

1\_

The screenshot shows a database console interface with a dark theme. On the left, the 'Database Explorer' pane shows a PostgreSQL database structure with a 'public' schema containing several tables, including 'flights'. The 'flights' table is expanded, showing columns like '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 'console' pane on the right shows a SQL script being executed: 

```
1 BEGIN;
2
3 DELETE FROM booking_flight WHERE booking_id = 123;
4 DELETE FROM booking WHERE booking_id = 123;
5
6 COMMIT;
7
```

 The 'Services' pane at the bottom shows the execution log for the 'console' transaction, which completed successfully at 11:16:17. The log includes timestamps and durations for each step: BEGIN (3 ms), DELETE FROM booking\_flight (3 ms), DELETE FROM booking (6 ms), and COMMIT (1 ms).

The screenshot shows the same database console interface as the previous one, but with a different SQL script being executed in the 'console' pane: 

```
1 BEGIN;
2
3 DELETE FROM booking_flight WHERE booking_id = 1199191;
4 DELETE FROM booking WHERE booking_id = 1199191;
5
6 ROLLBACK;
7
```

 The 'Services' pane at the bottom shows the execution log for the 'console' transaction, which completed successfully at 12:20:03. The log includes timestamps and durations for each step: BEGIN (3 ms), DELETE FROM booking\_flight (1 ms), DELETE FROM booking (1 ms), and ROLLBACK (2 ms).

2\_

The screenshot shows a database console interface with a 'Database Explorer' on the left and a 'console' window on the right. The 'Database Explorer' shows a tree structure for 'postgres@localhost' with a 'public' schema containing several tables. The 'console' window shows a SQL script being executed: 

```
1 BEGIN;
2
3 UPDATE flights
4 SET scheduled_departure = '2025-12-05',
5   scheduled_arrival = '2025-12-06'
6 WHERE flight_id = 101;
7
8 COMMIT;
9
```

 The 'Output' pane shows the execution results: 

```
[2025-12-03 12:20:03] dbwork.public> ROLLBACK
[2025-12-03 12:20:03] completed in 2 ms
[2025-12-03 12:20:39] dbwork.public> BEGIN
[2025-12-03 12:20:39] completed in 1 ms
[2025-12-03 12:20:39] dbwork.public> UPDATE flights
SET scheduled_departure = '2025-12-05',
scheduled_arrival = '2025-12-06'
WHERE flight_id = 101
[2025-12-03 12:20:39] 1 row affected in 5 ms
[2025-12-03 12:20:39] dbwork.public> COMMIT
[2025-12-03 12:20:39] completed in 2 ms
```

The screenshot shows the same database console interface. The 'console' window shows a SQL script being executed: 

```
1 BEGIN;
2
3 UPDATE flights
4 SET scheduled_departure = '1999-12-05',
5   scheduled_arrival = '1999-12-06'
6 WHERE flight_id = 101;
7
8 ROLLBACK;
9
```

 The 'Output' pane shows the execution results: 

```
[2025-12-03 12:20:39] dbwork.public> COMMIT
[2025-12-03 12:20:39] completed in 2 ms
[2025-12-03 12:21:08] dbwork.public> BEGIN
[2025-12-03 12:21:08] completed in 1 ms
[2025-12-03 12:21:08] dbwork.public> UPDATE flights
SET scheduled_departure = '1999-12-05',
scheduled_arrival = '1999-12-06'
WHERE flight_id = 101
[2025-12-03 12:21:08] 1 row affected in 3 ms
[2025-12-03 12:21:08] dbwork.public> ROLLBACK
[2025-12-03 12:21:08] completed in 1 ms
```

