

DB DB_lab2 Version control

Database Explorer

postgres@localhost

postgres

zhanel

Server Objects

console

1

--1

2

✓

SELECT UPPER(airline_name) AS airline_name_upper

3

FROM Airlines;

4

5

6

7

postgres.public

Services

Database

postgres@localhost

console 1 s 9 ms

Output

airline_name_upper:text

airline_name_upper

1

AIR FRANCE TEST

2

TAP PORTUGAL DEMO

3

LOT POLISH LINES

4

KAZAIR NATIONAL

5

QAZAQ SKY

6

BRITISH AIRWAYS MOCK

7

DELTA TEST

8

ITA AIRWAYS

10 rows

Database Consoles > postgres@localhost > console7:1 LF UTF-8 4 spaces Trial

DB DB_lab2 Version control

Database Explorer

postgres@localhost

postgres

zhanel

Server Objects

console

--1

SELECT UPPER(airline_name) AS airline_name_upper

FROM Airlines;

--2

SELECT REPLACE(airline_name, 'Air', 'Aero') AS new_name

FROM Airlines;

postgres.public

Services

Database

postgres@localhost

console 366 ms

Output --2

new_name

1Aero France Test

2TAP Portugal Demo

3LOT Polish Lines

4KazAero National

5Qazaq Sky

6British Aeroways Mock

7Delta Test

8ITA Aeroways

10 rows

Database Consoles > postgres@localhost > console9:15 LF UTF-8 4 spaces Trial

DB DB_lab2 Version control

Database Explorer

postgres@localhost

postgres

zhanel

Server Objects

console

4

5

6

7

8

9

10

11

12

13

14

15

16

17

--2

SELECT REPLACE(airline_name, 'Air', 'Aero') AS new_name

FROM Airlines;

--3

SELECT flight_number

FROM Flights

WHERE airline1_id IS NOT NULL

AND airline2_id IS NOT NULL;

postgres.public

Services

Tx

Database

postgres@localhost

console 342 ms

Output

--3

===== Задание 3: най...ne1, и airline2 =====

flight_number

1 QS-DF500

2 AF-QS700

2 rows

Database Consoles > postgres@localhost > console17:31 LF UTF-8 4 spaces Trial

DB DB_lab2 Version control

Database Explorer

postgres@localhost

postgres

zhanel

Server Objects

console

postgres.public

SELECT flight_number
FROM Flights
WHERE airline1_id IS NOT NULL
AND airline2_id IS NOT NULL;

SELECT airport_name
FROM Airport
WHERE airport_name LIKE '%Reginal%'
AND airport_name LIKE '%Air%';

Services

Database

postgres@localhost

console 338 ms

Output

airport_name

1Reginal Air Hub

2Big Reginal Airfield

2 rows

Database Consoles > postgres@localhost > console22:1 LF UTF-8 4 spaces Trial

DB DB_lab2 ▾Version control ▾

Database Explorer

postgres@localhost 2

postgres 3

zhanel 3

Server Objects

console ×

⏏⌚Ⓢ⚙️🖨Tx: Auto ▾Playground ▾postgres.public ▾

22SELECT airport_name

23FROM Airport

24WHERE airport_name LIKE '%Reginal%'

25AND airport_name LIKE '%Air%';

26

27

28

29--5

30✓SELECT passenger_name,

31TO_CHAR(birth_date, 'Month DD, YYYY') AS formatted_birthdate

32FROM Passengers;

33

34

35

Services

Tx⊕👁📄⏏⏏⏏

Database

postgres@localhost

console 336 ms

Output--4 2Result 17 ×

📄📊🔄⌚Ⓢ🔍🔍🔍

CSV ▾⏏⏏⏏👁⚙️

	passenger_name 🔍	formatted_birthdate 🔍
1	Aruzhan Sadykova	May 12, 2004
2	Baurzhan Tleubay	November 30, 1998
3	Dana Ibrayeva	February 14, 1985
4	Miras Kazybek	September 07, 1970
5	Zarina Akhmet	January 25, 2000

5 rows ▾⋮

Database Consoles > postgres@localhost > console29:4LFUTF-84 spaces🔗🔔Trial

DB DB_lab2 Version control

Database Explorer

postgres@localhost

postgres

zhanel

Server Objects

console

postgres.public

2

--5

SELECT passenger_name,

TO_CHAR(birth_date, 'Month DD, YYYY') AS formatted_birthdate

FROM Passengers;

--6

SELECT flight_number

FROM Flights

WHERE actual_arrival > scheduled_arrival;

