**DO $$**

**BEGIN**

**IF EXISTS (SELECT \* FROM leadcustomer WHERE EXISTS (SELECT CustomerID, Surname WHCMP-4010B Database Systems**

**Student Name:** 

**Reg No:** 

**Marker:** 

Before you begin running your SQL commands, delete all your own test data and insert my own test data (inserts2020.txt); use tests at the end of the document to ensure all data has being inserted correctly. Then proceed to run each tasks (1 to 8) as instructed below. Where your output is expected (in red), insert the SQL statement you run and then the output of running the statement on this document. It may be that a task requires you to run more than one SQL statement; in such cases insert in order the statements you run to achieve the task. You can insert any screenshots of running the tasks in PGAdmin as they will contain both the SQL and the output, but please make sure any text is of a size that can be read easily for marking. If the running of the SQL command results on an error reported by the SQL environment then record that error (i.e. copy it and paste the screenshot with the error shown). When asked, insert also the contents of particular tables (i.e. show the result of selecting all tuples from that table). If you have not implemented a particular task write ‘NOT DONE’ as the output. Please **do not delete** the marks allocated to each section (in blue) as those will be used by markers. An example of how to fill the form is in the appendix.

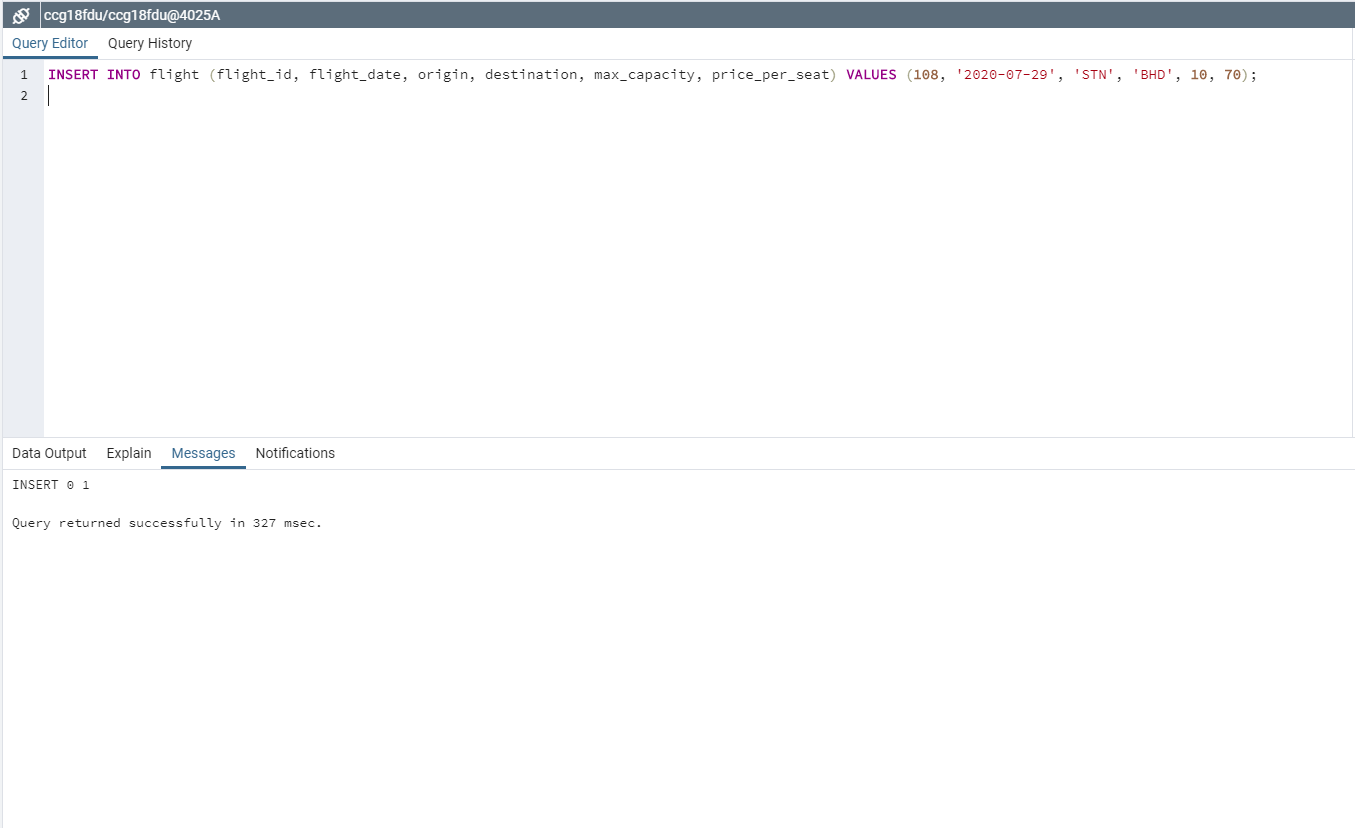
--------------------------------------------------------------------------------------------------------------------------------------------------

**Testing task 1**

--------------------------------------------------------------------------------------------------------------------------------------------------

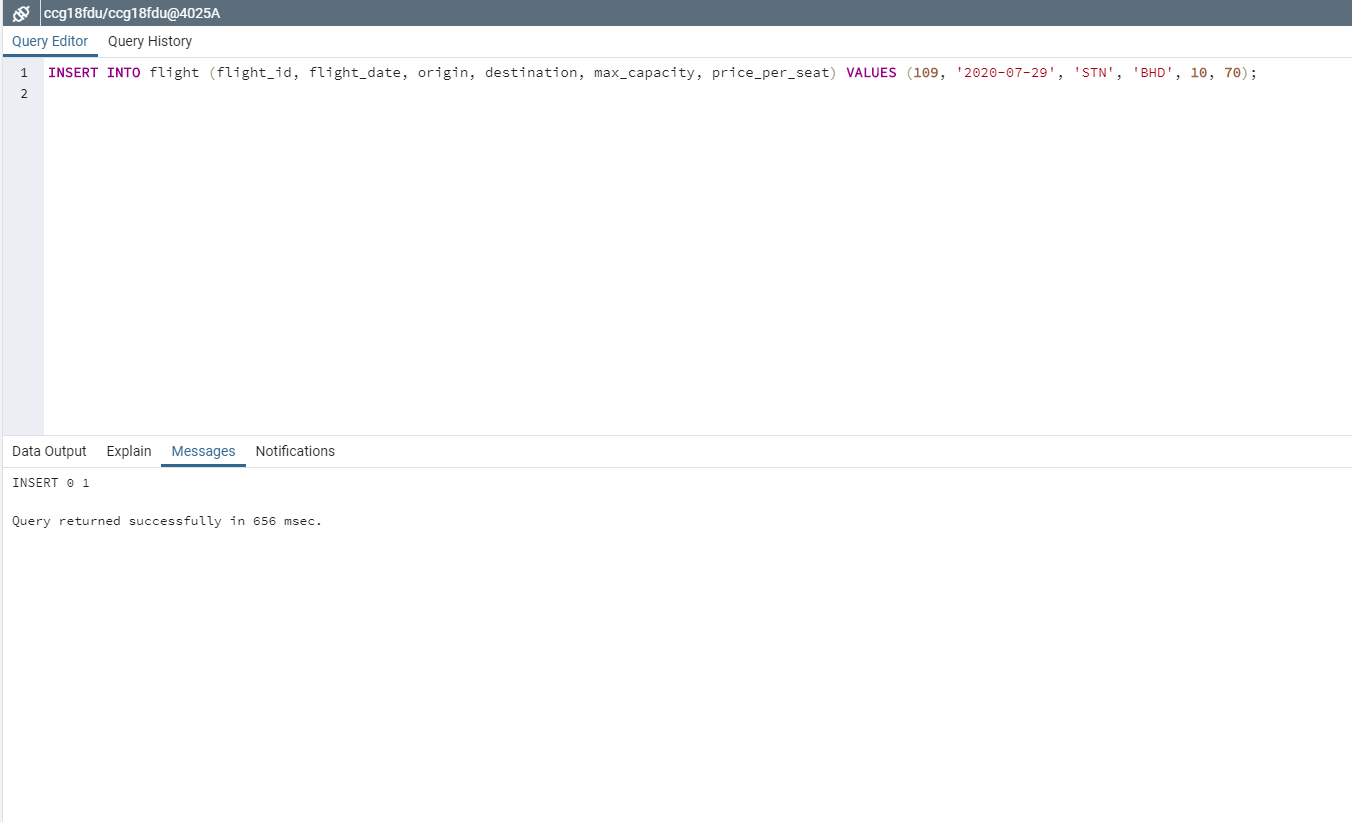
**1.b Given flight details create new flight record**

1. Create a new flight with values: flight ID = 108, origin = 'STN', destination = 'BHD', flight date = '29/7/2020', maximum capacity = 10, and price per seat = 70.



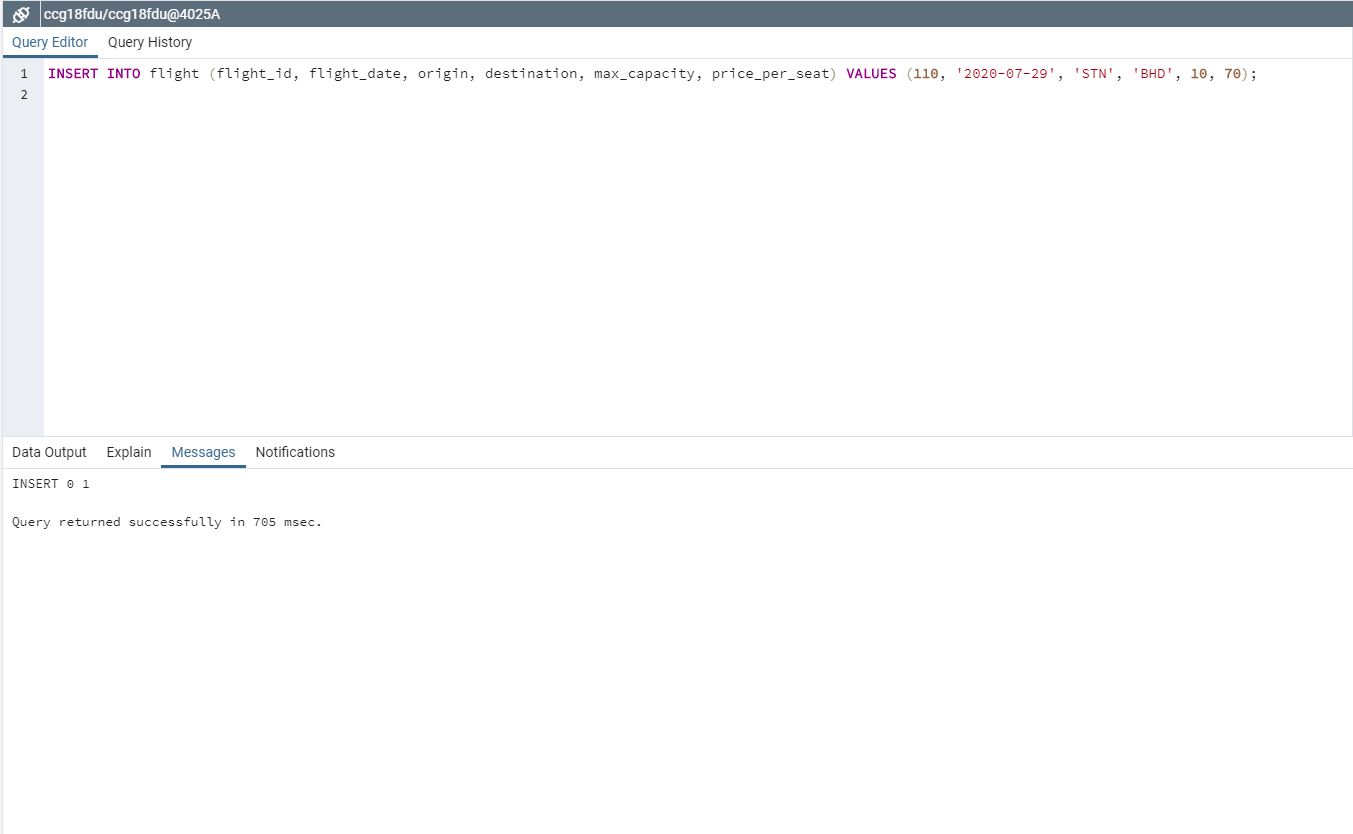
Marks: /1

1. Create a new flight with values: flight ID = 109, origin = 'STN', destination = 'BHD', flight date = '29/7/2019', maximum capacity = 10, and price per seat = 70.



