

Laboratory work 6

JOIN operations tasks

1. Write a query that displays all flights of a specific airline.

The screenshot shows the PostgreSQL IDE interface. The left sidebar displays the database schema, including tables like 'airline' and 'flights'. The main query editor contains the following SQL query:

```
SELECT f.flight_no,  
       f.scheduled_departure,  
       f.scheduled_arrival,  
       a.airline_name  
FROM flights f  
JOIN airline a ON f.airline_id = a.airline_id  
WHERE a.airline_name = 'IPC';
```

The 'Data Output' pane at the bottom shows the results of the query, displaying 12 rows of flight data for the airline 'IPC'. The columns are: flight_no, scheduled_departure, scheduled_arrival, and airline_name.

flight_no	scheduled_departure	scheduled_arrival	airline_name
BR-PE	2024-01-16	2023-06-02	IPC
MZ-G	2023-09-21	2023-11-29	IPC
AU-NT	2023-03-29	2023-05-06	IPC
FR-K	2023-12-26	2023-04-30	IPC
US-VT	2023-08-28	2023-08-02	IPC
PH-BUK	2023-08-20	2023-10-24	IPC
SD-01	2023-12-02	2023-12-17	IPC
PA-8	2023-03-19	2023-08-21	IPC
NA-KU	2023-06-26	2023-12-04	IPC
US-CT	2023-11-07	2023-05-03	IPC
RU-CU	2023-04-16	2023-09-15	IPC
BE-VAN	2023-05-03	2023-06-01	IPC

2. Compose a query to obtain a list of all flights with the names of departure airports.

The screenshot shows the PostgreSQL IDE interface. The left sidebar displays the database schema, including tables like 'flights' and 'airport'. The main query editor contains the following SQL query:

```
SELECT f.flight_no,  
       dep.airport_name AS departure_airport,  
       f.scheduled_departure  
FROM flights f  
JOIN airport dep ON f.departure_airport_id = dep.airport_id;
```

The 'Data Output' pane at the bottom shows the results of the query, displaying 12 rows of flight data with the departure airport name. The columns are: flightNo, departure_airport, and scheduled_departure.

flightNo	departure_airport	scheduled_departure
US-CT	Eiorza Airport	2024-01-22
US-NM	Figari Sud-Corse Airport	2023-07-21
FI-OL	Darchula Airport	2023-03-29
RU-KR	Lime Acres Finsch Mine Airport	2024-01-02
RO-DJ	Hana Airport	2023-07-03
CA-SK	Darchula Airport	2023-07-07
AU-TAS	Ocean Falls Seaplane Base	2023-10-12
US-AZ	Figari Sud-Corse Airport	2023-07-29
IN-OR	Hana Airport	2023-05-18
AU-NT	Longana Airport	2023-11-25
TH-57	Delta County Airport	2023-03-28
CA-NL	Eiorza Airport	2023-03-22

A green status bar at the bottom indicates: "Successfully run. Total query runtime: 44 msec. 1000 rows affected."

3. Create a query that finds all airlines that have no flights scheduled for the next month.

The screenshot shows the PostgreSQL query editor interface. The query is as follows:

```
1 SELECT a.airline_name
2 FROM airline a
3 LEFT JOIN flights f
4   ON a.airline_id = f.airline_id
5   AND f.scheduled_departure BETWEEN CURRENT_DATE AND (CURRENT_DATE + INTERVAL '1 mo')
6 WHERE f.flight_id IS NULL;
```

The Data Output pane shows the results of the query, displaying a list of airline names:

airline_name
IPC
PDN
KLE
KHS
YLQ
NGL
0
QIG
NQX
SOZ
IVA
KOQ

Total rows: 50 Query complete 00:00:00.057

4. Create a query to display a list of passengers on a specific flight.

The screenshot shows the PostgreSQL query editor interface. The query is as follows:

```
1 SELECT p.first_name || ' ' || p.last_name AS full_name,
2        f.flight_no
3 FROM passengers p
4 JOIN booking b ON p.passenger_id = b.passenger_id
5 JOIN booking_flight bf ON b.booking_id = bf.booking_id
6 JOIN flights f ON bf.flight_id = f.flight_id
7 WHERE f.flight_no = 'US-AR';
```

The Data Output pane shows the results of the query, displaying a list of passengers and their flight numbers:

full_name	flight_no
Tildy Shackelford	US-AR
Tom Gidney	US-AR
Humphrey Mc...	US-AR
Franny Orry	US-AR
Margarethe Yule	US-AR
Reinald Pococke	US-AR

Total rows: 6 Query complete 00:00:00.053

5. Write a query that calculates the average, total, maximum and minimum price of tickets for each flight.

The screenshot shows a PostgreSQL query editor with the following SQL query:

```

SELECT f.flight_no,
       ROUND(AVG(b.price), 2) AS avg_price,
       SUM(b.price) AS total_price,
       MAX(b.price) AS max_price,
       MIN(b.price) AS min_price
FROM flights f
JOIN booking_flight bf ON f.flight_id = bf.flight_id
JOIN booking b ON bf.booking_id = b.booking_id
GROUP BY f.flight_no;

```

The query results are displayed in a table with 6 columns: flight_no, avg_price, total_price, max_price, and min_price. The results show 12 rows of flight data.

flight_no	avg_price	total_price	max_price	min_price
FR-O	826.19	826.19	826.19	826.19
ZA-NL	3719.57	7439.13	5413.24	2025.89
AF-FRA	7638.55	7638.55	7638.55	7638.55
CH-NE	3598.91	7197.81	5140.50	2057.31
HR-18	1869.46	1869.46	1869.46	1869.46
US-NV	5642.37	5642.37	5642.37	5642.37
CA-AB	4395.23	13185.70	5203.21	2779.28
BR-PA	1863.01	1863.01	1863.01	1863.01
UA-32	2442.37	4884.73	3194.55	1690.18
RU-SAK	1590.31	6361.22	2592.30	141.15
AU-WA	4512.75	45127.51	8036.28	826.19
VU-SEE	2771.28	2771.28	2771.28	2771.28

Successfully run. Total query runtime: 57 msec. 384 rows affected.

