	Неуспешно
11	15	7857	[null]	1311284000	2C10	13	2022-03-24	В кассе	Неуспешно
12	16	14824	With an animal	1312494123	3E38	19	2022-03-05	Электронный	Ожидает оплаты
13	17	14603	With a baby	1323075339	3D37	14	2022-08-10	Электронный	Успешно
14	18	7204	Food	1365270282	3C38	17	2022-02-12	В кассе	Ожидает оплаты
15	19	10198	Food	1397593406	5A2	9	2022-10-08	В кассе	Успешно
16	24	6940	[null]	1511588854	3E35	2	2022-08-08	В кассе	Возврат
17	25	6329	[null]	1530634632	1A43	18	2022-12-12	В кассе	Ожидает оплаты
18	26	3967	With a baby	1533937430	6E34	5	2022-05-28	В кассе	Возврат
19	30	8616	Food	1625996801	1F37	9	2022-10-05	Электронный	Возврат
20	31	11767	With an animal	1635655982	1E38	7	2022-04-13	Электронный	Неуспешно
21	32	11323	[null]	1788922757	1C43	4	2022-12-03	В кассе	Ожидает оплаты
22	33	14200	[null]	1804317641	4A34	9	2022-07-24	В кассе	Неуспешно
23	34	13439	With an animal	1812376933	2A36	7	2022-03-23	В кассе	Ожидает оплаты

Total rows: 328 of 328 Query complete 00:00:00.505 ✓ Запрос выполнен успешно. Общее время выполнения: 505 msec.

3. Представления

- для пассажиров авиакомпании о рейсах в Москву на ближайшую неделю;

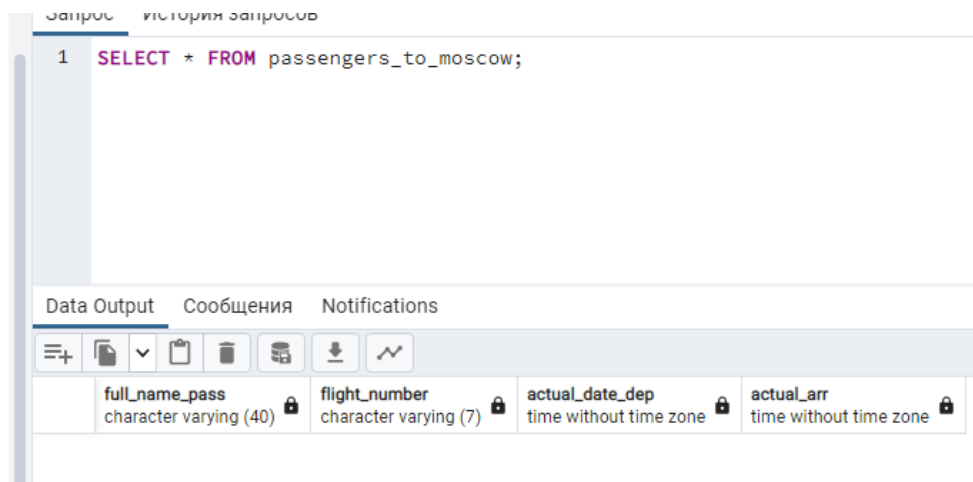
```
CREATE OR REPLACE VIEW passengers_to_moscow AS
SELECT p.full_name_pass, f.flight_number, f.actual_date_dep, f.actual_arr
FROM airport.flight f
JOIN airport.seat s ON f.id_flight = s.id_flight
JOIN airport.purchase pu ON s.id_seat = pu.id_seat
JOIN airport.passenger p ON pu.passports_pass = p.passports_pass
JOIN airport.airport a ON f.id_airport_arr = a.id_airport
WHERE a.city = 'Moscow' AND f.date BETWEEN NOW() AND NOW() + INTERVAL '1 WEEK';
```

Запрос	История запросов
--------	------------------

1	CREATE OR REPLACE VIEW passengers_to_moscow AS
2	SELECT p.full_name_pass, f.flight_number, f.actual_date_dep, f.actual_arr
3	FROM airport.flight f
4	JOIN airport.seat s ON f.id_flight = s.id_flight
5	JOIN airport.purchase pu ON s.id_seat = pu.id_seat
6	JOIN airport.passenger p ON pu.passports_pass = p.passports_pass
7	JOIN airport.airport a ON f.id_airport_arr = a.id_airport
8	WHERE a.city = 'Moscow' AND f.date BETWEEN NOW() AND NOW() + INTERVAL '1 WEEK';
9	

Data Output	Сообщения	Notifications
-------------	-----------	---------------

CREATE VIEW
Запрос завершен успешно, время выполнения: 170 msec.