Marks: /1

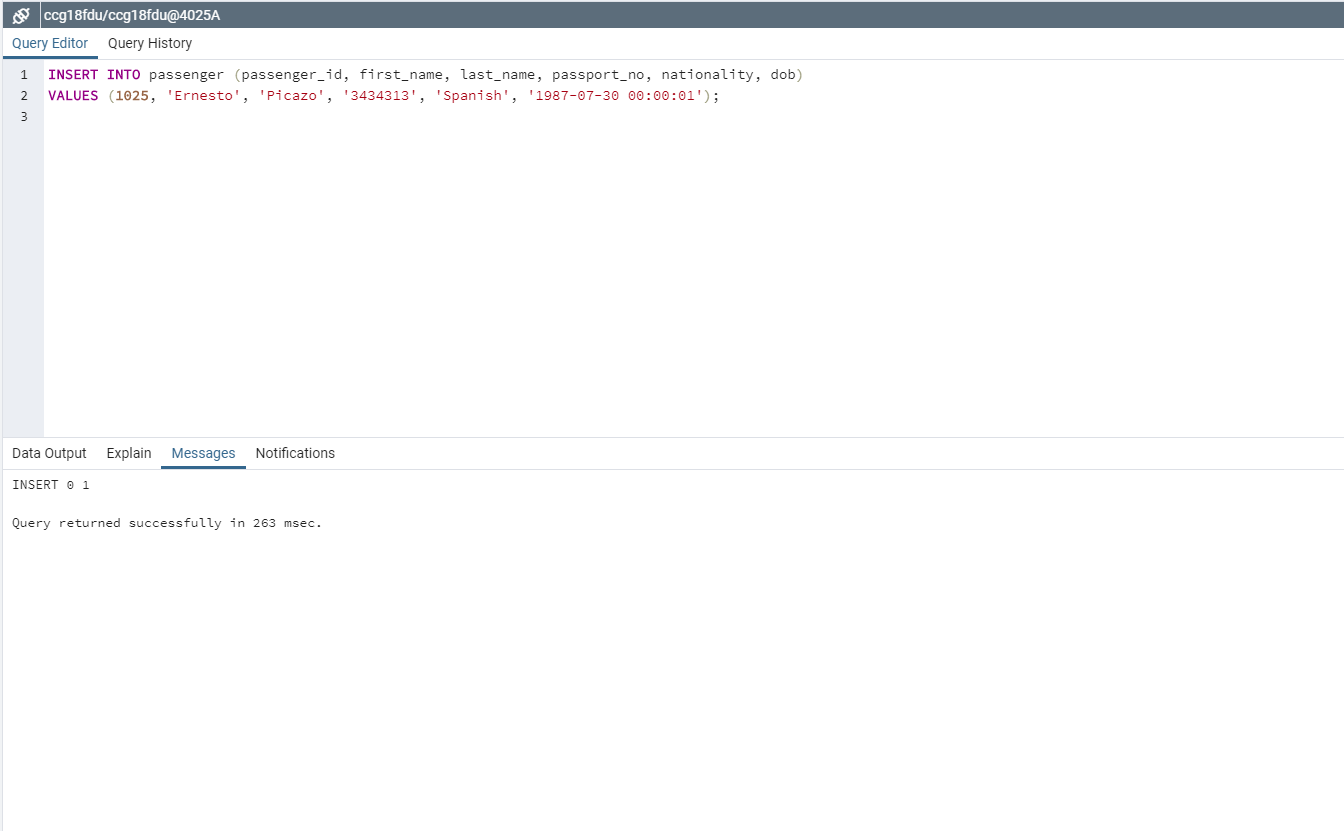
1. Create a new flight with values: flight ID = 110, origin = 'STN', destination = ‘STN’, flight date = '29/7/2020', maximum capacity = 10, and price per seat = 70.



Marks: /1

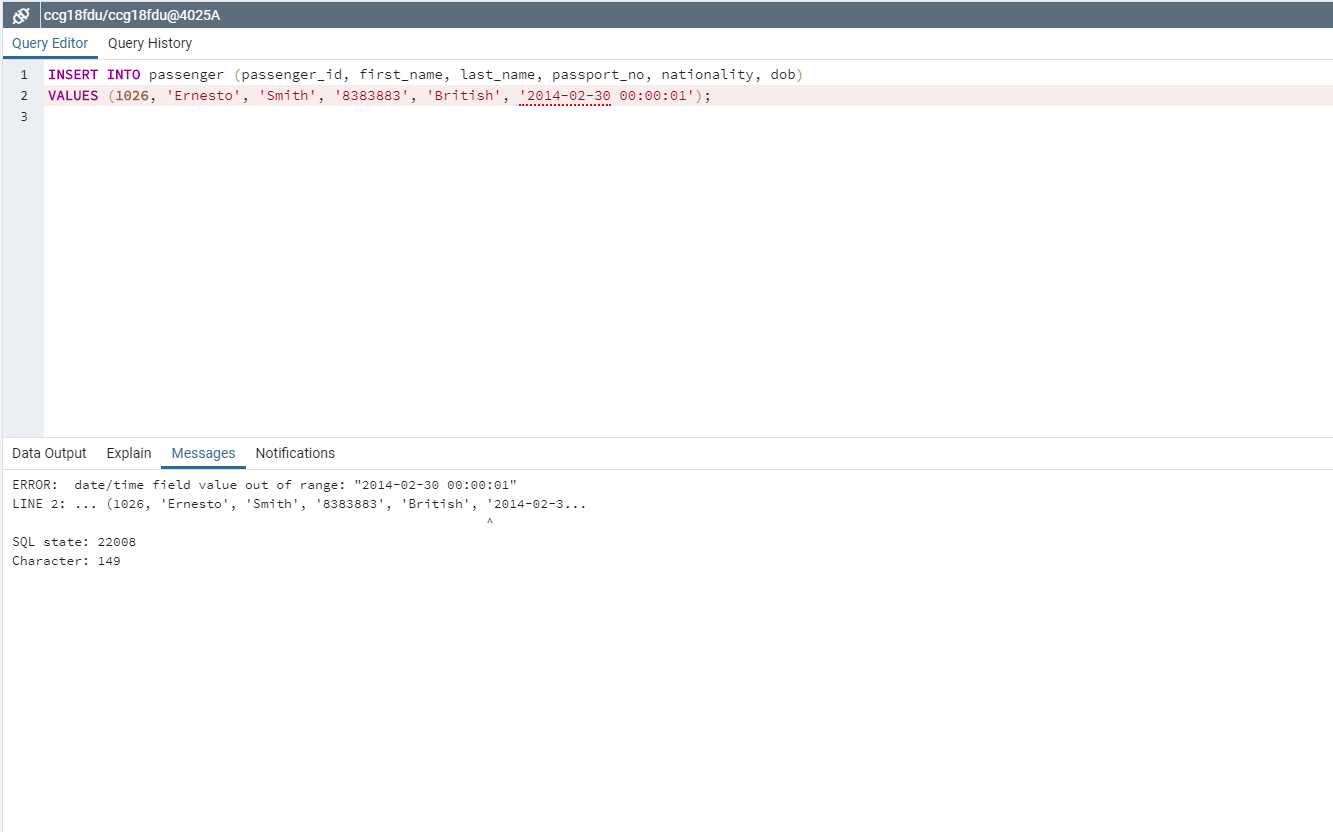
**1.c Given passenger details create new passenger records**

1. Create a new passenger with values: PassengerID = 1025, FirstName = 'Ernesto', Surname = 'Picazo', PassportNo = ' 3434313', Nationality 'Spanish', DoB='30/7/1987'.



Marks: /1

1. Create a new passenger with values: PassengerID = 1026, FirstName = 'Ernesto', Surname = ‘Smith’, PassportNo = '8383883', Nationality= ‘British’, DoB='30/2/2014'.



Marks: /1

--------------------------------------------------------------------------------------------------------------------------------------------------

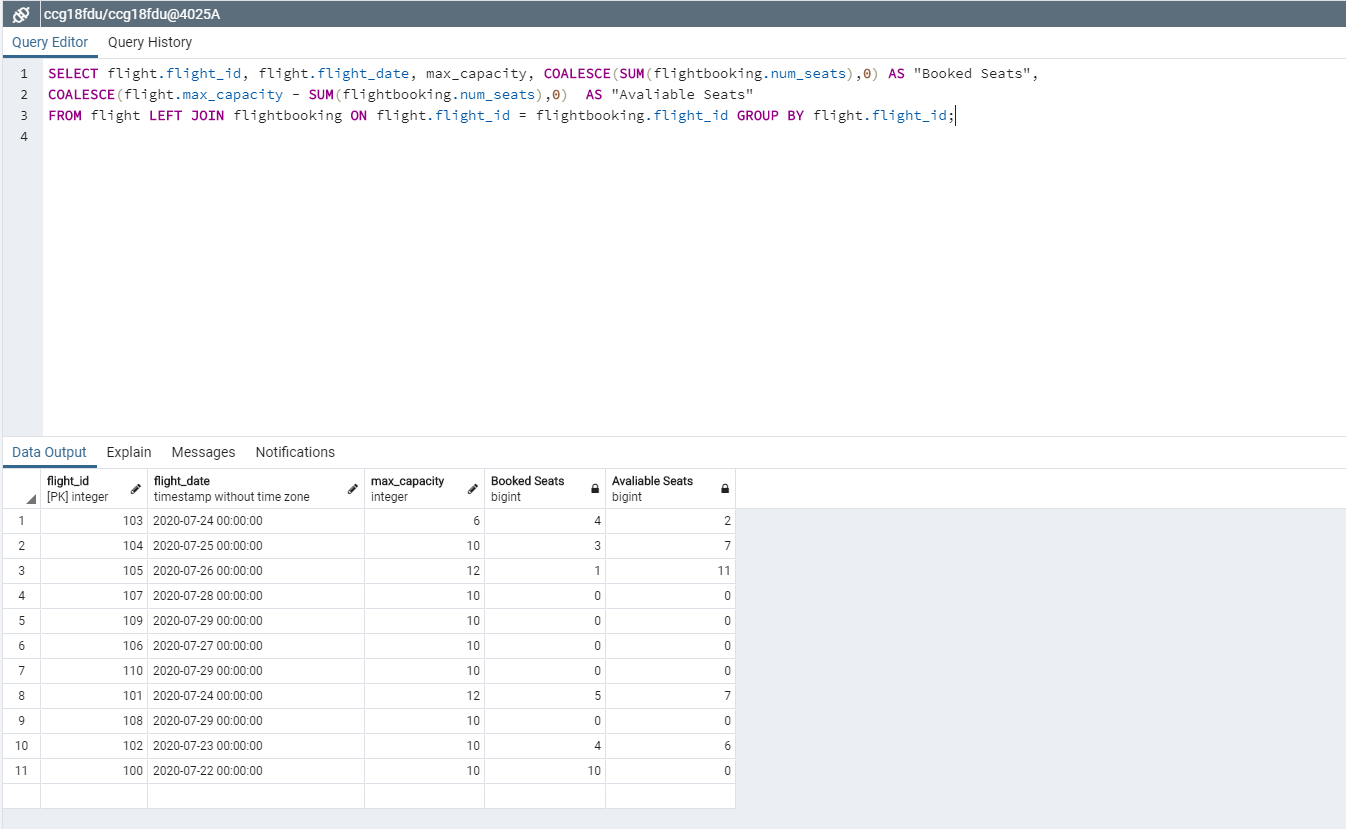
**Testing task 3**

--------------------------------------------------------------------------------------------------------------------------------------------------

**3. Check the availability of seats on all flights by showing the flight ID number, flight date along with the**

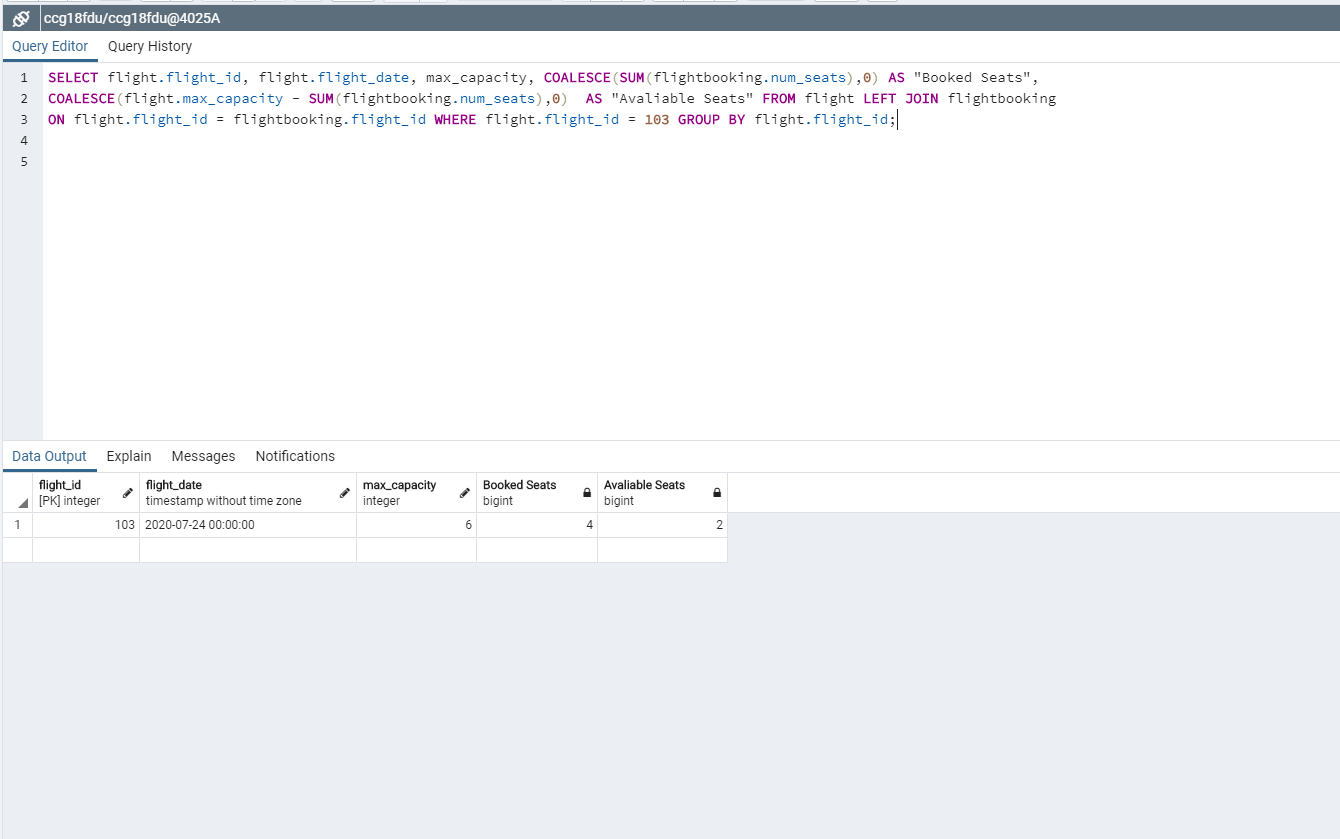
**number of booked seats, number of available seats and maximum capacity.**