3\_

The screenshot shows a database console interface with a dark theme. On the left, the 'Database Explorer' pane shows a PostgreSQL database structure with tables like 'airline', 'airport', 'baggage', 'boarding\_pass', 'booking', 'booking\_flight', and 'flights'. The 'flights' table is selected, showing its 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 main console area displays a SQL transaction being executed. The SQL code is as follows:

```
1 BEGIN;
2
3 UPDATE booking b
4 SET price = price * 0.9,
5     update_at = CURRENT_DATE
6 FROM booking_flight bf
7 WHERE b.booking_id = bf.booking_id
8     AND bf.flight_id = 101;
9
10 COMMIT;
```

The console output shows the transaction being executed successfully. The output is as follows:

```
[2025-12-03 12:22:08] dbwork.public> BEGIN
[2025-12-03 12:22:08] completed in 2 ms
[2025-12-03 12:22:08] dbwork.public> UPDATE booking b
SET price = price * 0.9,
update_at = CURRENT_DATE
FROM booking_flight bf
WHERE b.booking_id = bf.booking_id
AND bf.flight_id = 101
[2025-12-03 12:22:08] completed in 4 ms
[2025-12-03 12:22:08] dbwork.public> COMMIT
[2025-12-03 12:22:08] completed in 1 ms
```

The screenshot shows the same database console interface as the previous one, but with a different SQL transaction being executed. The SQL code is as follows:

```
1 BEGIN;
2
3 UPDATE booking b
4 SET price = price * 0.9,
5     update_at = CURRENT_DATE
6 FROM booking_flight bf
7 WHERE b.booking_id = bf.booking_id
8     AND bf.flight_id = 101;
9
10 ROLLBACK;
```

The console output shows the transaction being executed successfully. The output is as follows:

```
[2025-12-03 12:22:36] dbwork.public> BEGIN
[2025-12-03 12:22:36] completed in 1 ms
[2025-12-03 12:22:36] dbwork.public> UPDATE booking b
SET price = price * 0.9,
update_at = CURRENT_DATE
FROM booking_flight bf
WHERE b.booking_id = bf.booking_id
AND bf.flight_id = 101
[2025-12-03 12:22:36] completed in 2 ms
[2025-12-03 12:22:36] dbwork.public> ROLLBACK
[2025-12-03 12:22:36] completed in 2 ms
```

4\_

The screenshot shows a database console interface with a sidebar on the left displaying a database schema. The main area contains a SQL script with the following commands:

```
1 BEGIN;
2
3 UPDATE passengers
4 SET first_name = 'John',
5     last_name = 'Doe',
6     date_of_birth = '1990-01-01',
7     country_of_residence = 'USA',
8     update_at = CURRENT_DATE
9 WHERE passenger_id = 123;
10
11 UPDATE booking b
12 SET update_at = CURRENT_DATE
13 WHERE b.passenger_id = 123;
14
15 COMMIT;
```

The bottom panel shows the execution output:

```
Output Result 19
[2025-12-03 12:24:53] 1 row affected in 5 ms
[2025-12-03 12:24:53] dbwork.public> UPDATE booking b
SET update_at = CURRENT_DATE
WHERE b.passenger_id = 123
[2025-12-03 12:24:53] completed in 1 ms
[2025-12-03 12:24:53] dbwork.public> COMMIT
[2025-12-03 12:24:53] completed in 1 ms
```

5\_

The screenshot shows a database console interface with a sidebar on the left displaying a database schema. The main area contains a SQL script with the following commands:

```
1 BEGIN;
2
3 INSERT INTO passengers(passenger_id, first_name, last_name, date_of_birth, country_of_residence, created_at, update_at)
4 VALUES (passenger_id 2001, 'first_name' 'Alice', 'last_name' 'Smith', 'date_of_birth' '1995-06-15', 'country_of_residence' 'USA', (created_at CURRENT_DATE, (update_at CURRENT_DATE));
5
6 INSERT INTO booking(booking_id, passenger_id, booking_platform, created_at, update_at, status, price)
7 VALUES (booking_id 3001, passenger_id 2001, booking_platform 'OnLine', created_at CURRENT_DATE, update_at CURRENT_DATE, status 'confirmed', price 350.00);
8
9 INSERT INTO booking_flight(booking_flight_id, booking_id, flight_id)
10 VALUES (booking_flight_id 1001, booking_id 3001, flight_id 101);
11
12 COMMIT;
```

