

Kazakh-British Technical University

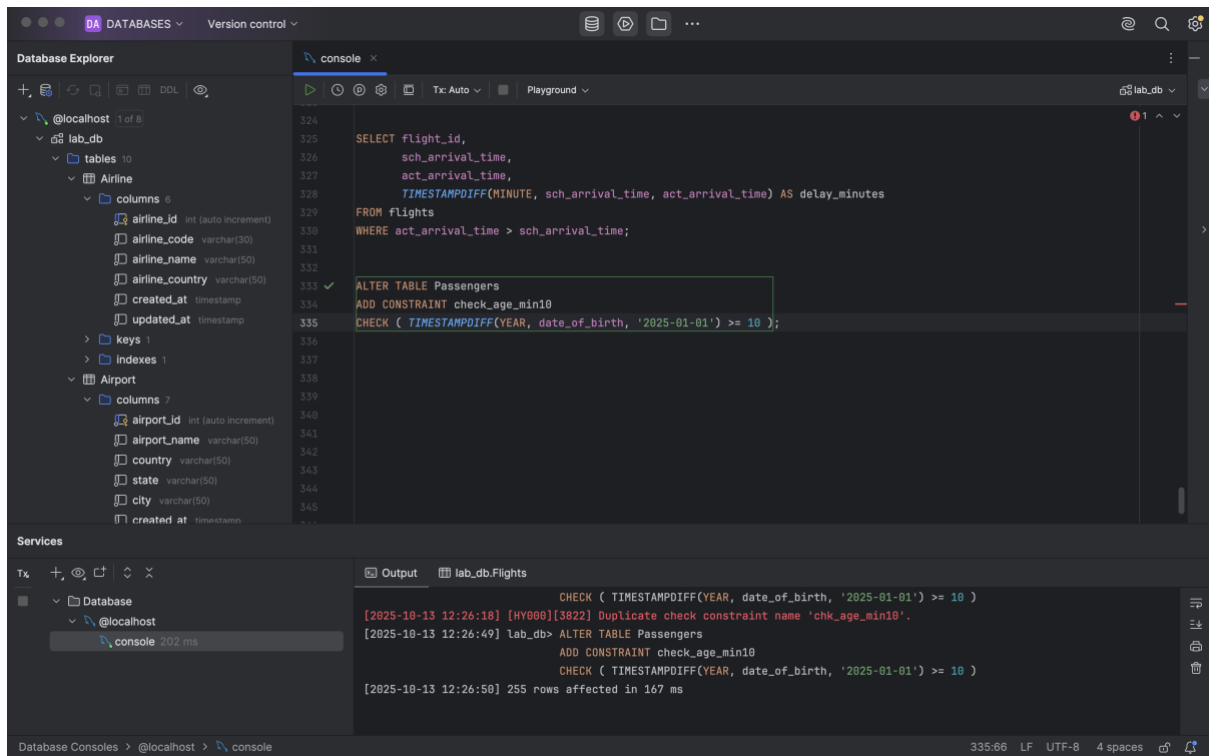
Databases

Laboratory work №5

Kislitsin Alexandr

23B031861

TASK 1. Add a CHECK constraint to passenger table to provide that passengers must be at least 10 years old.



The screenshot displays a database management interface with a 'Database Explorer' on the left, a 'console' window in the center, and a 'Services' section at the bottom. The 'Database Explorer' shows a database named 'lab_db' with tables 'Airline' and 'Airport'. The 'console' window contains the following SQL queries:

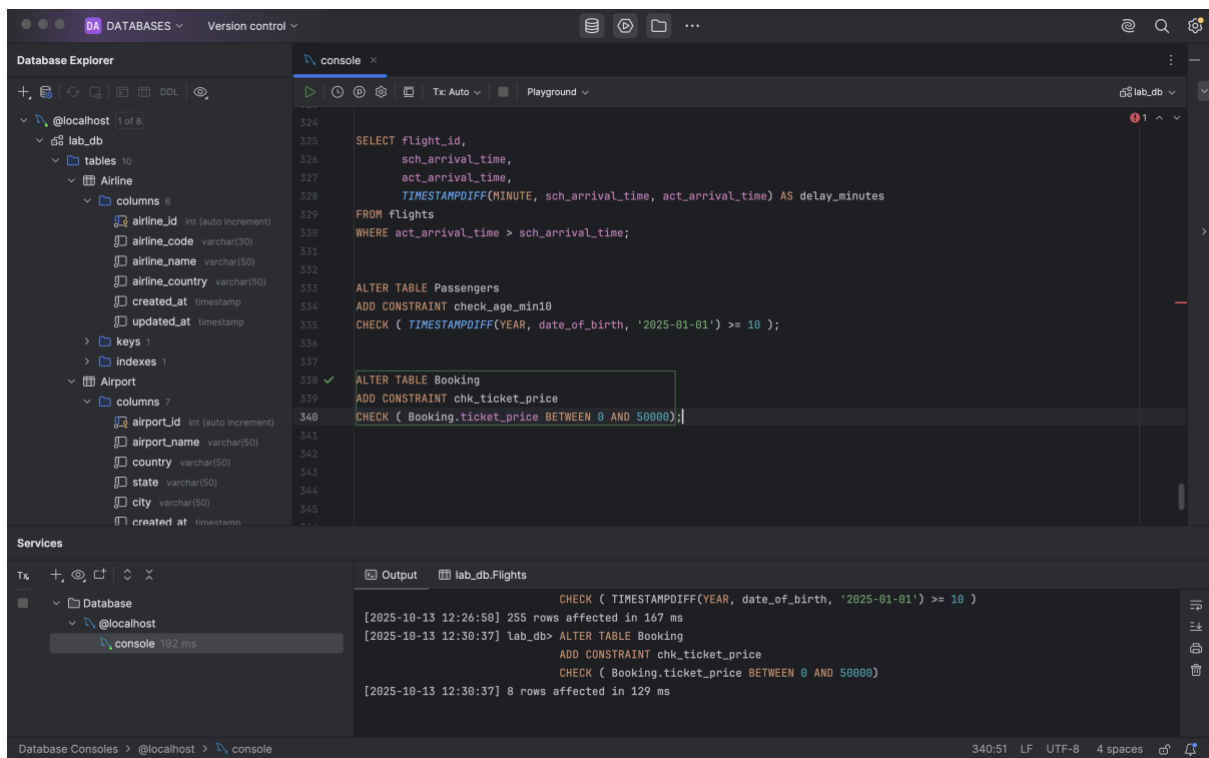
```
324  
325  
326 SELECT flight_id,  
327       sch_arrival_time,  
328       act_arrival_time,  
329       TIMESTAMPDIFF(MINUTE, sch_arrival_time, act_arrival_time) AS delay_minutes  
330 FROM flights  
331 WHERE act_arrival_time > sch_arrival_time;  
332  
333 ✓ ALTER TABLE Passengers  
334   ADD CONSTRAINT check_age_min10  
335   CHECK ( TIMESTAMPDIFF(YEAR, date_of_birth, '2025-01-01') >= 10 );
```

The 'Services' section shows the execution output for the last query:

```
Output lab_db.Flights  
CHECK ( TIMESTAMPDIFF(YEAR, date_of_birth, '2025-01-01') >= 10 )  
[2025-10-13 12:26:18] [HY000][3822] Duplicate check constraint name 'chk_age_min10'.  
[2025-10-13 12:26:49] lab_db> ALTER TABLE Passengers  
ADD CONSTRAINT check_age_min10  
CHECK ( TIMESTAMPDIFF(YEAR, date_of_birth, '2025-01-01') >= 10 )  
[2025-10-13 12:26:50] 255 rows affected in 167 ms
```

The status bar at the bottom indicates the file encoding is UTF-8 with 4 spaces.

TASK 2. Add a CHECK constraint to accept values in booking price not more than 50000tg and less than 0tg.



The screenshot displays a database management interface with a dark theme. On the left, the 'Database Explorer' pane shows a tree structure for a database named 'lab_db'. It includes tables 'Airline' and 'Airport', each with its columns and data types listed. The 'Airline' table has columns: airline_id (int, auto increment), airline_code (varchar(30)), airline_name (varchar(50)), airline_country (varchar(50)), created_at (timestamp), and updated_at (timestamp). The 'Airport' table has columns: airport_id (int, auto increment), airport_name (varchar(50)), country (varchar(50)), state (varchar(50)), city (varchar(50)), and created_at (timestamp). The main console area shows a series of SQL queries being executed. The first query is a SELECT statement from the 'flights' table, calculating delay minutes. The second query adds a CHECK constraint to the 'Passengers' table. The third query, which is highlighted, adds a CHECK constraint to the 'Booking' table. The bottom pane shows the output of these queries, indicating the number of rows affected and the execution time.

```
324
325
326 SELECT flight_id,
327        sch_arrival_time,
328        act_arrival_time,
329        TIMESTAMPDIFF(MINUTE, sch_arrival_time, act_arrival_time) AS delay_minutes
330 FROM flights
331 WHERE act_arrival_time > sch_arrival_time;
332
333
334 ALTER TABLE Passengers
335 ADD CONSTRAINT check_age_min10
336 CHECK ( TIMESTAMPDIFF(YEAR, date_of_birth, '2025-01-01') >= 10 );
337
338 ✓ ALTER TABLE Booking
339 ADD CONSTRAINT chk_ticket_price
340 CHECK ( Booking.ticket_price BETWEEN 0 AND 50000);
341
342
343
344
345
346
```

