

Министерство науки и высшего образования РФ
Федеральное государственное автономное
образовательное учреждение высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт космических и информационных технологий

институт

Программная инженерия

кафедра

ОТЧЕТ О ПРАКТИЧЕСКОЙ РАБОТЕ №5

Изменение данных

тема

Преподаватель

Студент КИ23-17/16

032318988

номер зачетной книжки

подпись, дата

подпись, дата

Вожжов А.Д.

инициалы, фамилия

Александров Е.А.

инициалы, фамилия

Красноярск 2025

1 Цель

Изучить основы изменения данных. Выполнить указанные в файле задания.

2 Ход работы

Результат выполнения заданий показан на рисунках с 1 по 8.

```
demo=# WITH add_row AS (  
demo(#   INSERT INTO aircrafts_tmp  
demo(#   SELECT * FROM aircrafts  
demo(#   RETURNING aircraft_code, model, range, current_timestamp, 'INSERT'  
demo(# )  
demo=# INSERT INTO aircrafts_log  
demo=# SELECT aircraft_code, model, range, current_timestamp, 'INSERT' FROM add_row;  
INSERT 0 9
```

Рисунок 1 – Задание 2

```
demo=# CREATE TABLE seats_copy (LIKE seats INCLUDING ALL);  
CREATE TABLE  
demo=# INSERT INTO seats_copy (aircraft_code, seat_no, fare_conditions)  
demo=# VALUES ('SU9', '1A', 'Economy')  
demo=# ON CONFLICT (aircraft_code, seat_no) DO NOTHING;  
INSERT 0 1  
demo=# INSERT INTO seats_copy (aircraft_code, seat_no, fare_conditions)  
demo=# VALUES ('SU9', '1A', 'Economy')  
demo=# ON CONFLICT ON CONSTRAINT seats_copy_pkey DO NOTHING;  
INSERT 0 0
```

Рисунок 2 – Задание 4

```
demo=# COPY aircrafts_tmp FROM STDIN WITH ( FORMAT csv );
Вводите данные для копирования, разделяя строки переводом строки.
Закончите ввод строкой '\.' или сигналом EOF.
>> IL9, Ilyushin IL96, 9800
>> I93, Ilyushin IL96-300, 9800
>> \.
COPY 2
demo=# SELECT * FROM aircrafts_tmp;
```

aircraft_code	model	range
773	Boeing 777-300	11100
763	Boeing 767-300	7900
320	Airbus A320-200	5700
321	Airbus A321-200	5600
319	Airbus A319-100	6700
733	Boeing 737-300	4200
CN1	Cessna 208 Caravan	1200
CR2	Bombardier CRJ-200	2700
SU9	Sukhoi SuperJet-100	6000
IL9	Ilyushin IL96	9800
I93	Ilyushin IL96-300	9800

(11 строк)

Рисунок 3 – Задание 6

```
demo=# SELECT flight_no, flight_id, departure_city,
demo=# arrival_city, scheduled_departure
demo=# FROM flights_v
demo=# WHERE scheduled_departure
demo=# BETWEEN bookings.now() AND bookings.now() + INTERVAL '15 days'
demo=# AND ( departure_city, arrival_city ) IN
demo=# ( ( 'Красноярск', 'Москва' ),
demo=# ( 'Москва', 'Сочи' ),
demo=# ( 'Сочи', 'Москва' ),
demo=# ( 'Сочи', 'Красноярск' )
demo=# )
demo=# ORDER BY departure_city, arrival_city, scheduled_departure;
```

flight_no	flight_id	departure_city	arrival_city	scheduled_departure
PG0548	13841	Красноярск	Москва	Fri Oct 14 11:40:00 2016 +07
PG0548	13829	Красноярск	Москва	Sat Oct 15 11:40:00 2016 +07
PG0548	13860	Красноярск	Москва	Sun Oct 16 11:40:00 2016 +07
PG0548	13826	Красноярск	Москва	Mon Oct 17 11:40:00 2016 +07
PG0548	13839	Красноярск	Москва	Tue Oct 18 11:40:00 2016 +07

Рисунок 4 – Задание 8

```

demo=# WITH sell_tickets AS
demo-# ( INSERT INTO ticket_flights_tmp
demo-# ( ticket_no, flight_id, fare_conditions, amount )
demo-# VALUES ( '1234567890123', 13829, 'Economy', 10500 ),
demo-# ( '1234567890123', 4728, 'Economy', 3400 ),
demo-# ( '1234567890123', 30523, 'Economy', 3400 ),
demo-# ( '1234567890123', 7757, 'Economy', 3400 ),
demo-# ( '1234567890123', 30829, 'Economy', 12800 )
demo-# RETURNING *
demo-# )
demo-# UPDATE tickets_directions td
demo-# SET last_ticket_time = current_timestamp,
demo-# tickets_num = tickets_num +
demo-# ( SELECT count( * )
demo-# FROM sell_tickets st, flights_v f
demo-# WHERE st.flight_id = f.flight_id
demo-# AND f.departure_city = td.departure_city
demo-# AND f.arrival_city = td.arrival_city
demo-# )
demo-# WHERE ( td.departure_city, td.arrival_city ) IN
demo-# ( SELECT departure_city, arrival_city
demo-# FROM flights_v
demo-# WHERE flight_id IN ( SELECT flight_id FROM sell_tickets )
demo-# );
UPDATE 4

