

Kazakh-British Technical University

DATABASES

Lab 6

Alexandr Kislistsin

23B031861

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

The screenshot shows a database console interface with a query editor and a results pane. The query editor contains the following SQL code:

```
ADD CONSTRAINT chk_ticket_discount
);
SELECT f.flight_id,
       f.sch_departure_time,
       f.sch_arrival_time,
       a.airline_name
FROM flights f
JOIN airline a ON f.airline_id = a.airline_id
WHERE a.airline_name = 'KazAir';
```

The results pane displays a single row of data:

flight_id	sch_departure_time	sch_arrival_time	airline_name
1	16 2025-11-16 10:33:57	2025-11-16 16:33:57	KazAir

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

The screenshot shows a database console interface with a query editor and a results pane. The query editor contains the following SQL code:

```
);
SELECT f.flight_id,
       f.sch_departure_time,
       f.sch_arrival_time,
       a.airline_name
FROM flights f
JOIN airline a ON f.airline_id = a.airline_id
WHERE a.airline_name = 'KazAir';
SELECT f.flight_id,
       f.sch_departure_time,
       a.airport_name AS departure_airport
FROM flights f
JOIN airport a ON f.departing_airport_id = a.airport_id;
```

The results pane displays a list of flights with their departure airports:

flight_id	sch_departure_time	departure_airport
1	2025-07-20 10:27:20	East Patricia Intl Airport
2	2025-06-19 22:56:44	Garciaville International Airport
3	2024-11-23 14:49:10	Montoyachester Intl Airport
4	2025-05-29 12:25:08	Port Beverly Intl Airport
5	2025-09-05 11:24:17	West Carolinestad International Airport
6	2025-09-24 01:45:31	New Christopherside Intl Airport
7	2025-06-17 14:56:28	/iew Intl Airport
8	2025-01-17 17:24:40	an Intl Airport

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

The screenshot shows a database console interface with a query editor and an output window. The query editor contains the following SQL code:

```
415 SELECT f.flight_id,  
416       f.sch_departure_time,  
417       a.airport_name AS departure_airport  
418 FROM flights f  
419 JOIN airport a ON f.departing_airport_id = a.airport_id;  
420  
421  
422 ✓ SELECT a.airline_name  
423 FROM airline a  
424 LEFT JOIN flights f ON a.airline_id = f.airline_id  
425 AND MONTH(f.sch_departure_time) = MONTH(CURDATE()) + 1  
426 WHERE f.flight_id IS NULL;  
427
```

The output window displays the results of the query, showing a list of airlines with no flights scheduled for the next month:

airline_name
4 FlyFly
5 Nelson Ltd
6 Ramirez, Green and Taylor
7 Parker Ltd
8 Bryan-Dixon
9 Anderson-Green
10 Ballard-Hamilton
11 Sloan-Welch

95 rows

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

The screenshot shows a database console interface with a query editor and an output window. The query editor contains the following SQL code:

```
421  
422 SELECT a.airline_name  
423 FROM airline a  
424 LEFT JOIN flights f ON a.airline_id = f.airline_id  
425 AND MONTH(f.sch_departure_time) = MONTH(CURDATE()) + 1  
426 WHERE f.flight_id IS NULL;  
427  
428  
429 ✓ SELECT p.first_name, p.last_name, f.flight_id  
430 FROM passengers p  
431 JOIN booking b ON p.passenger_id = b.passenger_id  
432 JOIN flights f ON b.flight_id = f.flight_id  
433 WHERE f.flight_id = 7;  
434  
435
```

The output window displays the results of the query, showing a list of passengers on flight 7:

first_name	last_name	flight_id
1 Carlos	Allen	7
2 Erik	Ferrell	7

2 rows

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

The screenshot shows a database console interface with a SQL query and its results. The query is as follows:

```
SELECT p.first_name, p.last_name, f.flight_id
FROM passengers p
JOIN booking b ON p.passenger_id = b.passenger_id
JOIN flights f ON b.flight_id = f.flight_id
WHERE f.flight_id = 7;

SELECT flight_id,
       AVG(ticket_price) AS avg_price,
       SUM(ticket_price) AS total_price,
       MAX(ticket_price) AS max_price,
       MIN(ticket_price) AS min_price
FROM booking
GROUP BY flight_id;
```

