

# Laboratory work 6

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

The screenshot shows the pgAdmin 4 interface. The left sidebar displays the database structure, including the 'flights' table. The main pane shows a SQL query:

```
1 SELECT f.flight_id, f.sch_departure_time, f.sch_arrival_time,
2       f.departing_airport_id, f.arriving_airport_id,
3       f.act_departure_time, f.act_arrival_time
4 FROM flights f
5 JOIN airline a ON f.airline_id = a.airline_id
6 WHERE airline_name = 'FlyHigh';
```

The 'Data Output' tab at the bottom shows the column headers for the query result:

flight_id	sch_departure_time	sch_arrival_time	departing_airport_id	arriving_airport_id	act_departure_time	act_arrival_time
-----------	--------------------	------------------	----------------------	---------------------	--------------------	------------------

The status bar indicates 'Total rows: 0' and 'Query complete 00:00:00.082'.

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

The screenshot shows the pgAdmin 4 interface. The left sidebar displays the database structure, including the 'flights' table. The main pane shows a SQL query:

```
1 SELECT f.flight_id, f.scheduled_departure,
2       f.departure_airport_id, a.airport_name as departure_airport_name,
3       f.actual_departure, f.actual_arrival
4 FROM flights f
5 JOIN airport a ON f.departure_airport_id = a.airport_id
```

The 'Data Output' tab at the bottom shows the query result table:

flight_id	scheduled_departure	departure_airport_id	departure_airport_name	actual_departure	actual_arrival
1	2024-01-22	12	Elorza Airport	2023-10-30	2023-11-07
2	2023-07-21	13	Figari Sud-Corse Airport	2024-02-09	2024-01-23
3	2023-03-29	18	Darchula Airport	2024-02-21	2023-05-14
4	2024-01-02	3	Lime Acres Finsch Mine Airport	2023-10-10	2023-04-07
5	2023-07-03	6	Hana Airport	2023-11-18	2023-07-02
6	2023-07-07	18	Darchula Airport	2024-02-19	2023-04-28
7	2023-10-12	15	Ocean Falls Seaplane Base	2023-12-04	2023-07-12
8	2023-07-29	13	Figari Sud-Corse Airport	2023-07-10	2023-08-01

The status bar indicates 'Total rows: 1000' and 'Query complete 00:00:00.119'.

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

The screenshot shows the pgAdmin 4 interface. On the left, the 'Object Explorer' pane is expanded to show the 'flights' table under the 'public' schema. The 'Columns' list for 'flights' is visible, including 'flight\_id', 'flight\_no', 'scheduled\_departure', 'scheduled\_arrival', 'departure\_airport\_id', 'arrival\_airport\_id', 'departing\_gate', 'arriving\_gate', 'airline\_id', 'status', 'actual\_departure', 'actual\_arrival', 'created\_at', and 'update\_at'. The 'Query' editor in the center contains the following SQL query:

```
1 SELECT a.airline_id, 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 >= date_trunc('month', now()) + interval '1 month'
6 AND f.scheduled_departure < date_trunc('month', now()) + interval '2 month'
7 WHERE f.flight_id IS NULL;
```

The 'Data Output' pane at the bottom shows the results of the query, displaying a table with two columns: 'airline\_id' and 'airline\_name'. The results are as follows:

airline_id	airline_name
20	SSL
25	CLY
26	YHB
27	KKL
11	IVA
39	GYA
17	YLP

The status bar at the bottom indicates 'Total rows: 50' and 'Query complete 00:00:00.143'.