1. Check availability on **ALL** flights.



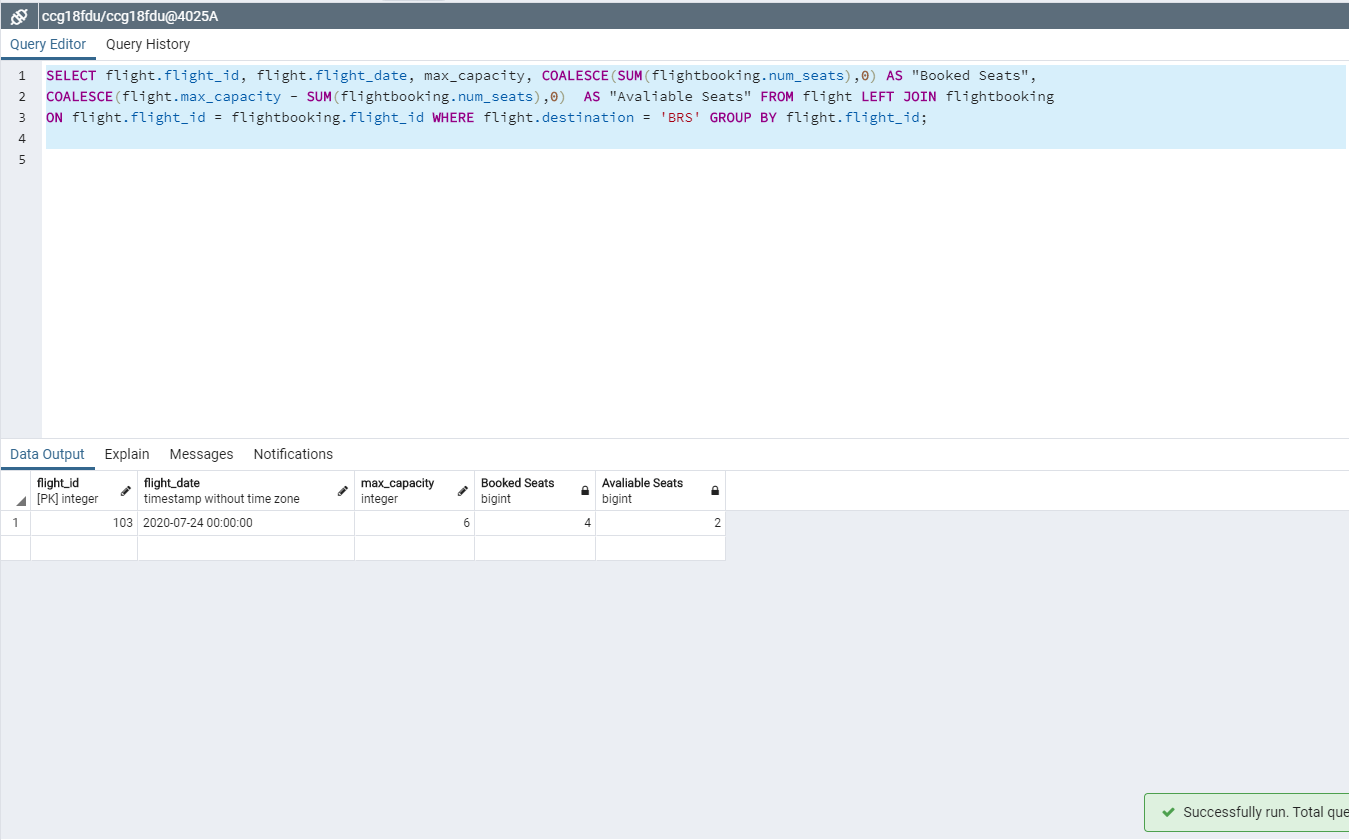
Marks: /6

1. Check availability on a specific flight with **flightID = 103.**

****

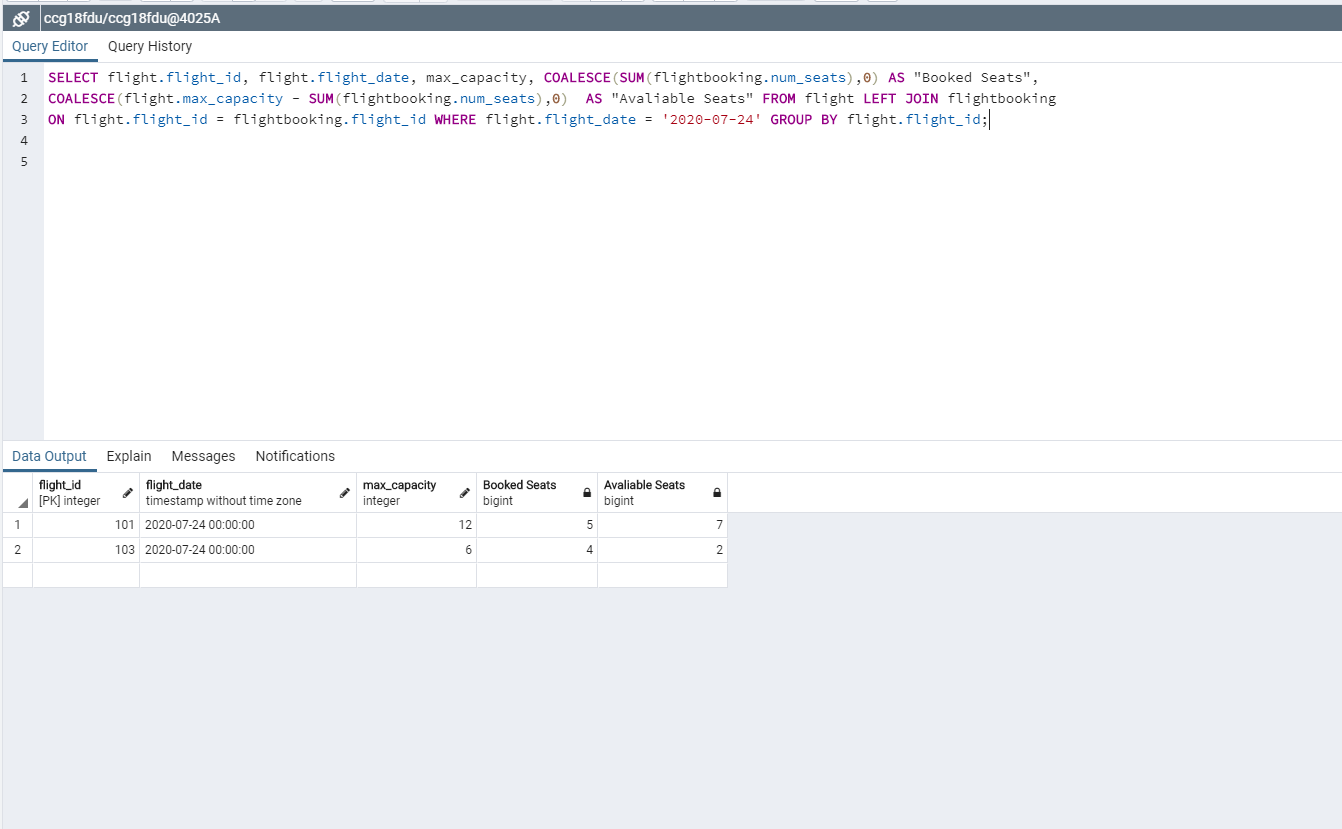
Marks: /1

1. Check availability on specific flights with **destination = ‘BRS’.**



Marks: /1

1. Check availability on specific flights with **date = ‘2020-07-24’**



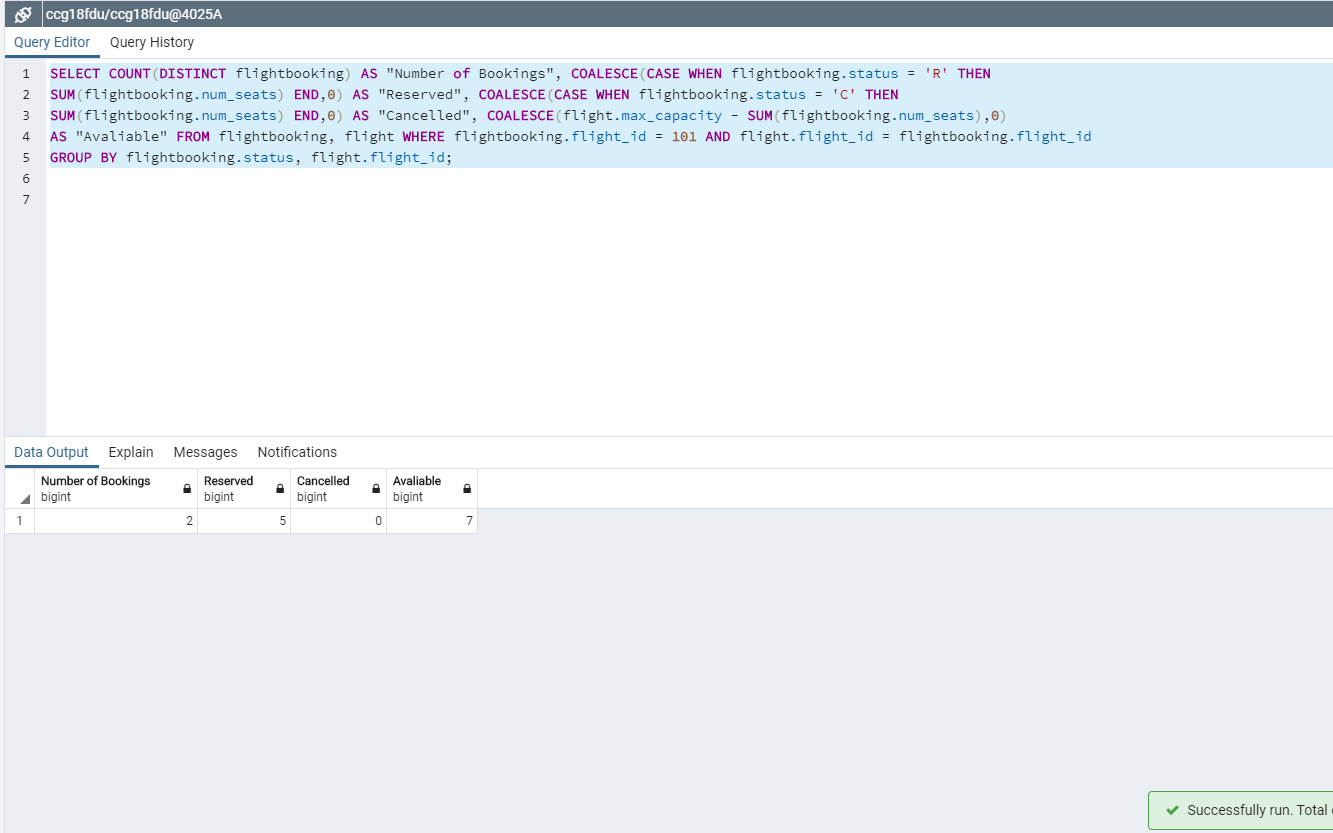
Marks: /1

--------------------------------------------------------------------------------------------------------------------------------------------------

**Testing task 4**

--------------------------------------------------------------------------------------------------------------------------------------------------