The results are displayed in a table with 6 columns: flight_id, avg_price, total_price, max_price, min_price, and an additional column. The data is as follows:

flight_id	avg_price	total_price	max_price	min_price	
3	5	1152.990000	1152.99	1152.99	1152.99
4	6	1830.066667	5490.20	2309.21	1544.11
5	7	1560.410000	3136.82	1984.49	1152.33
6	8	1740.360000	1740.36	1740.36	1740.36
7	9	1009.160000	3027.48	1612.05	602.60
8	12	1090.755000	2181.51	1095.99	1085.52
9	15	1188.603333	7565.81	1905.05	495.93
10	17	1048.460000	145.38	1505.02	346.02

6. Create a query that shows all flights flying to a specific country by combining flights, airports and airline.

The screenshot shows a database console interface with a SQL query and its results. The query is as follows:

```
SUM(ticket_price) AS total_price,
MAX(ticket_price) AS max_price,
MIN(ticket_price) AS min_price
FROM booking
GROUP BY flight_id;

SELECT f.flight_id,
       a.airline_name,
       ap.airport_name AS destination_airport,
       ap.country AS destination_country
FROM flights f
JOIN airline a ON f.airline_id = a.airline_id
JOIN airport ap ON f.arriving_airport_id = ap.airport_id
WHERE ap.country = 'China';
```

The results are displayed in a table with 4 columns: flight_id, airline_name, destination_airport, and destination_country. The data is as follows:

flight_id	airline_name	destination_airport	destination_country
1	7 Roberts-Delgado	New Patricktown Intl Airport	China
2	86 Pham-Mann	New Patricktown Intl Airport	China
3	21 Lewis, Gonzalez and Combs	Lake Gregoryburgh Intl Airport	China
4	35 West-Hawkins	Shawnaport Intl Airport	China
5	84 West-Hawkins	Shawnaport Intl Airport	China
6	87 AirEasy	Shawnaport Intl Airport	China

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

The screenshot shows the SQL Developer interface with a query executed in the console. The query selects the first and last names of passengers, the arrival airport name, and the arrival time, filtered by passengers under 18 years old. The result set is displayed in the Output window.

```
451 FROM flights f
452 JOIN airline a ON f.airline_id = a.airline_id
453 JOIN airport ap ON f.arriving_airport_id = ap.airport_id
454 WHERE ap.country = 'China';
455
456
457 SELECT p.first_name, p.last_name, ap.airport_name AS arrival_airport
458 FROM passengers p
459 JOIN booking b ON p.passenger_id = b.passenger_id
460 JOIN flights f ON b.flight_id = f.flight_id
461 JOIN airport ap ON f.arriving_airport_id = ap.airport_id
462 WHERE TIMESTAMPDIFF(YEAR, p.date_of_birth, CURDATE()) < 18;
```