The bottom panel shows the execution output:

```
Output Result 19
[2025-12-03 12:32:21] completed in 2 ms
[2025-12-03 12:32:21] dbwork.public> INSERT INTO passengers(passenger_id, first_name, last_name, date_of_birth, country_of_residence, created_at, update_at)
VALUES (2001, 'Alice', 'Smith', '1995-06-15', 'USA', CURRENT_DATE, CURRENT_DATE)
[2025-12-03 12:32:21] 1 row affected in 1 ms
[2025-12-03 12:32:21] dbwork.public> INSERT INTO booking(booking_id, passenger_id, booking_platform, created_at, update_at, status, price)
VALUES (3001, 2001, 'OnLine', CURRENT_DATE, CURRENT_DATE, 'confirmed', 350.00)
[2025-12-03 12:32:21] 1 row affected in 2 ms
[2025-12-03 12:32:21] dbwork.public> INSERT INTO booking_flight(booking_flight_id, booking_id, flight_id)
VALUES (1001, 3001, 101)
[2025-12-03 12:32:21] 1 row affected in 2 ms
[2025-12-03 12:32:21] dbwork.public> COMMIT
[2025-12-03 12:32:21] completed in 1 ms
```

6\_

The screenshot shows a database console interface with a dark theme. On the left, the 'Database Explorer' pane shows a tree view of a PostgreSQL database named 'postgres@localhost'. The 'public' schema is expanded, showing tables like 'airline', 'airport', 'baggage', 'baggage\_check', 'boarding\_pass', 'booking', 'booking\_flight', 'flights', 'passengers', 'security\_check', 'views', and 'routes'. The 'booking\_flight' table is selected. The main console area shows a SQL query being executed: 

```
1 BEGIN;  
2  
3 UPDATE booking b  
4 SET price = price + 50,  
5 update_at = CURRENT_DATE  
6 FROM booking_flight bf  
7 WHERE b.booking_id = bf.booking_id  
8 AND bf.flight_id = 101;  
9  
10 COMMIT;  
11
```

 The 'Output' pane on the right shows the execution results: 

```
[2025-12-03 12:34:52] dbwork.public> BEGIN  
[2025-12-03 12:34:52] completed in 2 ms  
[2025-12-03 12:34:52] dbwork.public> UPDATE booking b  
SET price = price + 50,  
update_at = CURRENT_DATE  
FROM booking_flight bf  
WHERE b.booking_id = bf.booking_id  
AND bf.flight_id = 101  
[2025-12-03 12:34:52] 1 row affected in 4 ms  
[2025-12-03 12:34:52] dbwork.public> COMMIT  
[2025-12-03 12:34:52] completed in 3 ms
```

 The 'Services' pane at the bottom shows the 'console' service running.

7\_

The screenshot shows the same database console interface as above, but with a different SQL query being executed: 

```
1 BEGIN;  
2  
3 UPDATE baggage  
4 SET weight_in_kg = 23.5,  
5 update_date = CURRENT_DATE  
6 WHERE baggage_id = 401;  
7  
8 COMMIT;  
9
```

 The 'Output' pane on the right shows the execution results: 

```
[2025-12-03 12:36:00] 1 row affected in 4 ms  
[2025-12-03 12:36:00] dbwork.public> COMMIT  
[2025-12-03 12:36:00] completed in 3 ms  
[2025-12-03 12:36:00] dbwork.public> BEGIN  
[2025-12-03 12:36:00] completed in 2 ms  
[2025-12-03 12:36:00] dbwork.public> UPDATE baggage  
SET weight_in_kg = 23.5,  
update_date = CURRENT_DATE  
WHERE baggage_id = 401  
[2025-12-03 12:36:00] 1 row affected in 4 ms  
[2025-12-03 12:36:00] dbwork.public> COMMIT  
[2025-12-03 12:36:00] completed in 2 ms
```

 The 'Services' pane at the bottom shows the 'console' service running.

