

JOIN operations tasks

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

The screenshot shows the pgAdmin 4 interface. The title bar says "pgAdmin 4" and "Welcome airport_db/postgres@PostgreSQL 17+ X". The main area has a "Query" tab selected, displaying the following SQL code:

```
1 v SELECT f.flight_id, a.airline_name
2   FROM flights f
3   INNER JOIN airline a
4     ON f.airline_id = a.airline_id;
```

Below the code is a "Data Output" tab showing a table with two columns: "flight_id" and "airline_name". The data consists of 98 rows, each containing a flight ID and its corresponding airline name. A green status bar at the bottom right indicates "Successfully run. Total query runtime: 96 msec. 98 rows affected."

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 title bar says "pgAdmin 4" and "Welcome airport_db/postgres@PostgreSQL 17+ X". The main area has a "Query" tab selected, displaying the following SQL code:

```
1 v SELECT f.flight_id, a.airport_name
2   FROM flights f
3   INNER JOIN airport a
4     ON f.departing_airport_id = a.airport_id;
```

Below the code is a "Data Output" tab showing a table with two columns: "flight_id" and "airport_name". The data consists of 98 rows, each containing a flight ID and its corresponding departure airport name. A green status bar at the bottom right indicates "Successfully run. Total query runtime: 85 msec. 98 rows affected."

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

pgAdmin 4

Welcome airport_db/postgres@PostgreSQL 17*

```

1 SELECT a.airline_name
2 FROM airline a
3 LEFT JOIN flights f
4 ON a.airline_id = f.airline_id
5 AND DATE_TRUNC('month', f.sch_departure_time) = DATE_TRUNC('month', CURRENT_DATE + INTERVAL '1 month')
6 WHERE f.flight_id IS NULL;

```

Data Output Messages Notifications

Showing rows: 1 to 200 Page No: 1 of 1

airline_name
QUATZ
TOPICLOUNGE
FANNOODLE
GEVEE
JANYX
QUAXO
REALFIRE
CHATTERPOINT
RIFFWIRE
ABATA

Total rows: 200 Query complete 00:00:00.227

2 cm of snow Wednesday 22:08 CRLF Ln 2, Col 9 04.11.2025

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

pgAdmin 4

Welcome airport_db/postgres@PostgreSQL 17*

```

1 SELECT
2     p.first_name,
3     p.last_name
4 FROM passengers p
5 JOIN booking b ON p.passenger_id = b.passenger_id
6 WHERE b.flight_id = 100;

```

Data Output Messages Notifications

Showing rows: 1 to 2 Page No: 1 of 1

first_name	last_name
Ebeneser	Klimecki
Aldous	Wild

Total rows: 2 Query complete 00:00:00.117

5°C Cloudy 22:20 CRLF Ln 6, Col 25 04.11.2025

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

pgAdmin 4

Welcome airport_db/postgres@PostgreSQL 17*

Query History

```

1 SELECT
2     f.flight_id,
3     AVG(b.ticket_price) AS avg_price,
4     SUM(b.ticket_price) AS total_price,
5     MAX(b.ticket_price) AS max_price,
6     MIN(b.ticket_price) AS min_price
7 FROM flights f
8 JOIN booking b
9      ON f.flight_id = b.flight_id
10 GROUP BY f.flight_id;
11

```

Data Output Messages Notifications

	flight_id	avg_price	total_price	max_price	min_price
1	116	21169.490000000000	21169.49	21169.49	21169.49
2	54	16624.250000000000	66497.00	19938.55	9984.88
3	71	16646.470000000000	16646.47	16646.47	16646.47
4	68	16089.150000000000	48267.45	20845.30	12022.32
5	138	20663.410000000000	20663.41	20663.41	20663.41
6	70	11437.510000000000	11437.51	11437.51	11437.51
7	146	15662.590000000000	15662.59	15662.59	15662.59
8	67	14420.780000000000	14420.78	14420.78	14420.78
9	63	16798.500000000000	16798.50	16798.50	16798.50
10	174	20453.320000000000	40906.64	22519.86	18386.78

Showing rows: 1 to 48 Page No: 1 of 1

Total rows: 48 Query complete 00:00:140

SUN - EVE Game score 22:43 CRLF Ln 10, Col 1 04.11.2025

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.