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

The screenshot shows the pgAdmin 4 interface. On the left, the 'Object Explorer' pane is expanded to show the 'booking\_flight' table under the 'public' schema. The 'Columns' list for 'booking\_flight' is visible, including 'booking\_flight\_id', 'booking\_id', 'flight\_id', 'created\_at', and 'update\_at'. The 'Query' editor in the center contains the following SQL query:

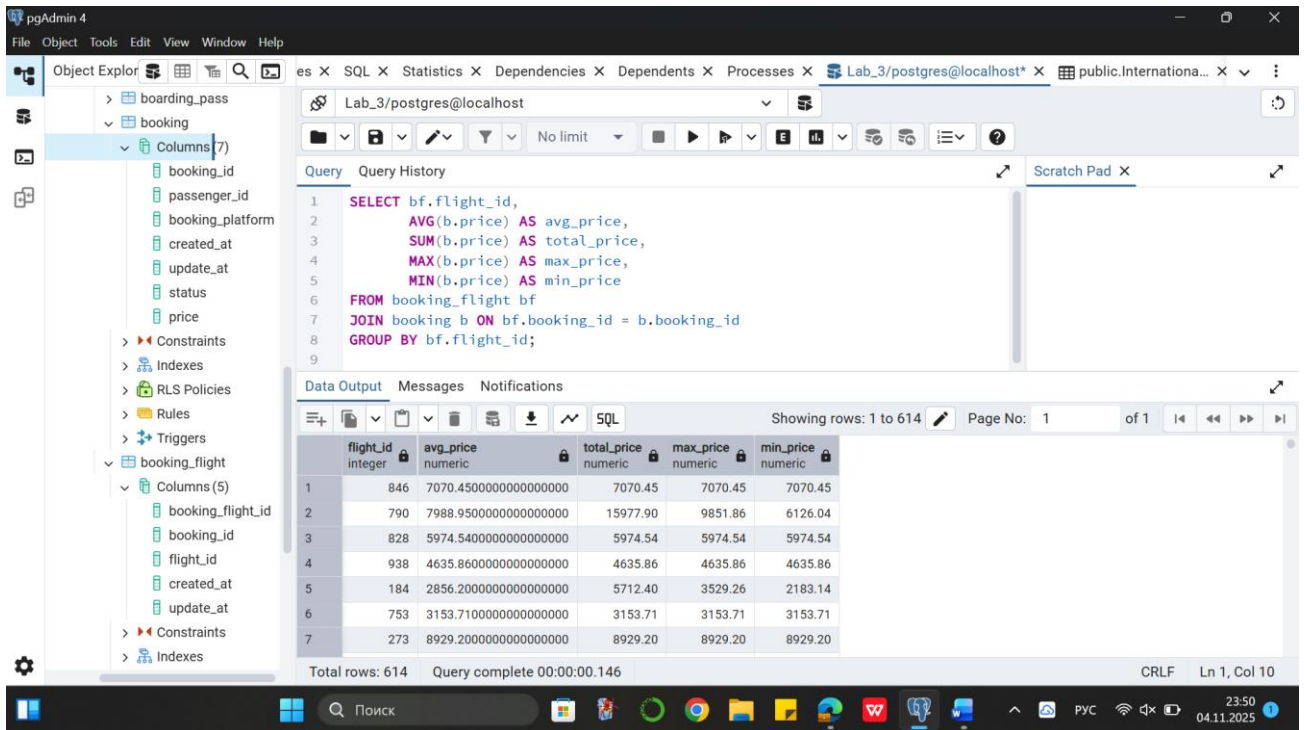
```
1 SELECT p.*
2 FROM passengers p
3 JOIN booking b ON p.passenger_id = b.passenger_id
4 JOIN booking_flight bf ON b.booking_id = bf.booking_id
5 WHERE bf.flight_id = 15;
```

The 'Data Output' pane at the bottom shows the results of the query, displaying a table with seven columns: 'passenger\_id', 'first\_name', 'last\_name', 'date\_of\_birth', 'gender', 'country\_of\_citizenship', and 'country\_of\_residence'. The results are as follows:

passenger_id	first_name	last_name	date_of_birth	gender	country_of_citizenship	country_of_residence
163	Maryellen	MacGilmartin	1975-02-25	Female	Finland	China

The status bar at the bottom indicates 'Total rows: 1' and 'Query complete 00:00:00.066'.