```

Рисунок 5 – Задание 8

```

demo=# SELECT departure_city AS dep_city,
demo=#   arrival_city AS arr_city,
demo=#   last_ticket_time,
demo=#   tickets_num AS num
demo=# FROM tickets_directions
demo=# WHERE tickets_num > 0
demo=# ORDER BY departure_city, arrival_city;

```

dep_city	arr_city	last_ticket_time	num
Красноярск	Москва	Fri Mar 28 22:52:58.771408 2025	1
Москва	Сочи	Fri Mar 28 22:52:58.771408 2025	2
Сочи	Красноярск	Fri Mar 28 22:52:58.771408 2025	1
Сочи	Москва	Fri Mar 28 22:52:58.771408 2025	1

(4 строки)

```

demo=# SELECT * FROM ticket_flights_tmp;

```

ticket_no	flight_id	fare_conditions	amount
1234567890123	13829	Economy	10500.00
1234567890123	4728	Economy	3400.00
1234567890123	30523	Economy	3400.00
1234567890123	7757	Economy	3400.00
1234567890123	30829	Economy	12800.00

(5 строк)

Рисунок 6 – Задание 8

```

demo=# CREATE TABLE tickets_directions (
demo(#   departure_city TEXT,
demo(#   arrival_city TEXT,
demo(#   fare_conditions TEXT,
demo(#   last_ticket_time TIMESTAMP,
demo(#   tickets_num INTEGER,
demo(#   PRIMARY KEY (departure_city, arrival_city, fare_conditions)
demo(# );
CREATE TABLE
demo=#
demo=# WITH sell_tickets AS (
demo(#   INSERT INTO ticket_flights_tmp (ticket_no, flight_id, fare_conditions, amount)
demo(#   VALUES
demo(#       ('1234567890123', 13829, 'Economy', 10500), -- Красноярск - Москва
demo(#       ('1234567890123', 4728, 'Economy', 3400), -- Москва - Сочи
demo(#       ('1234567890123', 30523, 'Economy', 3400), -- Сочи - Москва
demo(#       ('1234567890123', 7757, 'Business', 6800), -- Москва - Сочи (добавлен Business)
demo(#       ('1234567890123', 30829, 'Comfort', 15000) -- Сочи - Красноярск (добавлен Comfort)
demo(#   RETURNING *
demo(# )
demo=# UPDATE tickets_directions td
demo=# SET
demo(#   last_ticket_time = current_timestamp,
demo(#   tickets_num = COALESCE(tickets_num, 0) + (
demo(#       SELECT COUNT(*)
demo(#       FROM sell_tickets st
demo(#       JOIN flights_v f ON st.flight_id = f.flight_id
demo(#       WHERE f.departure_city = td.departure_city
demo(#       AND f.arrival_city = td.arrival_city
demo(#       AND st.fare_conditions = td.fare_conditions
demo(#   )
demo=# WHERE (td.departure_city, td.arrival_city, td.fare_conditions) IN (
demo(#   SELECT f.departure_city, f.arrival_city, st.fare_conditions
demo(#   FROM sell_tickets st
demo(#   JOIN flights_v f ON st.flight_id = f.flight_id
demo(# );

```

Рисунок 7 – Задание 8

```

demo=# INSERT INTO seats ( aircraft_code, seat_no, fare_conditions )
demo=# SELECT aircraft_code, seat_row || letter, fare_condition
demo=# FROM
demo=# -- компоновки салонов
demo=# ( VALUES ( 'SU9', 3, 20, 'F' ),
demo(#   ( '773', 5, 30, 'I' ),
demo(#   ( '763', 4, 25, 'H' ),
demo(#   ( '733', 3, 20, 'F' ),
demo(#   ( '320', 5, 25, 'F' ),
demo(#   ( '321', 4, 20, 'F' ),
demo(#   ( '319', 3, 20, 'F' ),
demo(#   ( 'CN1', 0, 10, 'B' ),
demo(#   ( 'CR2', 2, 15, 'D' )
demo(# ) AS aircraft_info ( aircraft_code, max_seat_row_business, max_seat_row_economy, max_letter )
demo=# CROSS JOIN
demo=# -- классы обслуживания
demo=# ( VALUES ( 'Business' ), ( 'Economy' ) ) AS fare_conditions ( fare_condition )
demo=# CROSS JOIN
demo=# -- список номеров рядов кресел
demo=# generate_series(1, 30) AS seat_rows ( seat_row )
demo=# CROSS JOIN
demo=# -- список номеров (позиций) кресел в ряду
demo=# ( VALUES ( 'A' ), ( 'B' ), ( 'C' ), ( 'D' ), ( 'E' ), ( 'F' ), ( 'G' ), ( 'H' ), ( 'I' ) ) AS letters ( letter )
demo=# WHERE
demo=# CASE WHEN fare_condition = 'Business'
demo(#   THEN seat_row::integer <= max_seat_row_business AND ascii(letter) <= ascii(max_letter) - 2 -- 2 места меньше в ряду для бизнес-класса
demo=#   WHEN fare_condition = 'Economy'
demo(#   THEN seat_row::integer > max_seat_row_business AND seat_row::integer <= max_seat_row_economy AND letter <= max_letter
demo=# END
demo=# RETURNING *;

```

Рисунок 8 – Задание 10

3 Заключение

В ходе практической работы были изучены основы изменения данных, были выполнены указанные в файле задания.