pgAdmin 4

Welcome airport_db/postgres@PostgreSQL 17*

Query History

```

1 SELECT
2     f.flight_id,
3     a.airport_name AS arrival_airport,
4     a.country,
5     ai.airline_name
6 FROM flights f
7 JOIN airport a
8      ON f.arriving_airport_id = a.airport_id
9 JOIN airline ai
10     ON f.airline_id = ai.airline_id
11 WHERE a.country = 'Turkey';
12

```

Data Output Messages Notifications

flight_id	arrival_airport	country	airline_name

Showing rows: 1 to 0 Page No: 1 of 1

Total rows: 0 Query complete 00:00:00.078

Feels colder Now 22:56 CRLF Ln 11, Col 28 04.11.2025

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

pgAdmin 4

Welcome airport_db/postgres@PostgreSQL 17*

```

1 SELECT
2     p.first_name,
3     p.last_name,
4     a.airport_name AS arrival_destination
5 FROM passengers p
6 JOIN booking b
7     ON p.passenger_id = b.passenger_id
8 JOIN flights f
9     ON b.flight_id = f.flight_id
10 JOIN airport a
11     ON f.arriving_airport_id = a.airport_id
12 WHERE EXTRACT(YEAR FROM AGE(CURRENT_DATE, p.date_of_birth)) < 18;

```

Data Output Messages Notifications

first_name	last_name	arrival_destination

Total rows: 0 Query complete 00:00:00.074

Successfully run. Total query runtime: 74 msec. 0 rows affected.

23:00 04.11.2025

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

pgAdmin 4

Welcome airport_db/postgres@PostgreSQL 17*

```

1 SELECT
2     p.first_name || ' ' || p.last_name AS full_name,
3     p.passport_number,
4     f.sch_arrival_time AS arrival_time
5 FROM passengers p
6 JOIN booking b
7     ON p.passenger_id = b.passenger_id
8 JOIN flights f
9     ON b.flight_id = f.flight_id;

```

Data Output Messages Notifications

full_name	passport_number	arrival_time
Buron Branscombe	P010222	2025-04-19 00:00:00
Kirbee Elloy	P188647	2023-11-03 00:00:00
Gino Derkes	P827654	2025-03-16 00:00:00
Nikoletta Demcak	P767986	2023-10-04 00:00:00
Tadeas Crichtmore	P150074	2025-04-28 00:00:00
Abramo Pond-Jones	P913700	2025-05-12 00:00:00
Engracia Fernandez	P056899	2025-03-05 00:00:00
Aldwin Laviss	P546764	2025-07-23 00:00:00
Stern Marr	P859822	2023-12-16 00:00:00
Ebenezer Klimecki	P353234	2025-07-25 00:00:00

Total rows: 67 Query complete 00:00:00.093

Showing rows: 1 to 67 / Page No: 1 of 1 14 44 66 77 88 99 100

Successfully run. Total query runtime: 93 msec. 67 rows affected.

23:07 04.11.2025

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. The top menu bar includes File, Object, Tools, Edit, View, Window, Help, and a Welcome message. The title bar indicates the connection is to 'airport_db/postgres@PostgreSQL 17*'. The main area displays a SQL query window with the following code:

```
1 ✓ SELECT
2     f.flight_id,
3     ai.airline_name,
4     a.country AS airport_country
5   FROM flights f
6   JOIN airline ai
7     ON f.airline_id = ai.airline_id
8   JOIN airport a
9     ON f.departing_airport_id = a.airport_id
10 WHERE ai.airline_country = a.country
11 GROUP BY a.country, f.flight_id, ai.airline_name;
```

Below the query window, there are tabs for Data Output, Messages, and Notifications. The Data Output tab is active, showing a table structure with three columns: flight_id, airline_name, and airport_country. The flight_id column is defined as integer, airline_name as character varying(50), and airport_country as character varying(50). A status bar at the bottom shows 'Total rows: 0' and 'Query complete 00:00:00.080'. A green success message in the bottom right corner states 'Successfully run. Total query runtime: 80 msec. 0 rows affected.' The bottom right also shows system icons for temperature (5°C), battery, signal strength, and date/time (04.11.2025).