**4. Check status of all seats for flightID = 101.**

****

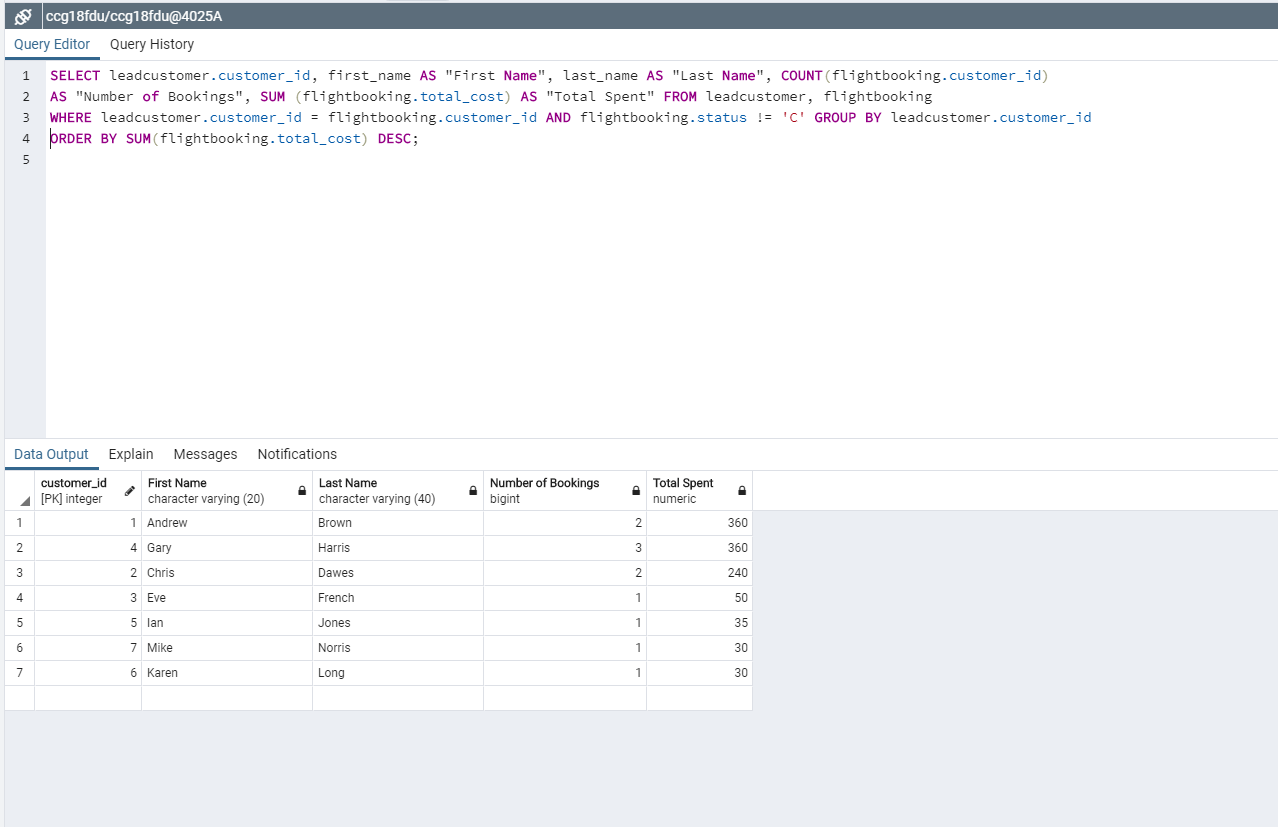
Marks: /4

--------------------------------------------------------------------------------------------------------------------------------------------------

**Testing task 5**

--------------------------------------------------------------------------------------------------------------------------------------------------

**5. Produce Ranked list of lead customers by total spend.**

****

Marks: /5

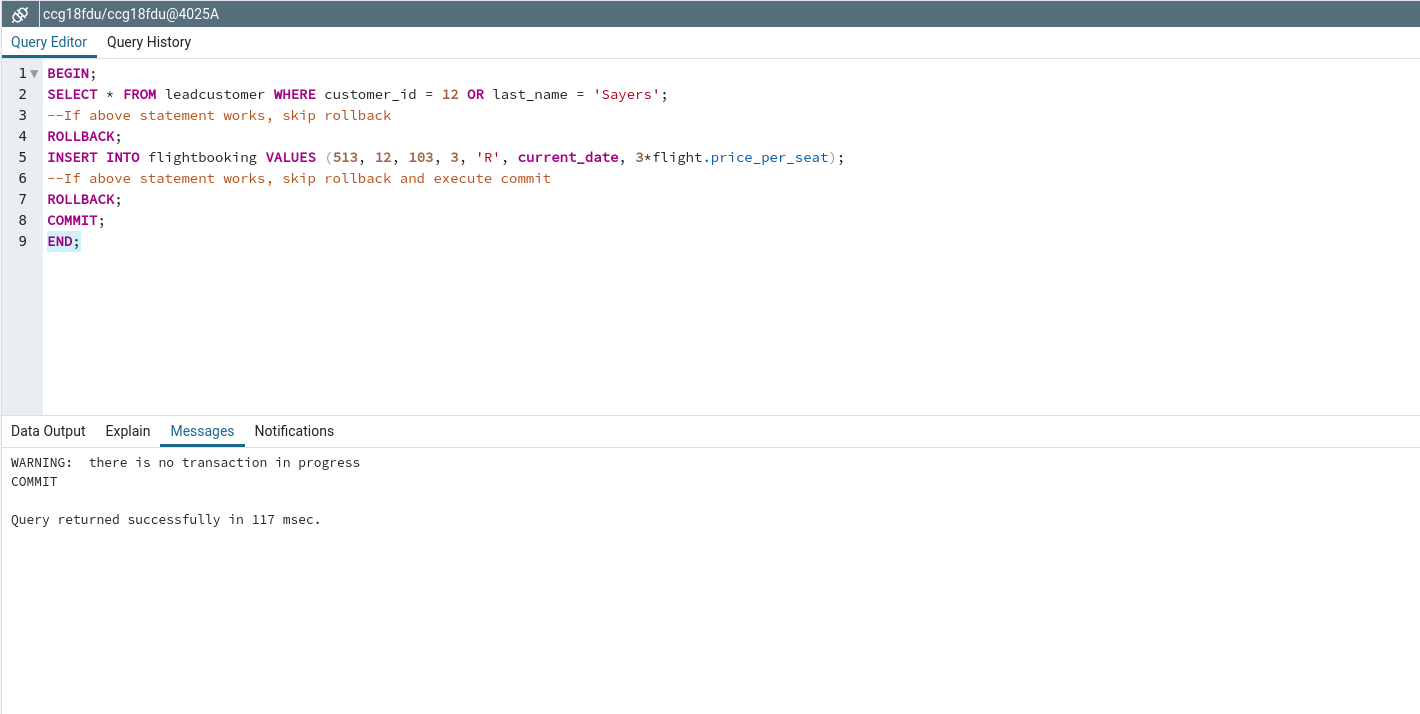
--------------------------------------------------------------------------------------------------------------------------------------------------

**Testing task 6**

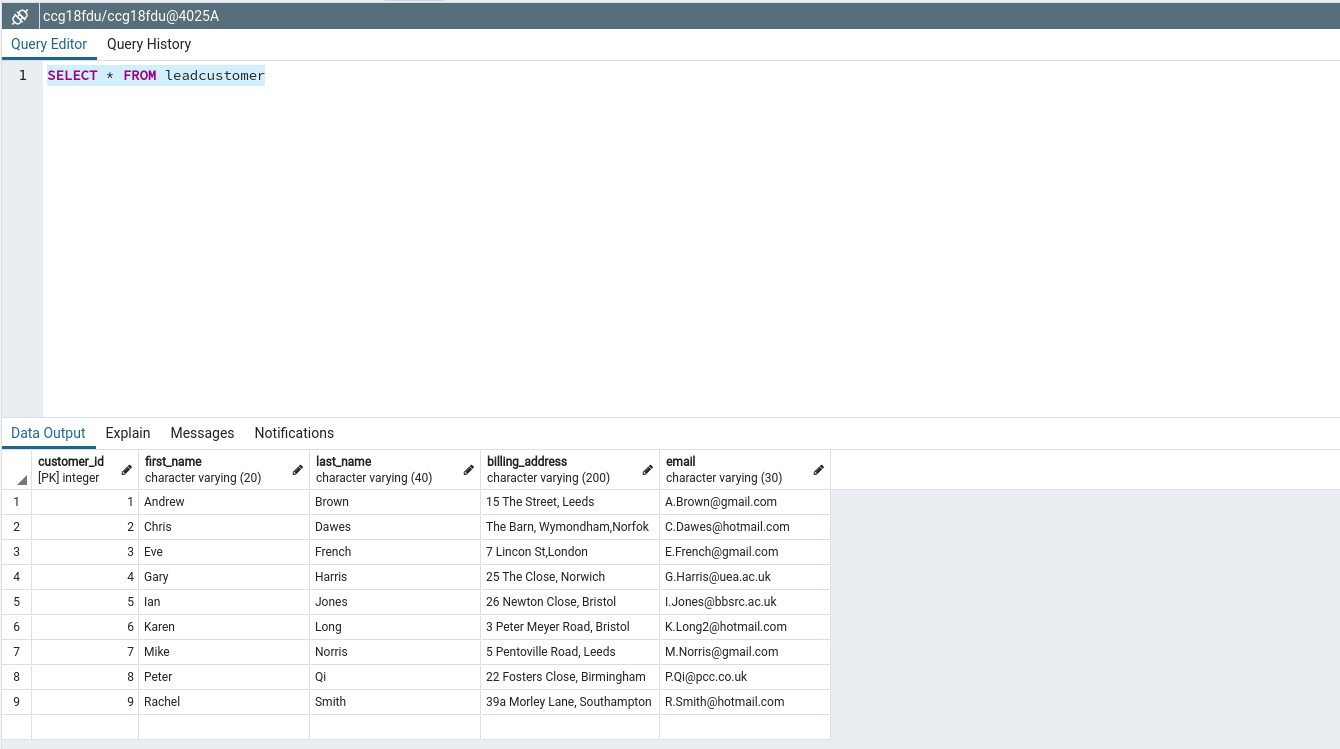
--------------------------------------------------------------------------------------------------------------------------------------------------

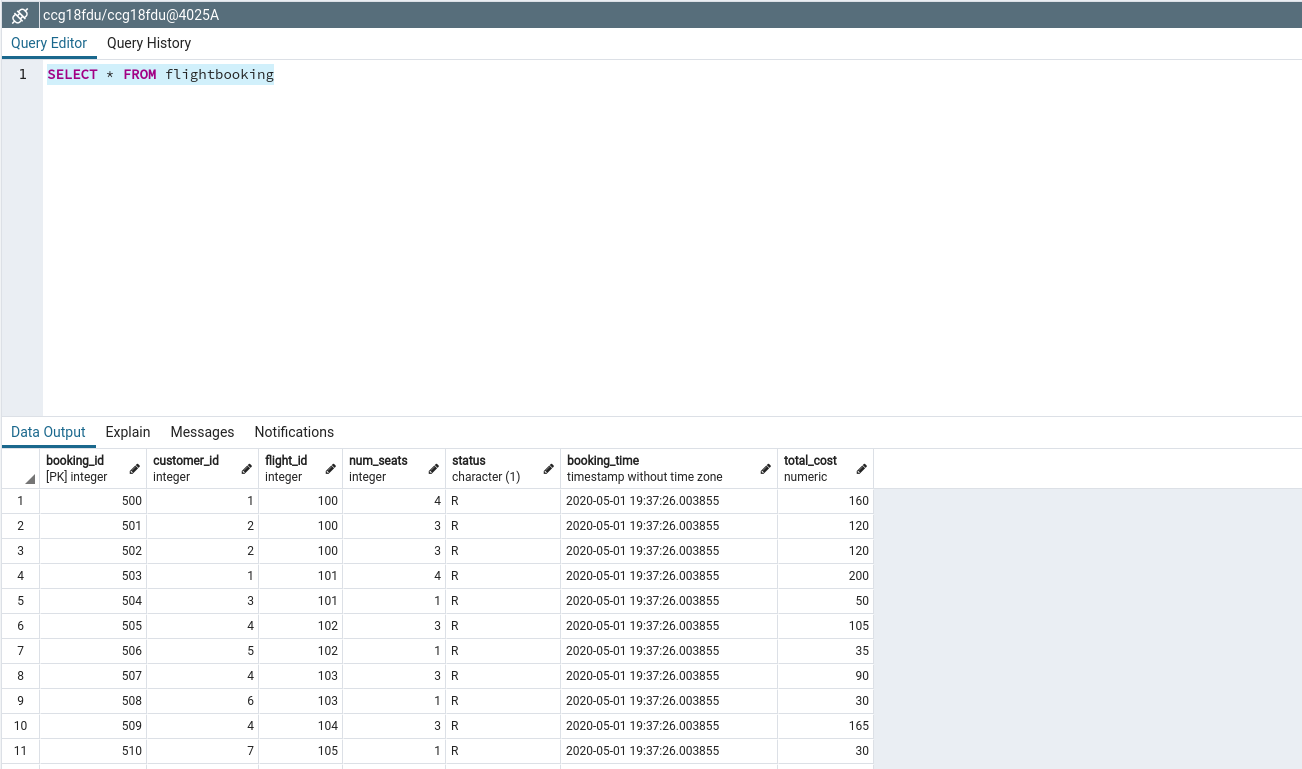
**6. Insert a flight booking.**

1. Check if a customer exists as part of inserting a new flight booking. Check for lead customer with customerID= 12 or with Surname ‘Sayers’. Show results. If found, insert flight booking with details: BookingID = 513, customerID=12, FlightID=103 , NumSeats = 3 , Status = 'R'. If customer not found, then the process of inserting a flight booking should be cancelled.