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



The screenshot shows the pgAdmin 4 interface. On the left, the 'Object Explorer' pane displays the database structure, including tables like 'booking' and 'booking\_flight'. The main query editor contains the following SQL query:

```

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

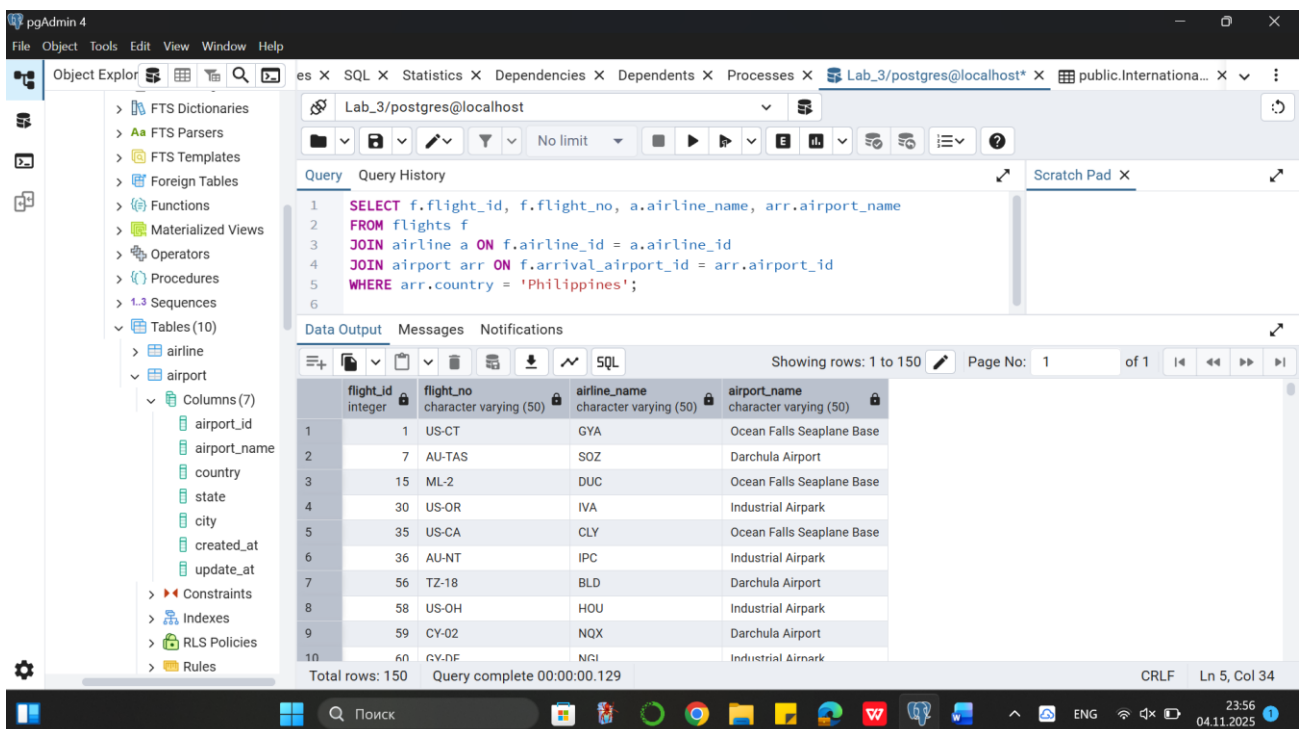
```

The 'Data Output' pane shows the results of the query, displaying columns: flight\_id, avg\_price, total\_price, max\_price, and min\_price. The results are as follows:

flight_id	avg_price	total_price	max_price	min_price
846	7070.45000000000000000000	7070.45	7070.45	7070.45
790	7988.95000000000000000000	15977.90	9851.86	6126.04
828	5974.54000000000000000000	5974.54	5974.54	5974.54
938	4635.86000000000000000000	4635.86	4635.86	4635.86
184	2856.20000000000000000000	5712.40	3529.26	2183.14
753	3153.71000000000000000000	3153.71	3153.71	3153.71
273	8929.20000000000000000000	8929.20	8929.20	8929.20

Total rows: 614. Query complete 00:00:00.146.