first_name	last_name	arrival_airport
------------	-----------	-----------------

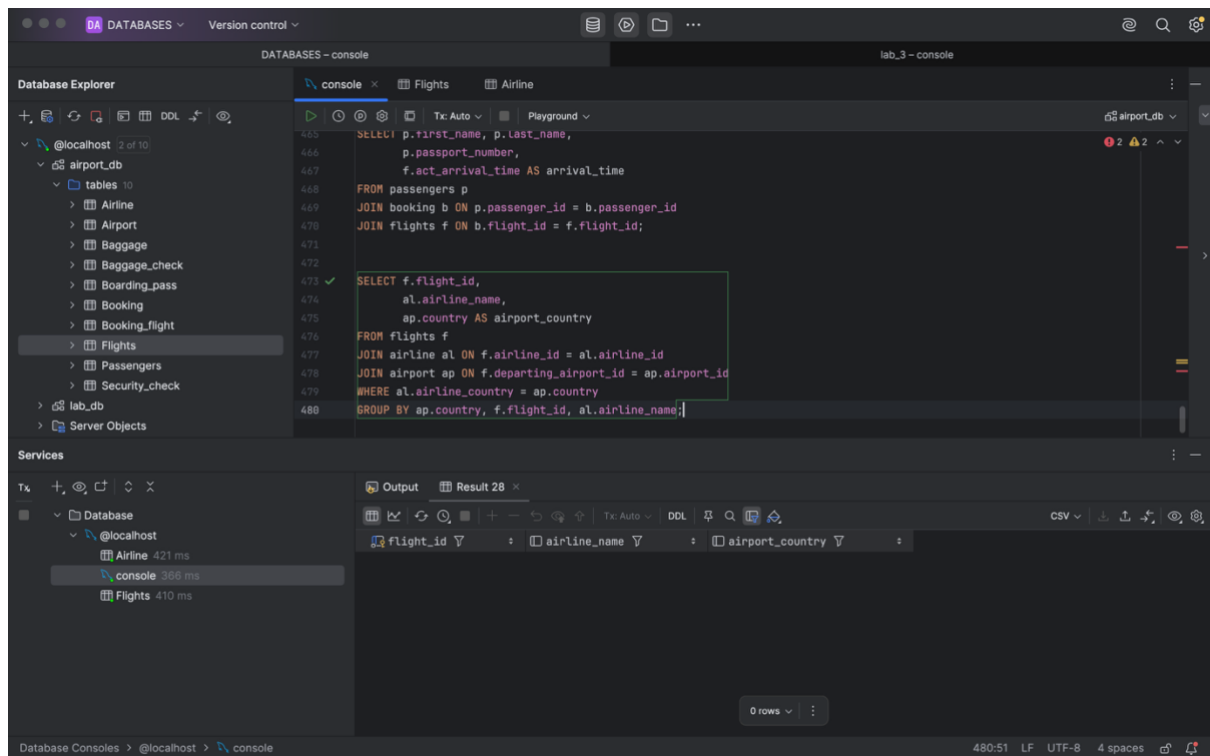
8. Display the passenger's full name, passport number, and the passenger's current time of arrival at the destination.

The screenshot shows the SQL Developer interface with a query executed in the console. The query selects the first and last names of passengers, their passport numbers, and the arrival time, filtered by passengers under 18 years old. The result set is displayed in the Output window.

```
455
456
457 SELECT p.first_name, p.last_name, ap.airport_name AS arrival_airport
458 FROM passengers p
459 JOIN booking b ON p.passenger_id = b.passenger_id
460 JOIN flights f ON b.flight_id = f.flight_id
461 JOIN airport ap ON f.arriving_airport_id = ap.airport_id
462 WHERE TIMESTAMPDIFF(YEAR, p.date_of_birth, CURDATE()) < 18;
463
464
465 SELECT p.first_name, p.last_name,
466        p.passport_number,
467        f.act_arrival_time AS arrival_time
468 FROM passengers p
469 JOIN booking b ON p.passenger_id = b.passenger_id
470 JOIN flights f ON b.flight_id = f.flight_id;
```

first_name	last_name	passport_number	arrival_time	
1	Anthony	Allison	P800057	2025-02-03 15:24:31
2	Mary	Porter	P800038	2025-10-05 00:53:39
3	Robert	Andrews	P800070	2025-06-02 14:27:32
4	Cory	Espinoza	P800033	2025-03-01 12:52:06
5	Devin	Richardson	P800060	2025-05-11 21:37:21
6	Marcia	Hanson	P800093	2025-10-30 06:23:56
7	Samantha	Kennedy	P800000	2025-06-20 09:15:44
8	Samantha	Kennedy	P800000	2025-10-30 06:23:56

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



Conclusion

In this lab work, I learned how to use different types of SQL JOINS to combine data from multiple related tables.

I practiced using **INNER JOIN**, **LEFT JOIN**, **RIGHT JOIN**, and **FULL JOIN** to extract connected information such as flight details, passengers, and airlines.

This helped me understand how relationships between tables work and how to write more complex queries to get meaningful results from a relational database.