Services

TX +, -, <, >, ↺, ✕

Database

@localhost

console 192 ms

Output lab_db.Flights

```

CHECK ( TIMESTAMPDIFF(YEAR, date_of_birth, '2025-01-01') >= 10 )
[2025-10-13 12:26:50] 255 rows affected in 167 ms
[2025-10-13 12:30:37] lab_db> ALTER TABLE Booking
ADD CONSTRAINT chk_ticket_price
CHECK ( Booking.ticket_price BETWEEN 0 AND 50000)
[2025-10-13 12:30:37] 8 rows affected in 129 ms
```

Database Consoles > @localhost > console 340/51 LF UTF-8 4 spaces

TASK 3. Add a CHECK constraint to accept the luggage weight between 1 and 23 kg.

The screenshot shows a database IDE interface with the following components:

- Database Explorer:** Displays the database structure for 'lab_db'. It shows tables 'Airline' and 'Airport'. The 'Airline' table has columns: 'airline_id' (int, auto increment), 'airline_code' (varchar(30)), 'airline_name' (varchar(50)), 'airline_country' (varchar(50)), 'created_at' (timestamp), and 'updated_at' (timestamp). The 'Airport' table has columns: 'airport_id' (int, auto increment), 'airport_name' (varchar(50)), 'country' (varchar(50)), 'state' (varchar(50)), 'city' (varchar(50)), and 'created_at' (timestamp).
- Console:** Contains the following SQL commands:

```
341  
342  
343 ✓ ALTER TABLE Baggage  
344     ADD CONSTRAINT chk_baggage_weight  
345     CHECK ( weight_in_kg BETWEEN 1 AND 23);  
346  
347  
348 SELECT *  
349 FROM baggage  
350 WHERE weight_in_kg NOT BETWEEN 1 AND 23  
351        OR weight_in_kg IS NULL;  
352  
353 UPDATE baggage  
354 SET weight_in_kg = 20  
355 WHERE weight_in_kg NOT BETWEEN 1 AND 23  
356        OR weight_in_kg IS NULL;  
357  
358  
359  
360  
361  
362  
363
```
- Services:** Shows the execution results in the 'Output' tab for 'lab_db.Baggage'.

```
OR weight_in_kg IS NULL  
[2025-10-13 12:35:56] 1 row affected in 14 ms  
[2025-10-13 12:36:09] lab_db> ALTER TABLE Baggage  
                        ADD CONSTRAINT chk_baggage_weight  
                        CHECK ( weight_in_kg BETWEEN 1 AND 23)  
[2025-10-13 12:36:09] 8 rows affected in 47 ms
```

The status bar at the bottom indicates the file encoding is UTF-8 and the indentation is 4 spaces.

4. Add a CHECK constraint to ensure that all values in airport_name must have at least 10 characters.

The screenshot shows a database console interface with the following components:

- Database Explorer:** Displays the database structure for 'lab_db'. It shows two tables: 'Airline' and 'Airport'. The 'Airport' table has columns: 'airport_id' (int, auto-increment), 'airport_name' (varchar(50)), 'country' (varchar(50)), 'state' (varchar(50)), 'city' (varchar(50)), and 'created_at' (timestamp).
- Console:** Contains SQL commands and their execution results.
 - Line 354: `SET weight_in_kg = 20`
 - Line 355: `WHERE weight_in_kg NOT BETWEEN 1 AND 23`
 - Line 356: `OR weight_in_kg IS NULL;`
 - Line 358: `ALTER TABLE Airport`
 - Line 359: `ADD CONSTRAINT chk_airport_name_length`
 - Line 360: `CHECK (LENGTH(airport_name) >= 10);`
- Output:** Shows the execution results of the SQL commands.
 - Line 358: `[HY000][3819] Check constraint 'chk_airport_name_length' is violated.`
 - Line 359: `[2025-10-13 12:38:07] [HY000][3819] Check constraint 'chk_airport_name_length' is violated.`
 - Line 360: `[2025-10-13 12:39:58] lab_db> ALTER TABLE Airport`
 - Line 361: `ADD CONSTRAINT chk_airport_name_length`
 - Line 362: `CHECK (LENGTH(airport_name) >= 10)`
 - Line 363: `[2025-10-13 12:39:58] [HY000][3819] Check constraint 'chk_airport_name_length' is violated.`