6. 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 the pgAdmin 4 interface. On the left, the 'Object Explorer' pane displays the database structure, including tables like 'airline' and 'airport'. The main query editor contains the following SQL query:

```

SELECT f.flight_id, f.flight_no, a.airline_name, arr.airport_name
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 = 'Philippines';

```

The 'Data Output' pane shows the results of the query, displaying columns: flight\_id, flight\_no, airline\_name, and airport\_name. The results are as follows:

flight_id	flight_no	airline_name	airport_name
1	US-CT	GYA	Ocean Falls Seaplane Base
2	AU-TAS	SOZ	Darchula Airport
3	ML-2	DUC	Ocean Falls Seaplane Base
4	US-OR	IVA	Industrial Airpark
5	US-CA	CLY	Ocean Falls Seaplane Base
6	AU-NT	IPC	Industrial Airpark
7	TZ-18	BLD	Darchula Airport
8	US-OH	HOU	Industrial Airpark
9	CY-02	NQX	Darchula Airport
10	CV-DE	NGI	Industrial Airpark

Total rows: 150. Query complete 00:00:00.129.

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

The screenshot shows the pgAdmin 4 interface. On the left, the 'Object Explorer' pane is open, showing the 'passengers' table under the 'public' schema. The 'Columns' tab is selected, showing 10 columns: passenger\_id, first\_name, last\_name, date\_of\_birth, gender, country\_of\_citizenship, country\_of\_residence, passport\_number, created\_at, and update\_at. The 'Query' pane contains the following SQL query:

```
1 SELECT p.passenger_id,
2       p.first_name || ' ' || p.last_name AS full_name,
3       arr.airport_name AS arrival_airport
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
8 JOIN airport arr ON f.arrival_airport_id = arr.airport_id
9 WHERE EXTRACT(YEAR FROM AGE(now(), p.date_of_birth)) < 18;
10
```

The 'Data Output' pane shows the results of the query, displaying 37 rows. The columns are passenger\_id, full\_name, and arrival\_airport. The first 6 rows are shown:

passenger_id	full_name	arrival_airport
159	Vivyan Mallabone	Alert Bay Airport
70	Lester Blades	Armidale Airport
70	Lester Blades	Figari Sud-Corse Airport
41	Cleve Edgeler	Industrial Airpark
80	Bradley Grolle	Armidale Airport
41	Cleve Edgeler	Fort Worth Alliance Airport

The status bar at the bottom indicates 'Total rows: 37' and 'Query complete 00:00:00.065'.

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

The screenshot shows the pgAdmin 4 interface. On the left, the 'Object Explorer' pane is open, showing the 'flights' table under the 'public' schema. The 'Columns' tab is selected, showing 14 columns: flight\_id, flight\_no, scheduled\_departure, scheduled\_arrival, departure\_airport\_id, arrival\_airport\_id, departing\_gate, arriving\_gate, airline\_id, status, actual\_departure, actual\_arrival, created\_at, and update\_at. The 'Query' pane contains the following SQL query:

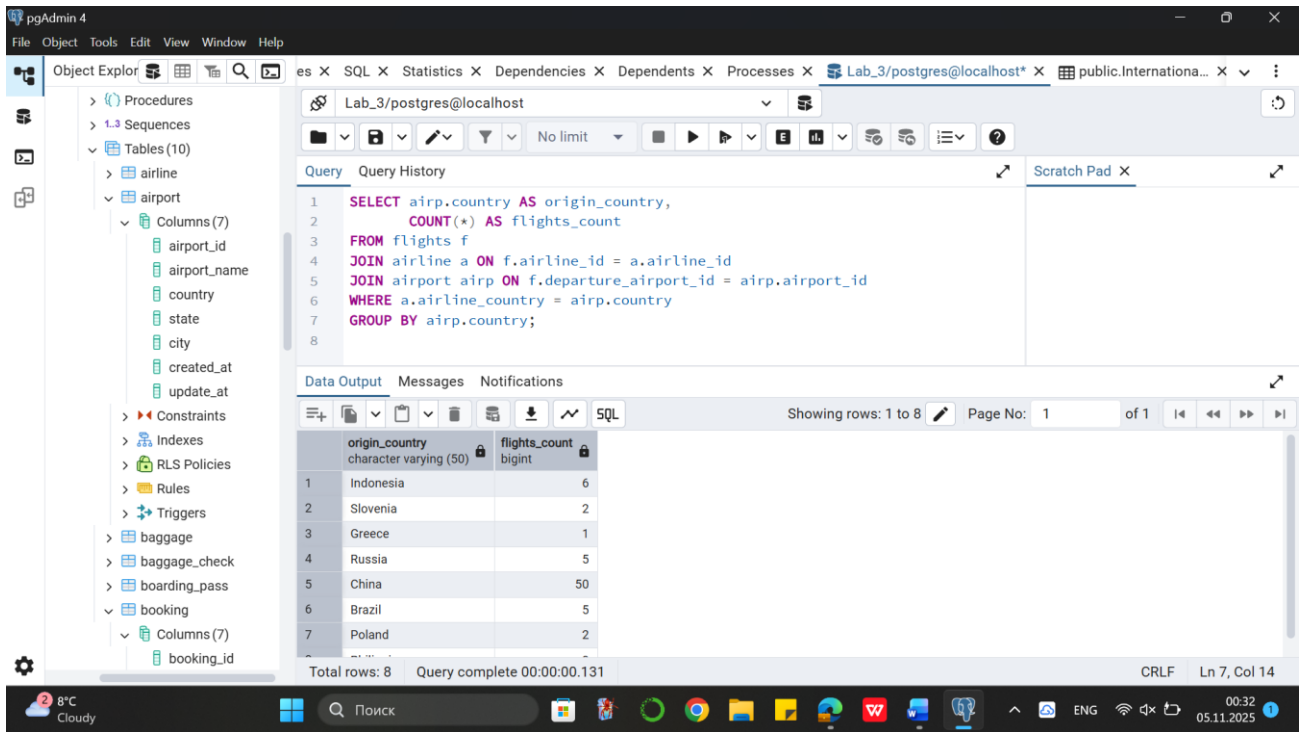
```
1 SELECT p.passport_number,
2       p.first_name || ' ' || p.last_name AS full_name,
3       f.actual_arrival
4 FROM passengers p
5 JOIN booking b ON p.passenger_id = b.passenger_id
6 JOIN booking_flight bf ON b.booking_id = bf.booking_id
7 JOIN flights f ON bf.flight_id = f.flight_id
8
```

The 'Data Output' pane shows the results of the query, displaying 1000 rows. The columns are passport\_number, full\_name, and actual\_arrival. The first 7 rows are shown:

passport_number	full_name	actual_arrival
109932770-9	Muhammad Fass	2023-07-18
788025864-7	Trevar Broun	2024-02-11
570537341-4	Auria Breffit	2023-07-11
677556708-1	Archie Toffel	2023-06-17
514546405-3	Sanders Biddles	2023-09-05
514546405-3	Sanders Biddles	2024-03-01
470074456-1	Remington Piggot	2023-05-31

The status bar at the bottom indicates 'Total rows: 1000' and 'Query complete 00:00:00.130'.

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



The screenshot shows the pgAdmin 4 interface. On the left, the Object Explorer displays the database structure, including the 'airport' table. The central pane shows a SQL query that selects the origin country and the count of flights, grouped by airport country. The bottom pane displays the results of the query in a table format.

```
SELECT airp.country AS origin_country,
       COUNT(*) AS flights_count
FROM flights f
JOIN airline a ON f.airline_id = a.airline_id
JOIN airport airp ON f.departure_airport_id = airp.airport_id
WHERE a.airline_country = airp.country
GROUP BY airp.country;
```

	origin_country character varying (50)	flights_count bigint
1	Indonesia	6
2	Slovenia	2
3	Greece	1
4	Russia	5
5	China	50
6	Brazil	5
7	Poland	2

Total rows: 8    Query complete 00:00:00.131    CRLF    Ln 7, Col 14