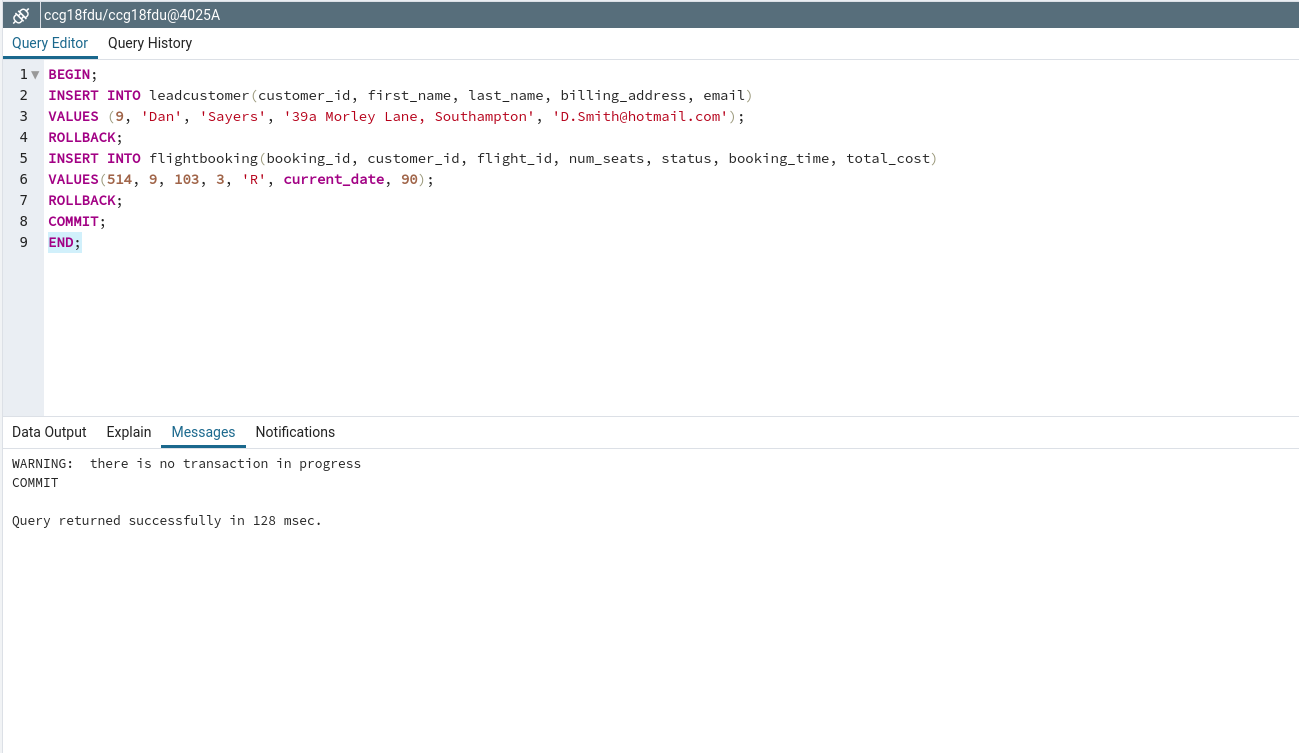
Marks: /4



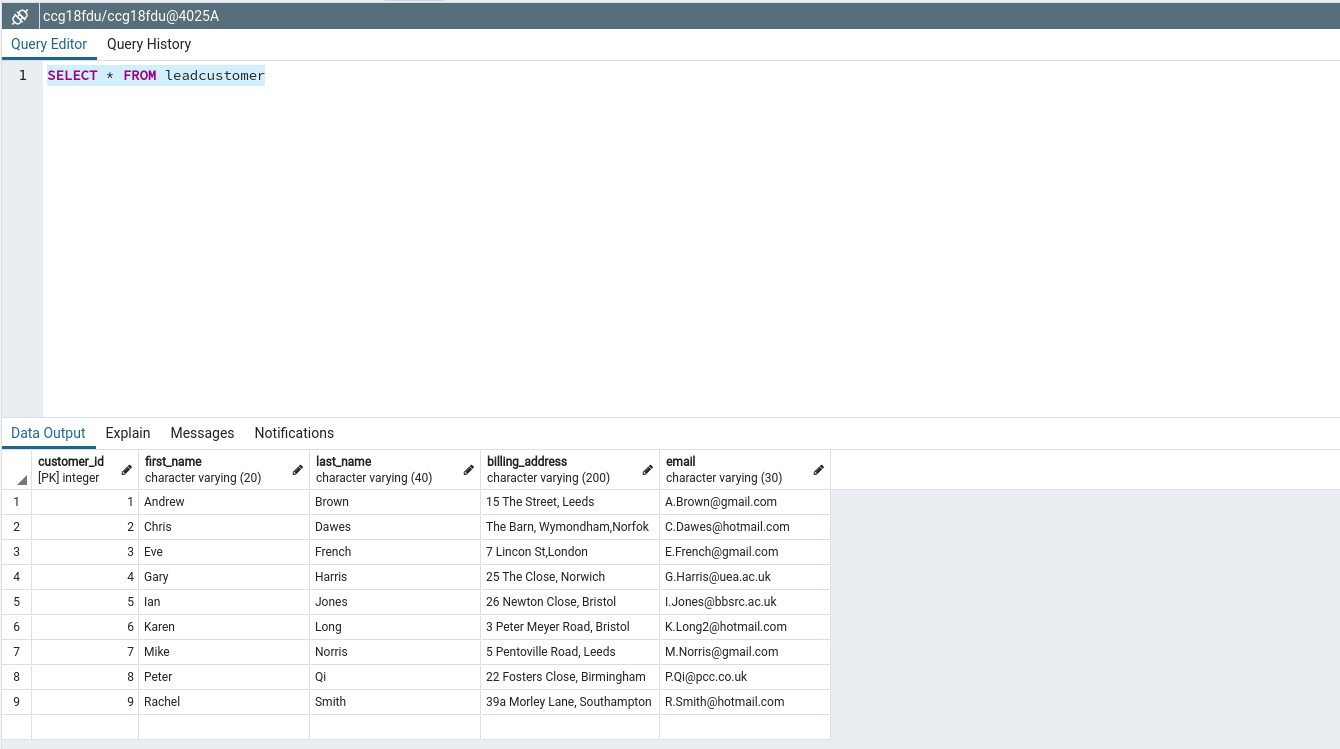


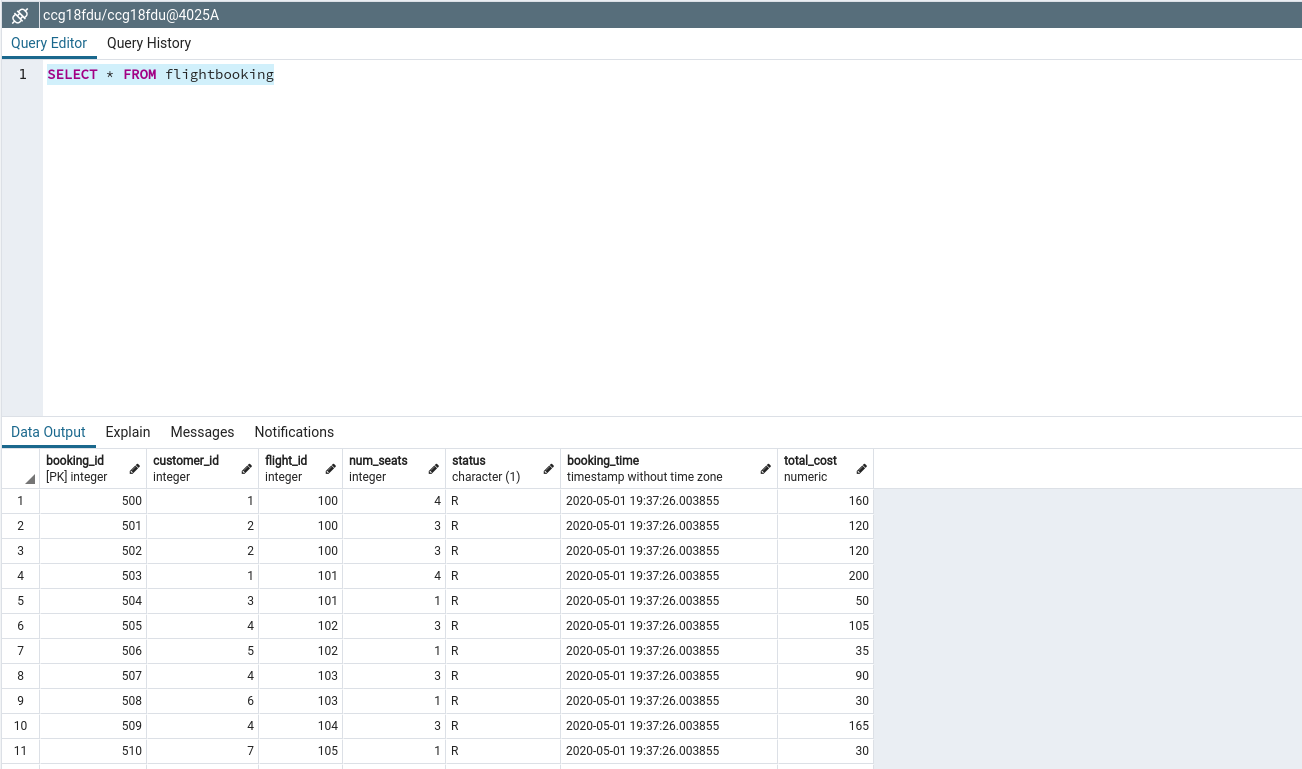
Marks: /2

1. Insert a new lead customer as part of inserting a new flight booking. The details of the lead customer are customerID= 9, FirstName = ‘Dan’ , Surname = ‘Sayers’, Billing Address = '39a Morley Lane, Southampton', email = 'D.Smith@hotmail.com'. If the lead customer is inserted correctly, insert flight booking with details: BookingID = 514, customerID=9 , FlightID=103 , NumSeats = 3 , Status = 'R'. If there are any problems, then the process of inserting a flight booking should be cancelled.