- количество самолетов каждого типа, летавшими за последний месяц.

```
CREATE VIEW flights_per_aircraft_type AS
SELECT COUNT(DISTINCT a.onboard_number) AS num_flights, t.id_type_aircraft AS
aircraft_type_id
FROM airport.aircraft a
JOIN airport.flight_schedule fs ON a.onboard_number = fs.onboard_number
JOIN airport.flight f ON fs.number_flight = f.flight_number
JOIN airport.type_aircraft t ON a.id_type_aircraft = t.id_type_aircraft
WHERE f.date >= CURRENT_TIMESTAMP - INTERVAL '1 month'
GROUP BY t.id_type_aircraft;
```



Запрос

История запросов









1

SELECT * FROM flights_per_aircraft_type;

Data Output

Сообщения

Notifications



	num_flights bigint	aircraft_type_id character varying (10)
1	1	A320
2	1	B737
3	1	C130
4	1	DHC6
5	1	E170
6	1	KC-135
7	2	L1011

4. Запросы с индексами

Простой индекс

EXPLAIN SELECT id_flight FROM airport.flight WHERE flight_number LIKE 'A%';

CREATE INDEX flight_number_idx ON airport.flight (flight_number);

Общее время выполнения запроса без индекса: 471 msec.

Общее время выполнения запроса с индексом: 84 msec.

Запрос

История запросов

1

EXPLAIN SELECT id_flight FROM airport.flight WHERE flight_number LIKE 'A%';

2

3

CREATE INDEX flight_number_idx ON airport.flight (flight_number);

4

Data Output

Сообщения

План выполнения ×

Notifications

≡

📄

▼

📋

🗑

🔍

⬇

📈

QUERY PLAN

text

🔒

1

Seq Scan on flight (cost=0.00..1.24 rows=3 width=4)

2

Filter: ((flight_number)::text ~~ 'A%':text)

Составная индексация:

```
EXPLAIN SELECT id_seat,  
flight.date  
FROM airport.seat  
JOIN airport.flight ON flight.id_flight = seat.id_flight WHERE occupied = 'Свободно';
```

```
CREATE INDEX flight_occupied_idx ON airport.seat (id_flight, occupied);
```

Общее время выполнения запроса без индекса: 119 msec.

Общее время выполнения запроса с индексом: 97 msec.

The screenshot shows a database query tool interface. At the top, there are tabs for 'Запрос' (Query) and 'История запросов' (Query History). The 'Запрос' tab is active, displaying the following SQL query:

```
1 EXPLAIN SELECT id_seat,  
2 flight.date  
3 FROM airport.seat  
4 JOIN airport.flight ON flight.id_flight = seat.id_flight WHERE occupied = 'Свободно';  
5  
6 CREATE INDEX flight_occupied_idx ON airport.seat (id_flight, occupied);  
7
```

Below the query editor, there are tabs for 'Data Output', 'Сообщения' (Messages), 'План выполнения' (Execution Plan), and 'Notifications'. The 'План выполнения' tab is active, showing the query plan for the first query (the EXPLAIN statement). The plan is as follows:

	QUERY PLAN
1	Hash Join (cost=1.43..17.02 rows=324 width=8)
2	Hash Cond: (seat.id_flight = flight.id_flight)
3	-> Seq Scan on seat (cost=0.00..14.55 rows=324 width=8)
4	Filter: (occupied = 'Свободно':airport.occupied)
5	-> Hash (cost=1.19..1.19 rows=19 width=8)
6	-> Seq Scan on flight (cost=0.00..1.19 rows=19 width=8)

Вывод: создание запросов на выборку данных и представлений, а также использование подзапросов при модификации данных являются стандартными задачами при работе с базами данных, и их практическое освоение позволит улучшить производительность работы с БД.

Изучение графического представления запросов и просмотр истории запросов также являются важными навыками для разработчика баз данных, поскольку позволяют улучшить эффективность работы с запросами и уменьшить время на их выполнение.

Создание индексов является важной задачей для оптимизации работы с БД, поскольку ускоряет выполнение запросов на выборку данных. При этом важно понимать, как правильно выбирать столбцы для индексации и какие типы индексов лучше использовать в конкретных случаях. Использование команды EXPLAIN позволяет оценить эффективность индексов и улучшить производительность работы с БД.