5. Add UNIQUE constraint to some columns in each table in database.

The screenshot displays a database management interface with the following components:

- Database Explorer:** Shows a tree view of the database structure. The 'Passengers' table is selected, and its columns are listed: `passenger_id` (int, auto-increment), `first_name` (varchar(50)), `last_name` (varchar(50)), `date_of_birth` (date), `gender` (varchar(50)), `country_of_citizenship` (varchar), `country_of_residence` (varchar), `passport_number` (varchar(20)), `created_at` (timestamp), and `updated_at` (timestamp).
- Console:** Contains the following SQL commands:

```
ALTER TABLE Passengers
ADD CONSTRAINT uq_passport UNIQUE (passport_number);

ALTER TABLE Airline
ADD CONSTRAINT uq_airline_code UNIQUE (airline_code);

ALTER TABLE Airport
ADD CONSTRAINT uq_airport_name UNIQUE (airport_name);
```
- Services:** Shows the execution output for the commands:

```
[2025-10-13 12:41:39] lab_db> ALTER TABLE Passengers
ADD CONSTRAINT uq_passport UNIQUE (passport_number)
[2025-10-13 12:41:39] [HY000][1831] Duplicate index 'uq_passport' defined on the table 'lab_db.passengers'. This is deprecate
[2025-10-13 12:41:39] completed in 103 ms
[2025-10-13 12:43:39] lab_db> ALTER TABLE Airline
ADD CONSTRAINT uq_airline_code UNIQUE (airline_code)
[2025-10-13 12:43:39] completed in 127 ms
[2025-10-13 12:44:36] lab_db> ALTER TABLE Airport
ADD CONSTRAINT uq_airport_name UNIQUE (airport_name)
[2025-10-13 12:44:36] completed in 46 ms
```

6. Add a CHECK constraint to ensure that male passengers must be at least 18 years old and female passengers must be 19 years old.

The screenshot shows a database IDE with the following components:

- Database Explorer:** Shows a tree view of the database structure. The 'Passengers' table is selected, and its columns are listed: `passenger_id` (int, auto-increment), `first_name` (varchar(50)), `last_name` (varchar(50)), `date_of_birth` (date), `gender` (varchar(50)), `country_of_citizenship` (varchar(50)), `country_of_residence` (varchar(50)), `passport_number` (varchar(20)), `created_at` (timestamp), and `updated_at` (timestamp).
- Console:** Displays the SQL script being executed:

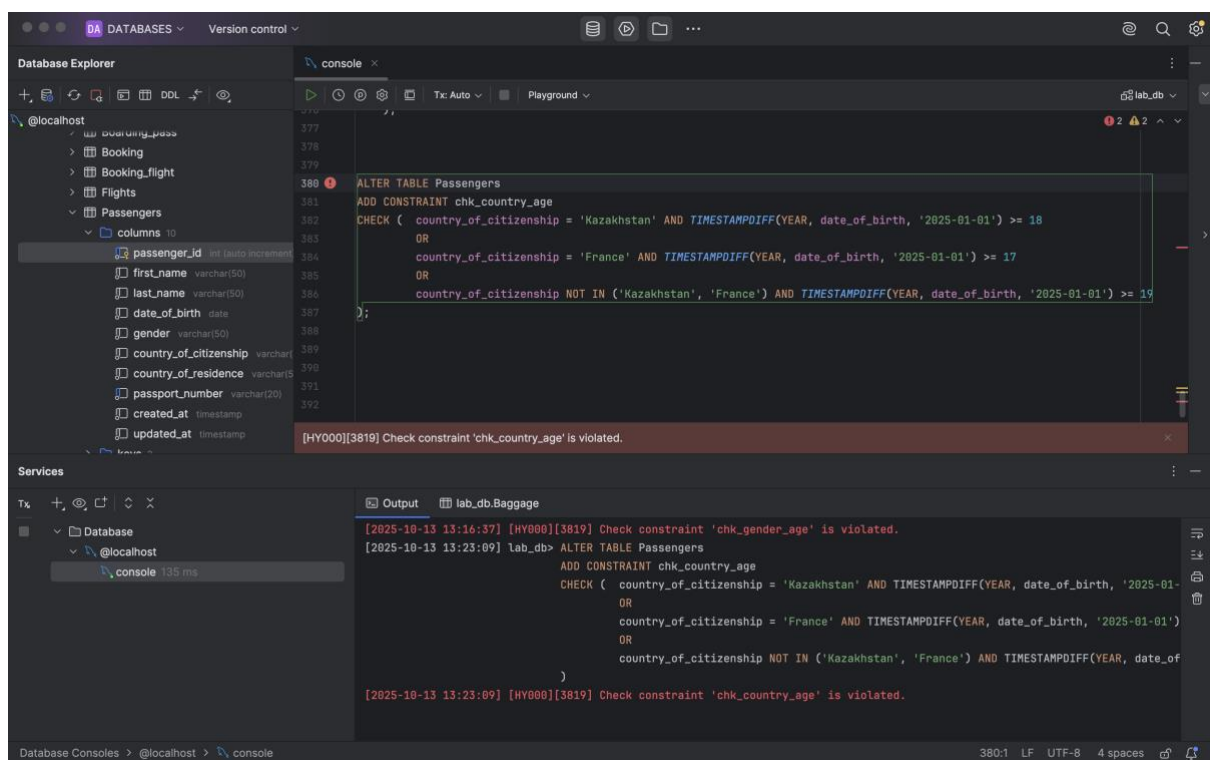
```
361 ALTER TABLE Passengers
362 ADD CONSTRAINT uq_passport UNIQUE (passport_number);
363
364 ALTER TABLE Airline
365 ADD CONSTRAINT uq_airline_code UNIQUE (airline_code);
366
367 ALTER TABLE Airport
368 ADD CONSTRAINT uq_airport_name UNIQUE (airport_name);
369
370
371 ALTER TABLE Passengers
372 ADD CONSTRAINT chk_gender_age
373 CHECK ( gender = 'male' AND TIMESTAMPDIFF(YEAR, date_of_birth, '2025-01-01' >= 18)
374 OR
375 gender = 'female' AND TIMESTAMPDIFF(YEAR, date_of_birth, '2025-01-01' >= 19)
376 );
377
```