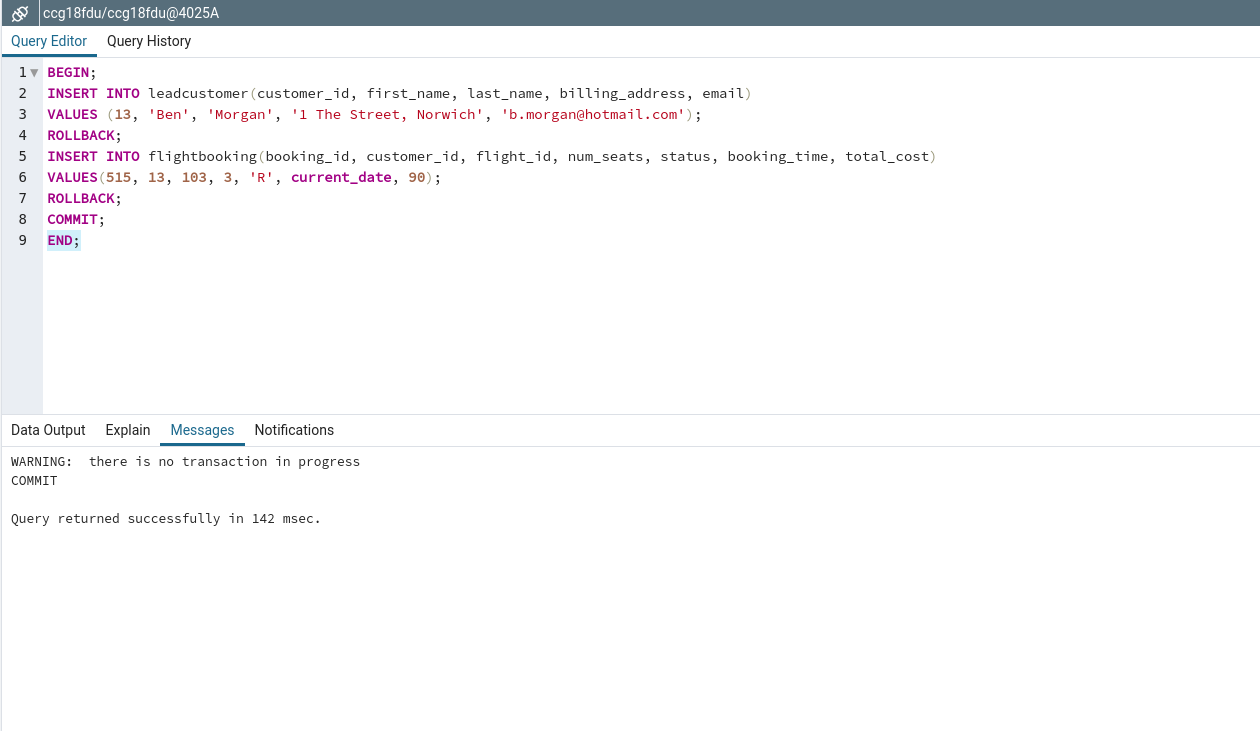
Marks: /3



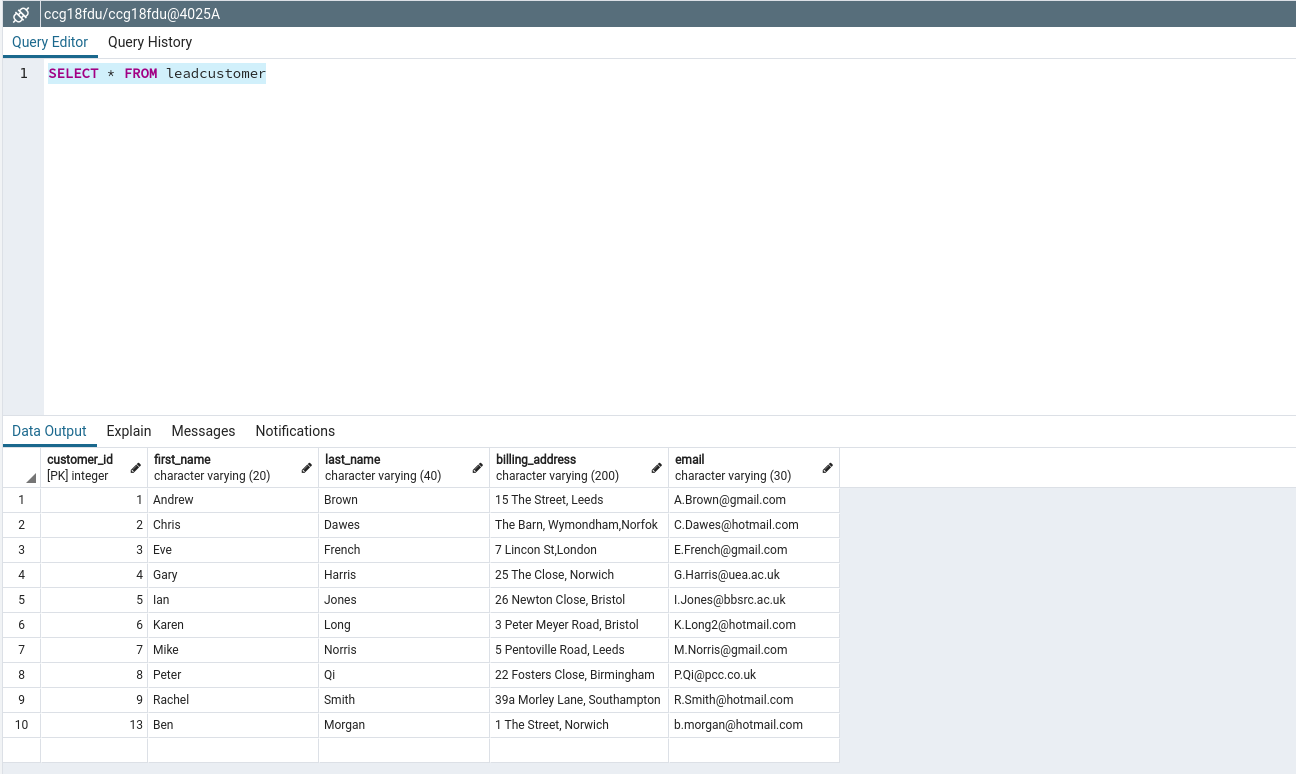


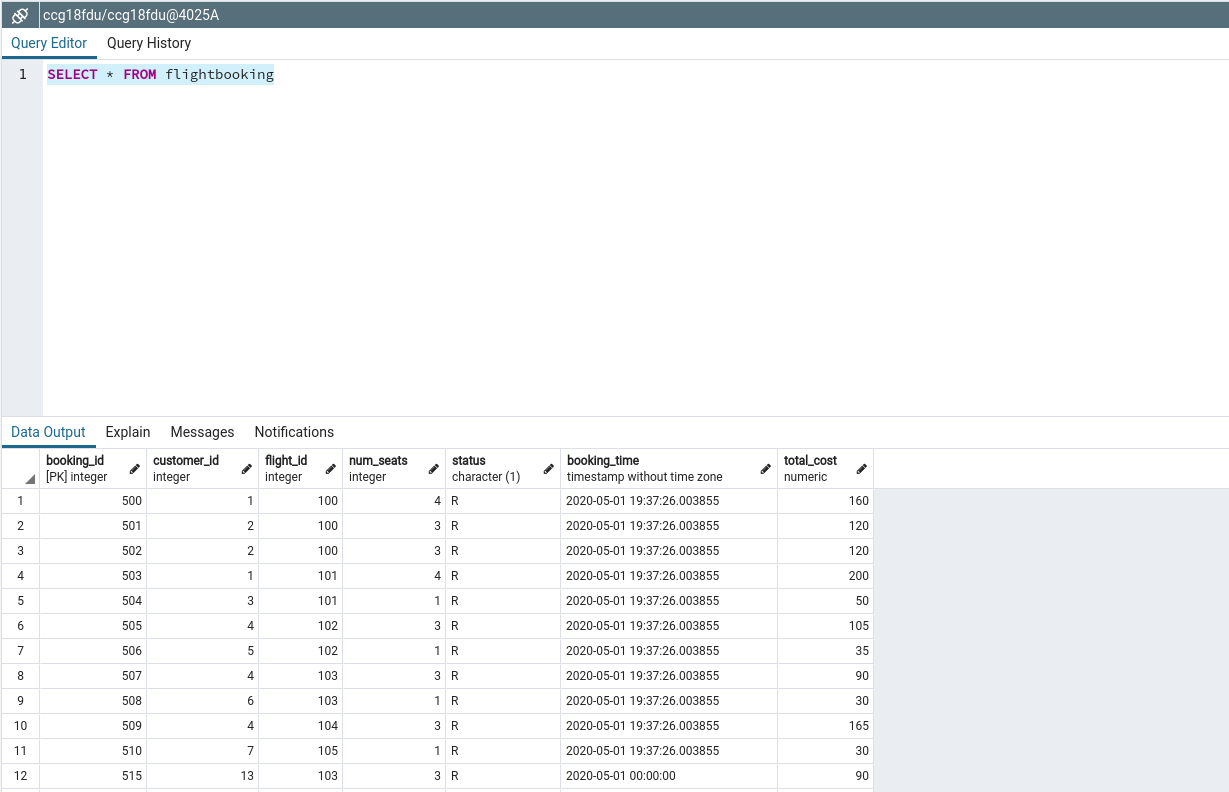
Marks: /2

1. Insert a new lead customer as part of inserting a new flight booking. The details of the lead customer are customerID= 13, FirstName = ‘Ben’ , Surname = ‘Morgan’, Billing Address = '1 The Street, Norwich, email = 'b.morgan@hotmail.com'. If the lead customer is inserted correctly, insert flight booking with details: BookingID = 515, customerID=13, FlightID=103 , NumSeats = 3 , Status = 'R'. If there are any problems, then the process of inserting a flight booking should be cancelled.