3 rows

Services

Database

postgres@localhost

console 343 ms

Output

--6

Result 17

flight_number

1 AF101

2 DX300

3 AF700

Database Consoles > postgres@localhost > console 39:42 LF UTF-8 4 spaces Trial

Database Explorer

postgres@localhost

postgres

zhanel

Server Objects

console

Playground

postgres.public

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

--6

SELECT flight_number

FROM Flights

WHERE actual_arrival > scheduled_arrival;

--7

SELECT passenger_name,

CASE

WHEN EXTRACT(YEAR FROM AGE(CURRENT_DATE, birth_date)) BETWEEN 18 AND 35 THEN 'Young'

WHEN EXTRACT(YEAR FROM AGE(CURRENT_DATE, birth_date)) BETWEEN 36 AND 55 THEN 'Adult'

ELSE 'Other'

END AS age_group

FROM Passengers;

Services

Database

postgres@localhost

console 342 ms

Output

--6

--7

CSV

	passenger_name	age_group
1	Aruzhan Sadykova	Young
2	Baurzhan Tleubay	Young
3	Dana Ibrayeva	Adult
4	Miras Kazybek	Adult
5	Zarina Akhmet	Young

5 rows

DB DB_lab2 ▾

Version control ▾

Database Explorer

postgres@localhost 2

postgres 3

zhanel 3

Server Objects

console ×

▶ ⌚ Ⓟ ⚙ 🖨 Tx: Auto ▾ Playground ▾

postgres.public ▾

44 SELECT passenger_name,

45 CASE

46 WHEN EXTRACT(YEAR FROM AGE(CURRENT_DATE, birth_date)) BETWEEN 18 AND 35 THEN 'Young'

47 WHEN EXTRACT(YEAR FROM AGE(CURRENT_DATE, birth_date)) BETWEEN 36 AND 55 THEN 'Adult'

48 ELSE 'Other'

49 END AS age_group

50 FROM Passengers;

51

52

53

54 --8

55 ✓ SELECT ticket_id,

56 CASE

57 WHEN price < 100 THEN 'Cheap'

58 WHEN price BETWEEN 100 AND 300 THEN 'Medium'

59 ELSE 'Expensive'

60 END AS price_category

61 FROM Tickets;

62

! 2 ^ ▾

Services

Tx + 👁 📄 ↕ ×

Database

postgres@localhost

console 340 ms

Output --8 × --7

🗃 📈 ⌛ ⌚ ⏹ 🔍 🔍 🔍 🔍

CSV ▾ ⬇ ⬆ ↶ 👁 ⚙

	ticket_id 🔍	price_category 🔍
1	1	Medium
2	2	Medium
3	3	Medium
4	4	Medium
5	5	Medium
6	6	Medium
7	7	Medium
8	8	Medium

10 rows ▾ ⋮

Database Consoles > postgres@localhost > console

61:14 LF UTF-8 4 spaces 🔒 🔔 Trial

DB DB_lab2 Version control

Database Explorer

postgres@localhost

postgres

zhanel

Server Objects

console

59

60

61

62

63

64

65

66

67

68

69

70

71

72

ELSE 'Expensive'

END AS price_category

FROM Tickets;

--9

SELECT country, COUNT(*) AS airlines_count

FROM Airlines

GROUP BY country

ORDER BY country;

2

Services

Database

postgres@localhost

console 352 ms

Output

--8

--9

country

airlines_count

1	France	2
2	Kazakhstan	2
3	Poland	1
4	USA	3

4 rows

Database Consoles > postgres@localhost > console71:1 L F UTF-8 4 spaces Trial

DB DB_lab2 Version control

Database Explorer

postgres@localhost

postgres

zhanel

Server Objects

console

FROM Tickets;

--9

SELECT country, COUNT(*) AS airlines_count

FROM Airlines

GROUP BY country

ORDER BY country;

--10

SELECT flight_number,

scheduled_arrival,

actual_arrival

FROM Flights

WHERE actual_arrival > scheduled_arrival;

2

Services

Database

postgres@localhost

console 347 ms

Output

--10

--9

flight_number

scheduled_arrival

actual_arrival

1	AF101	2025-10-06 08:00:00.000000	2025-10-06 08:10:00.000000
2	DX300	2025-10-06 11:00:00.000000	2025-10-06 11:45:00.000000
3	AF700	2025-10-06 13:50:00.000000	2025-10-06 14:30:00.000000

3 rows

Database Consoles > postgres@localhost > console78:42 LF UTF-8 4 spaces Trial