A red error message is displayed below the script: `[HY000][3819] Check constraint 'chk_gender_age' is violated.`
- Services:** Shows the 'Output' tab with the following log entries:

```
[2025-10-13 13:15:26] [HY000][3814] An expression of a check constraint 'chk_gender_age' contains disallowed function: curdate()
[2025-10-13 13:16:37] lab_db> ALTER TABLE Passengers
ADD CONSTRAINT chk_gender_age
CHECK ( gender = 'male' AND TIMESTAMPDIFF(YEAR, date_of_birth, '2025-01-01' >= 18)
OR
gender = 'female' AND TIMESTAMPDIFF(YEAR, date_of_birth, '2025-01-01' >= 19)
)
[2025-10-13 13:16:37] [HY000][3819] Check constraint 'chk_gender_age' is violated.
```

7. Add a CHECK constraint to add rule as follow (use column `country_of_citizenship`):

- **Passengers from Kazakhstan must be at least 18 years old.**
- **Passengers from France must be at least 17 years old.**
- **Passengers from other countries must be at least 19 years old.**



8. Add a `ticket_discount` column to table `booking` and a `CHECK` constraint to apply some discount based on ticket price and created time:

- the constraint applies a 5% discount for tickets created after 2024-01-01,

and 10% discount for tickets created before 2024-01-01.

The screenshot shows a database IDE with the following components:

- Database Explorer:** Shows a database named `lab_db` with tables `Booking`, `Booking_flight`, `Flights`, and `Passengers`. The `Passengers` table is expanded, showing columns: `passenger_id` (int, auto increment), `first_name` (varchar(50)), `last_name` (varchar(50)), `date_of_birth` (date), `gender` (varchar(50)), `country_of_citizenship` (varchar(50)), `country_of_residence` (varchar(50)), `passport_number` (varchar(20)), `created_at` (timestamp), and `updated_at` (timestamp).
- Console:** Contains the following SQL commands:

```
381 ADD CONSTRAINT chk_country_age
382 );
383
384 ALTER TABLE Booking
385 ADD COLUMN ticket_discount DECIMAL(5,2);
386
387 ALTER TABLE Booking
388 ADD CONSTRAINT chk_ticket_discount
389 CHECK ( created_at < '2024-01-01' AND Booking.ticket_discount = 10
390 OR
391 created_at >= '2024-01-01' AND Booking.ticket_discount = 5
392 );
```
- Services:** Shows the execution output for the `lab_db.Baggage` table:

```
[2025-10-13 13:36:49] completed in 43 ms
[2025-10-13 13:36:54] lab_db> ALTER TABLE Booking
ADD CONSTRAINT chk_ticket_discount
CHECK ( created_at < '2024-01-01' AND Booking.ticket_discount = 10
OR
created_at >= '2024-01-01' AND Booking.ticket_discount = 5
)
[2025-10-13 13:36:54] 8 rows affected in 63 ms
```