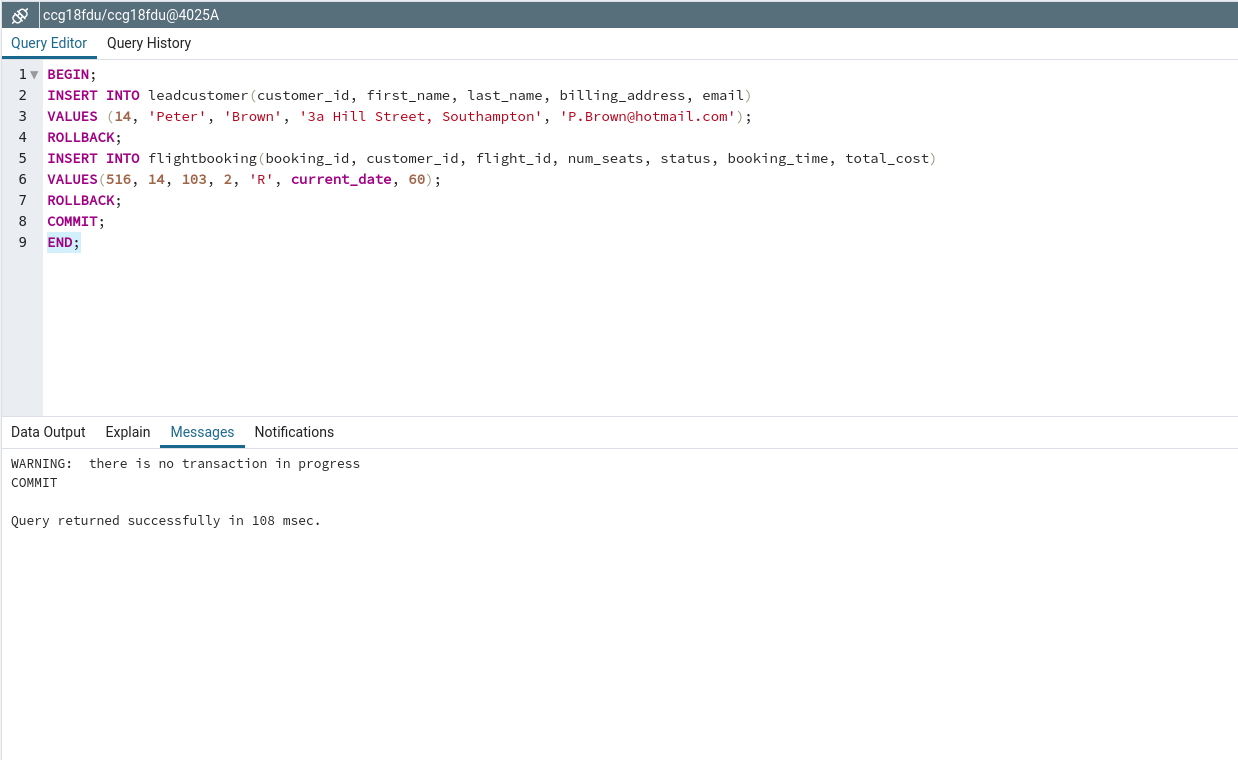
Marks: /4



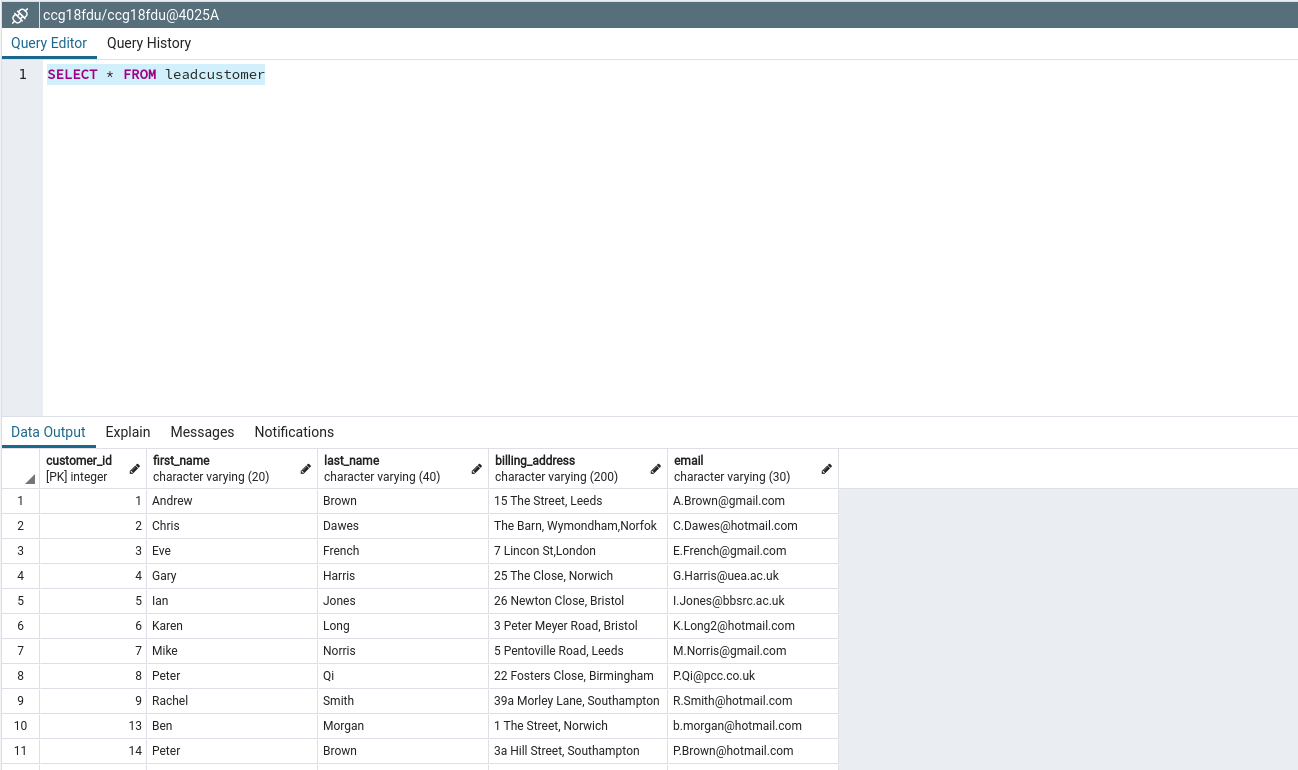


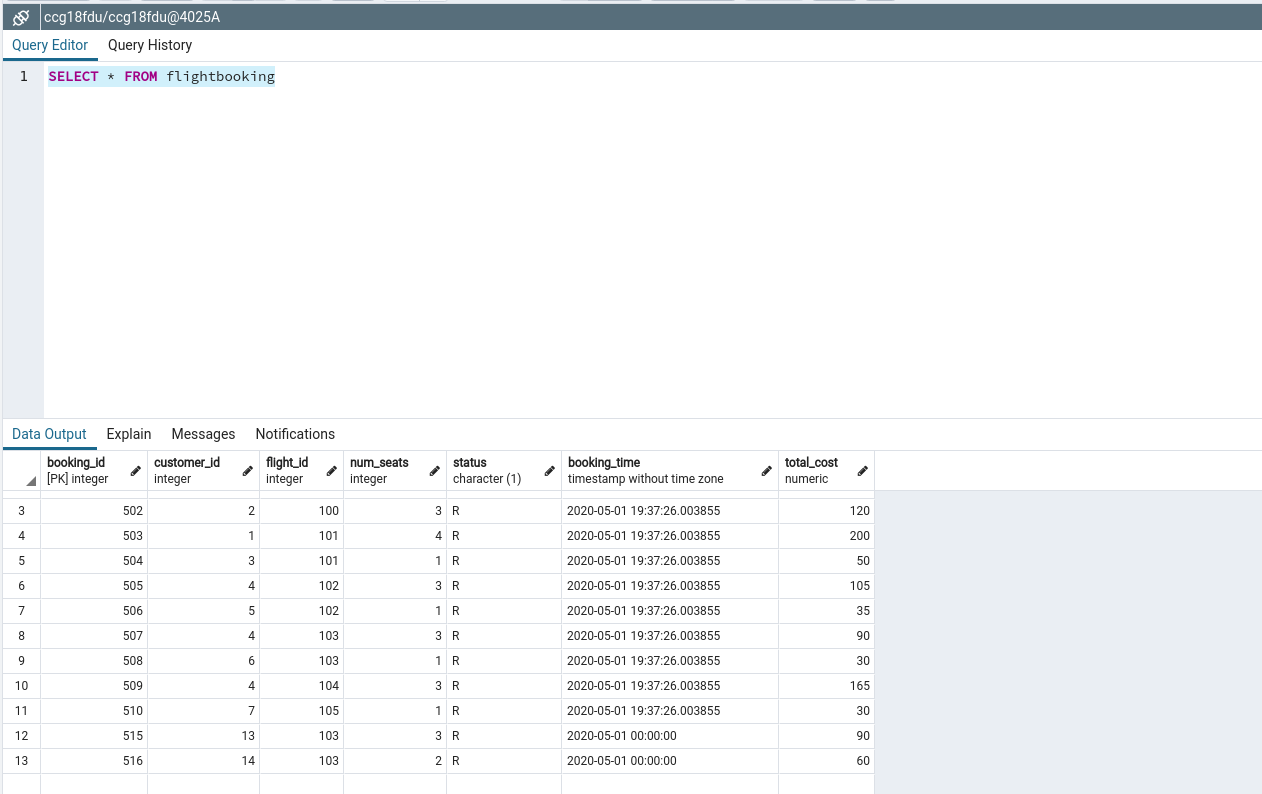
Marks: /2

1. Insert a new lead customer as part of inserting a new flight booking. The details of the lead customer are customerID= 14, FirstName = ‘Peter’ , Surname = ‘Brown’, Billing Address = '3a Hill Street, Southampton', email = ‘P.Brown@hotmail.com'. If the lead customer is inserted correctly, insert flight booking with details: BookingID = 516, customerID= 14, FlightID=103, NumSeats = 2 , Status = 'R'. If there are any problems, then the process of inserting a flight booking should be cancelled.



Marks: /4





Marks: /2

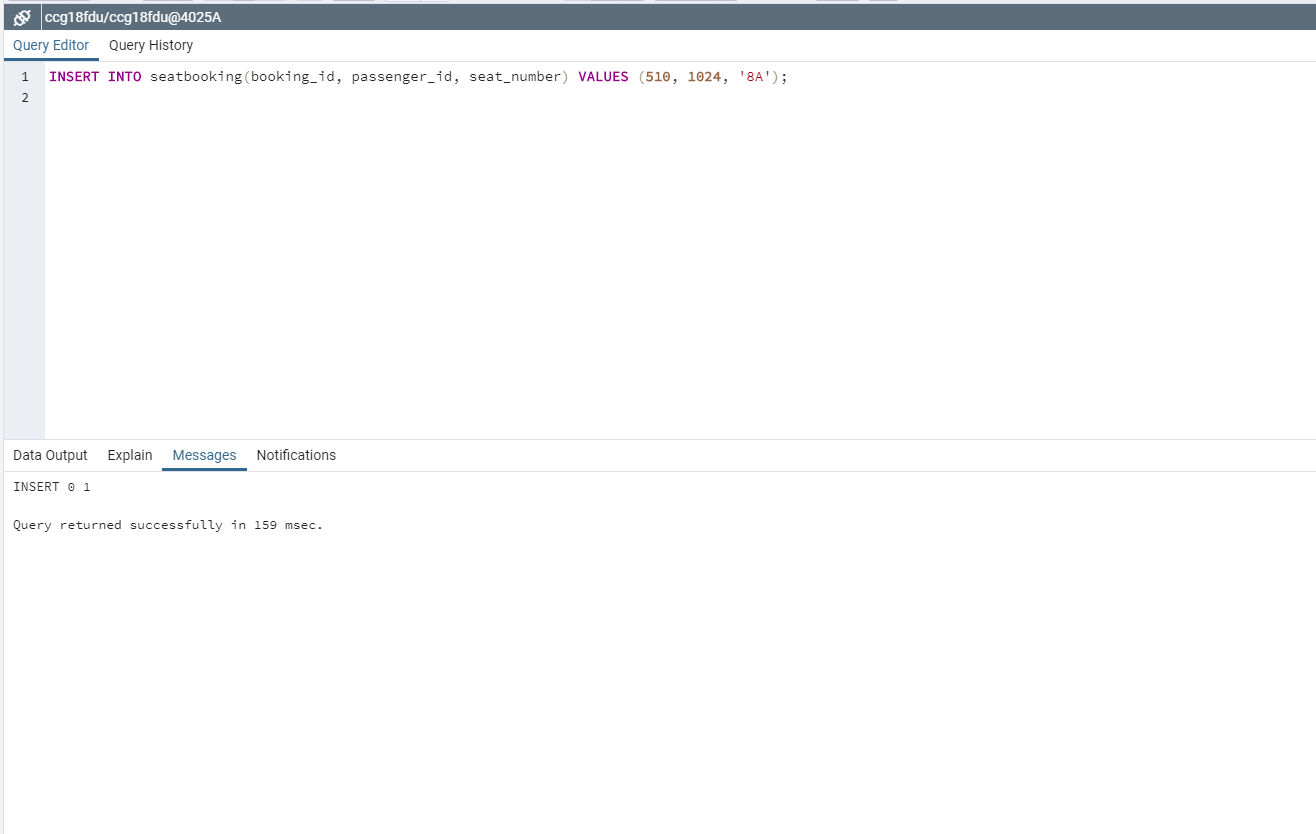
--------------------------------------------------------------------------------------------------------------------------------------------------

**Testing task 7**

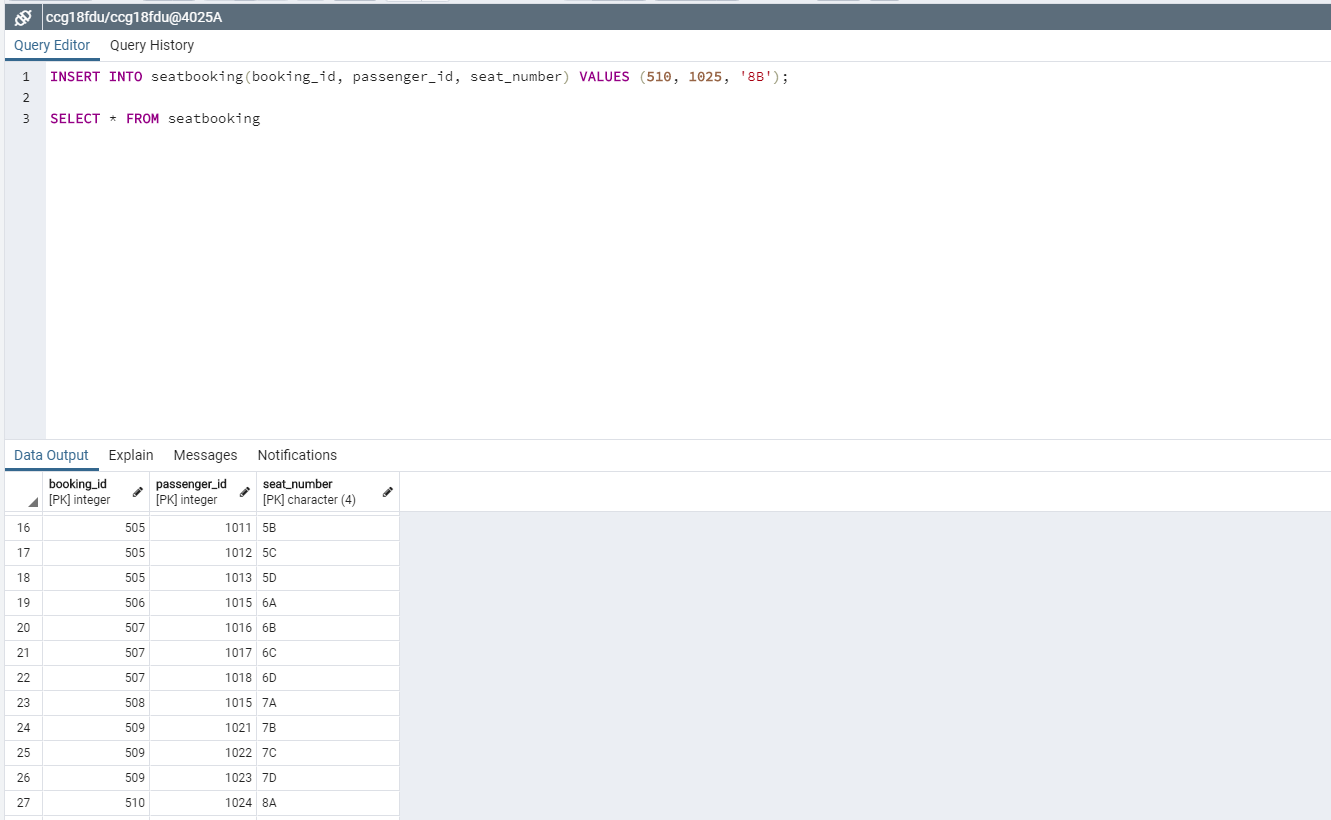
--------------------------------------------------------------------------------------------------------------------------------------------------