8\_

The screenshot shows a database console interface with a left sidebar for 'Database Explorer' and a main area for SQL execution. The 'Database Explorer' shows a tree structure with 'postgres@localhost' selected, containing 'dbwork', 'information\_schema', 'pg\_catalog', and 'public'. Under 'public', there are tables like 'airline', 'airport', 'baggage', 'baggage\_check', 'boarding\_pass', 'booking', 'booking\_flight', 'flights', 'passengers', 'security\_check', and 'views'. The 'booking\_flight' table is selected. The main area shows a SQL query being executed in a console window:

```
1 BEGIN;
2
3 UPDATE booking
4 SET price = price - 50,
5     update_at = CURRENT_DATE
6 WHERE passenger_id = 201;
7
8 COMMIT;
```

The 'Output' pane shows the execution results:

```
[2025-12-03 12:36:00] dbwork.public> COMMIT
[2025-12-03 12:36:00] completed in 2 ms
[2025-12-03 12:36:57] dbwork.public> BEGIN
[2025-12-03 12:36:57] completed in 1 ms
[2025-12-03 12:36:57] dbwork.public> UPDATE booking
SET price = price - 50,
update_at = CURRENT_DATE
WHERE passenger_id = 201
[2025-12-03 12:36:57] 1 row affected in 2 ms
[2025-12-03 12:36:57] dbwork.public> COMMIT
[2025-12-03 12:36:57] completed in 2 ms
```

9\_

The screenshot shows a database console interface with a left sidebar for 'Database Explorer' and a main area for SQL execution. The 'Database Explorer' shows a tree structure with 'postgres@localhost' selected, containing 'dbwork', 'information\_schema', 'pg\_catalog', and 'public'. Under 'public', there are tables like 'airline', 'airport', 'baggage', 'baggage\_check', 'boarding\_pass', 'booking', 'booking\_flight', 'flights', 'passengers', 'security\_check', and 'views'. The 'booking\_flight' table is selected. The main area shows a SQL query being executed in a console window:

```
1 BEGIN;
2
3 INSERT INTO flights(flight_id, flight_no, scheduled_departure, scheduled_arrival,
4 departure_airport_id, arrival_airport_id, departing_gate, arriving_gate,
5 airline_id, status, actual_departure, actual_arrival, created_at, update_at)
6 VALUES (flight_id 1002, flight_no 'A123', scheduled_departure '2025-12-10',
7         scheduled_arrival '2025-12-11', departure_airport_id 1, arrival_airport_id 2,
8         departing_gate 'A1', arriving_gate 'B2', airline_id 1, status 'Scheduled',
9         actual_departure NULL, actual_arrival NULL, created_at CURRENT_DATE, update_at CURRENT_DATE);
10
11 UPDATE booking_flight
12 SET flight_id = 1002
13 WHERE flight_id = 101;
14
15 COMMIT;
```

The 'Output' pane shows the execution results:

```
[2025-12-03 12:39:51] 1 row affected in 5 ms
[2025-12-03 12:39:51] dbwork.public> UPDATE booking_flight
SET flight_id = 1002
WHERE flight_id = 101
[2025-12-03 12:39:51] 1 row affected in 2 ms
[2025-12-03 12:39:51] dbwork.public> COMMIT
[2025-12-03 12:39:51] completed in 2 ms
[2025-12-03 12:39:51] dbwork.public> ROLLBACK
[2025-12-03 12:39:51] [25P01] нет незавершенной транзакции
[2025-12-03 12:39:51] completed in 11 ms
```