- Create a query that shows all flights flying to a specific country by combining flights, airports and airline, and using the condition on the country name.

The screenshot shows a PostgreSQL query editor with the following SQL query:

```

SELECT f.flight_no,
       a.airline_name,
       arr.airport_name AS arrival_airport,
       arr.country AS destination_country
FROM flights f
JOIN airline a ON f.airline_id = a.airline_id
JOIN airport arr ON f.arrival_airport_id = arr.airport_id
WHERE arr.country = 'Russia';

```

The query results are displayed in a table with 5 columns: flight_no, airline_name, arrival_airport, and destination_country. The results show 12 rows of flight data arriving in Russia.

flight_no	airline_name	arrival_airport	destination_country
CA-SK	SMM	Akunaq Heliport	Russia
US-MS	HYL	Akunaq Heliport	Russia
BR-PR	SOZ	Akunaq Heliport	Russia
US-AR	NHT	Akunaq Heliport	Russia
US-AK	JKR	Akunaq Heliport	Russia
US-FL	RUG	Akunaq Heliport	Russia
TD-OD	BLD	Akunaq Heliport	Russia
KZ-SEV	NQX	Akunaq Heliport	Russia
FI-LL	SJS	Akunaq Heliport	Russia
CN-32	YHB	Akunaq Heliport	Russia
US-OK	KKG	Akunaq Heliport	Russia
AU-NSW	DNV	Akunaq Heliport	Russia

Successfully run. Total query runtime: 46 msec. 56 rows affected.

- Display a list of minor passengers and their arrival destination.

The screenshot shows a PostgreSQL database interface with a query window. The query is as follows:

```

1 SELECT p.first_name || ' ' || p.last_name AS full_name,
2       arr.country AS arrival_country
3 FROM passengers p
4 JOIN booking b ON p.passenger_id = b.passenger_id
5 JOIN booking_flight bf ON b.booking_id = bf.booking_id
6 JOIN flights f ON bf.flight_id = f.flight_id
7 JOIN airport arr ON f.arrival_airport_id = arr.airport_id
8 WHERE EXTRACT(YEAR FROM AGE(CURRENT_DATE, p.date_of_birth)) < 18;

```

The Data Output tab shows the following results:

full_name	arrival_country
text	character varying (50)

Total rows: 0 Query complete 00:00:00.058 LF Ln 8, Col 66

- Display the passenger's full name, passport number, and the passenger's current time of arrival at the destination.

The screenshot shows a PostgreSQL database interface with a query window. The query is as follows:

```

1 SELECT p.first_name || ' ' || p.last_name AS full_name,
2       p.passport_number,
3       f.actual_arrival
4 FROM passengers p
5 JOIN booking b ON p.passenger_id = b.passenger_id
6 JOIN booking_flight bf ON b.booking_id = bf.booking_id
7 JOIN flights f ON bf.flight_id = f.flight_id;

```

The Data Output tab shows the following results:

full_name	passport_number	actual_arrival
text	character varying (50)	date
1 Muhammad Fasa	109932770-9	2023-07-18
2 Trevar Broun	788025864-7	2024-02-11
3 Auria Breffit	570537341-4	2023-07-11
4 Archie Toffel	677556708-1	2023-06-17
5 Sanders Biddles	514546405-3	2023-09-05
6 Sanders Biddles	514546405-3	2024-03-01
7 Remington Piggot	470074456-1	2023-05-31
8 Glynis Marle	209933120-0	2024-02-02
9 Cesaro McGennis	364900190-X	2023-04-03
10 Con Borrel	837859465-3	2024-02-15
11 Lorianne Robbie	435354132-3	2024-01-19
12 Brad Apperley	842288763-0	2024-02-10

Total rows: 1000 Query complete 00:00:00.049 LF Ln 7, Col 46

Showing rows: 1 to 1000 Page No: 1 of 1

Successfully run. Total query runtime: 49 msec. 1000 rows affected.

- Print a list of flights where the airline's home country and origin country are the same. Group them by the airport country.

database_subj/postgres@PostgreSQL 17* X public.flights/data... X public.airport/data...

database_subj/postgres@PostgreSQL 17

Query Query History

```
1 SELECT f.flight_no,
2       dep.country AS origin_country,
3       a.airline_country AS airline_country
4 FROM flights f
5 JOIN airline a ON f.airline_id = a.airline_id
6 JOIN airport dep ON f.departure_airport_id = dep.airport_id
7 WHERE a.airline_country = dep.country
8 GROUP BY f.flight_no, dep.country, a.airline_country;
```

Scratch Pad X

Data Output Messages Notifications

Showing rows: 1 to 69 Page No: 1 of 1

flight_no	origin_country	airline_country
character varying (50)	character varying (50)	character varying (50)
1 AU-NSW	China	China
2 AU-NT	China	China
3 AU-NT	Greece	Greece
4 AU-QLD	Poland	Poland
5 AU-WA	China	China
6 BR-AM	China	China
7 BR-BA	China	China
8 BR-MG	China	China
9 BR-PE	China	China
10 BR-PR	Philippines	Philippines
11 BS-NP	China	China
12 CA-BC	China	China

Successfully run. Total query runtime: 49 msec. 69 rows affected.

Total rows: 69 Query complete 00:00:00.049 LF Ln 8, Col 54