**7. Allocate seats to passengers**

1. Allocate seat for passenger 1024 in flight booking 510. The seat allocated is ‘8A’.

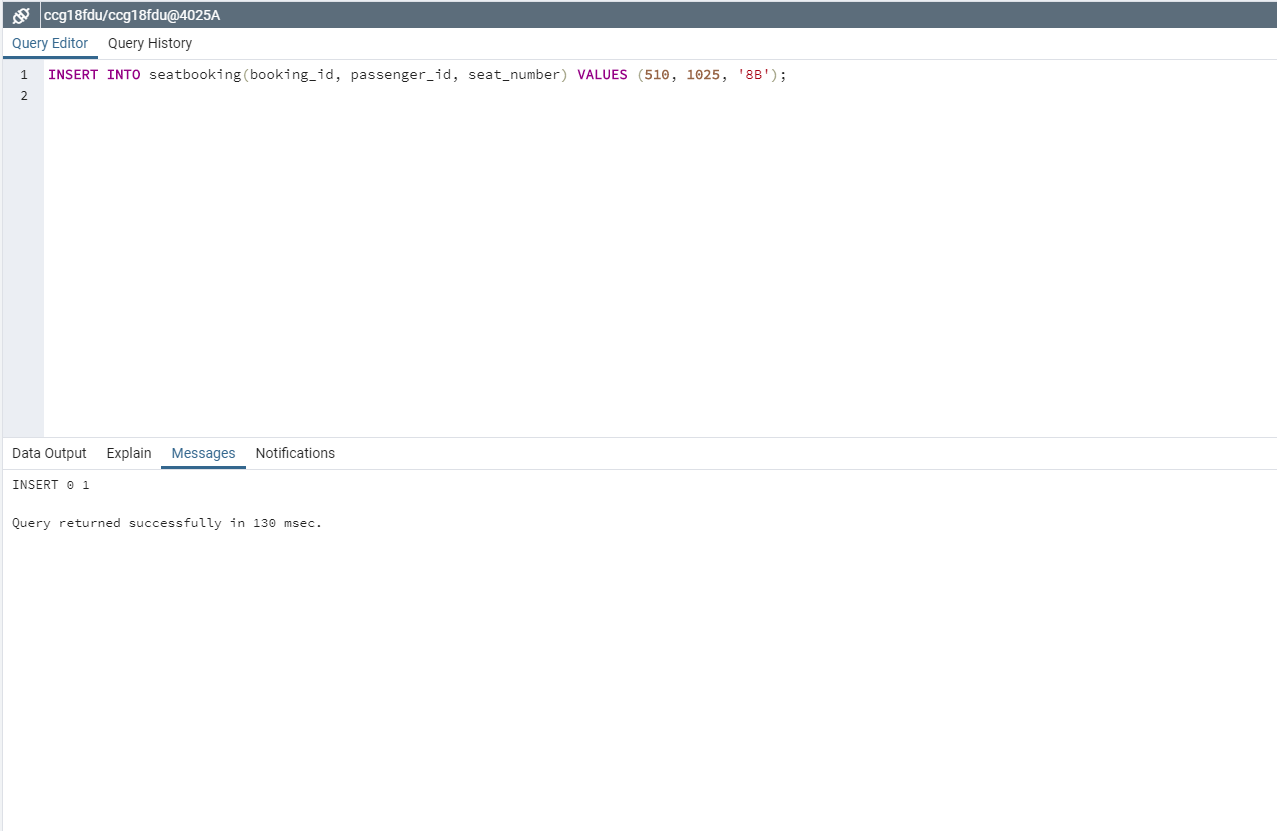


Marks: /1

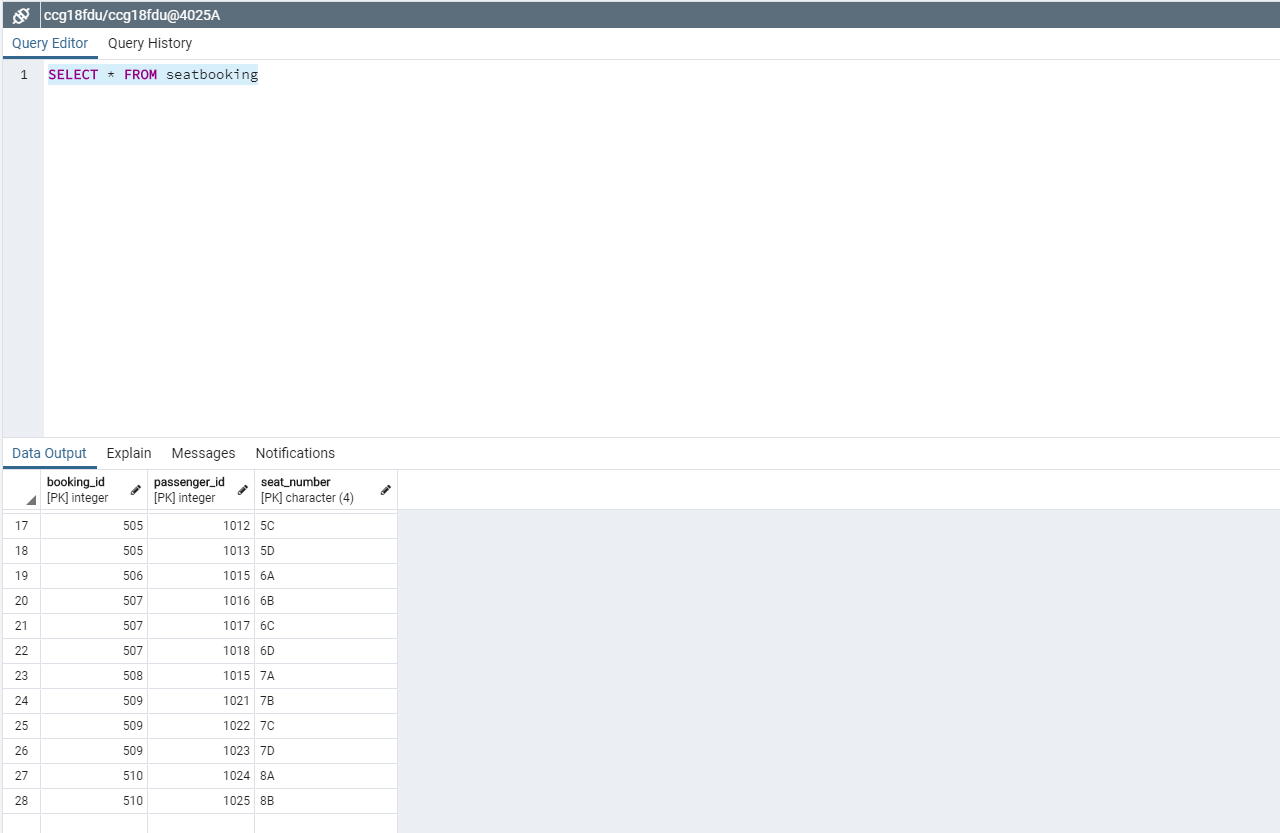
****

Marks: /1

1. Allocate seat for passenger 1025 in flight booking 510. The seat allocated is ‘8B’.



Marks: /1



Marks: /1

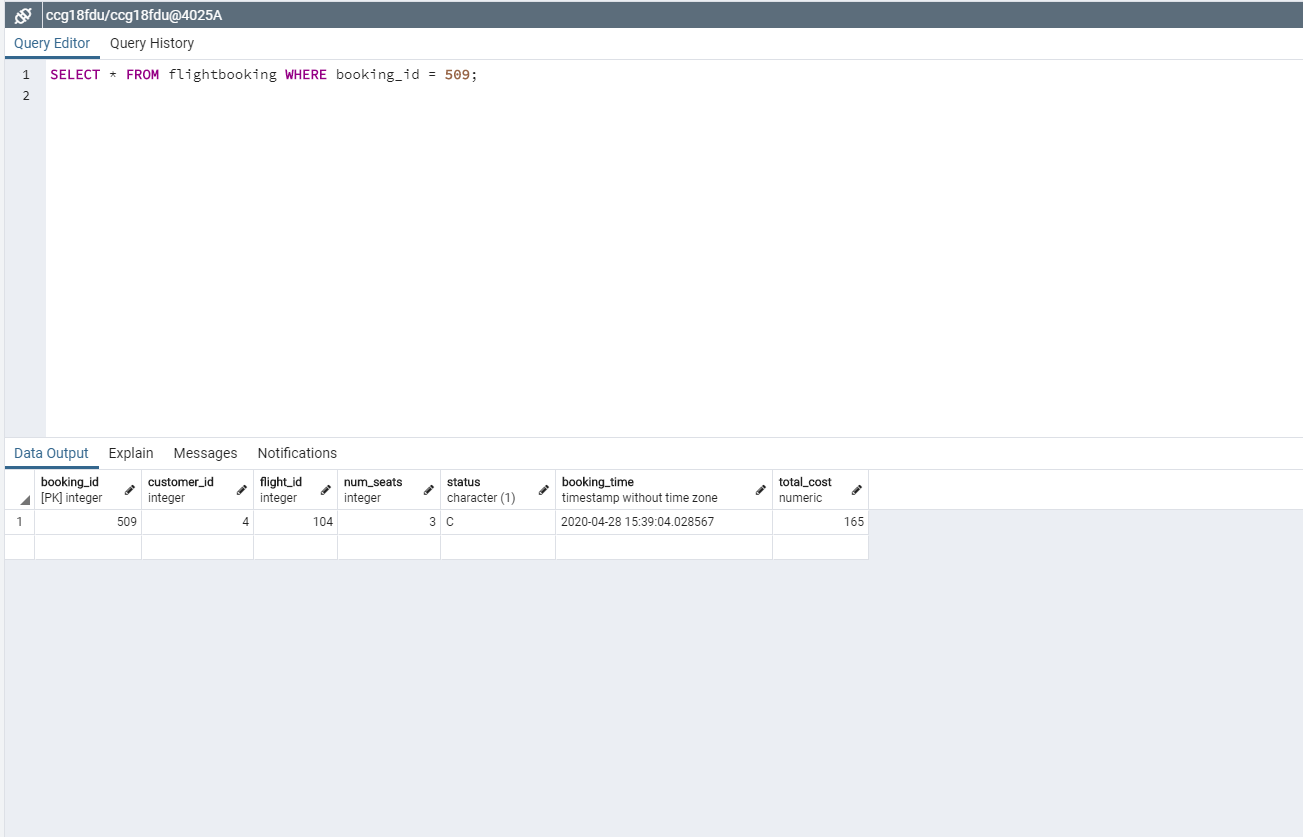
--------------------------------------------------------------------------------------------------------------------------------------------------

**Testing task 8**

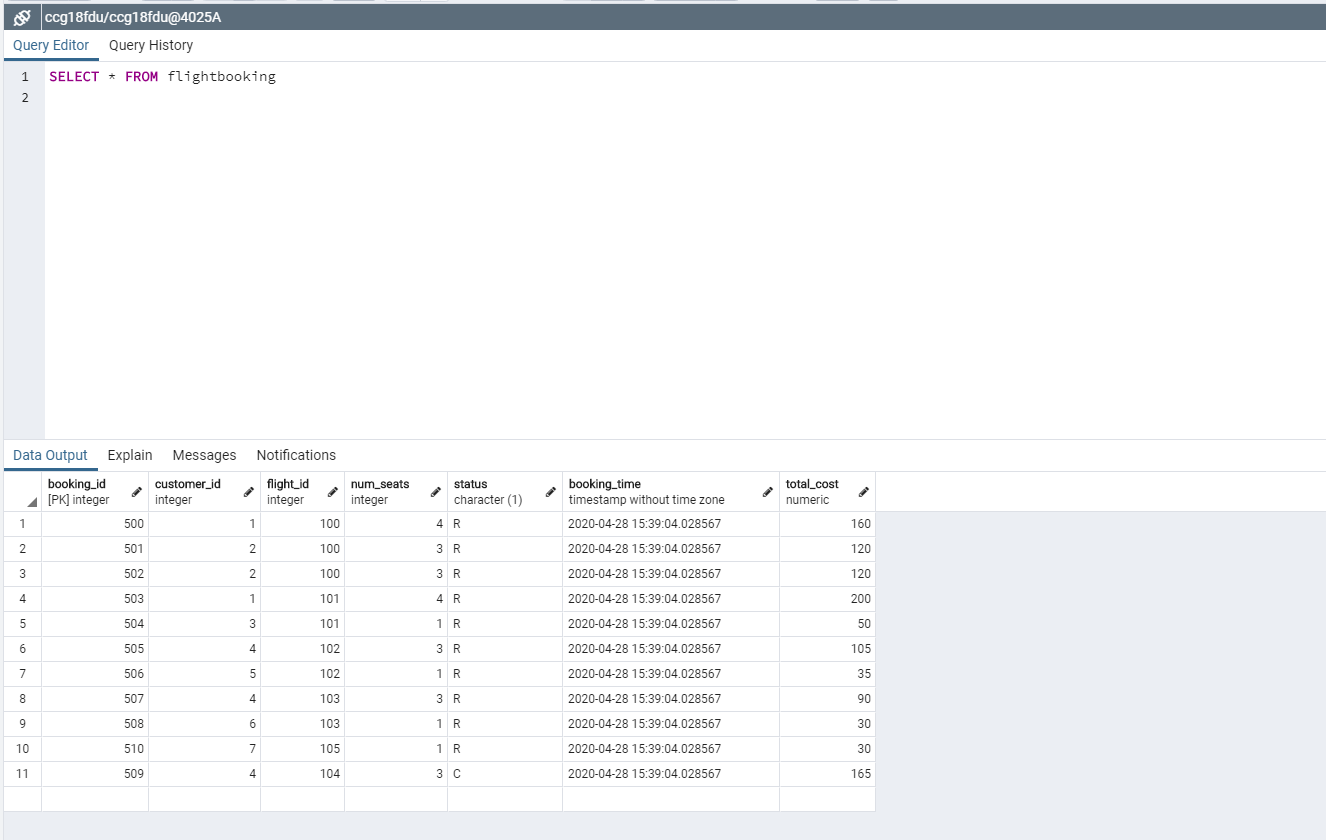
--------------------------------------------------------------------------------------------------------------------------------------------------

**8. Given bookingID cancel booking for BookingID= 509.**

1. Select from flight booking table to show it has been cancelled.

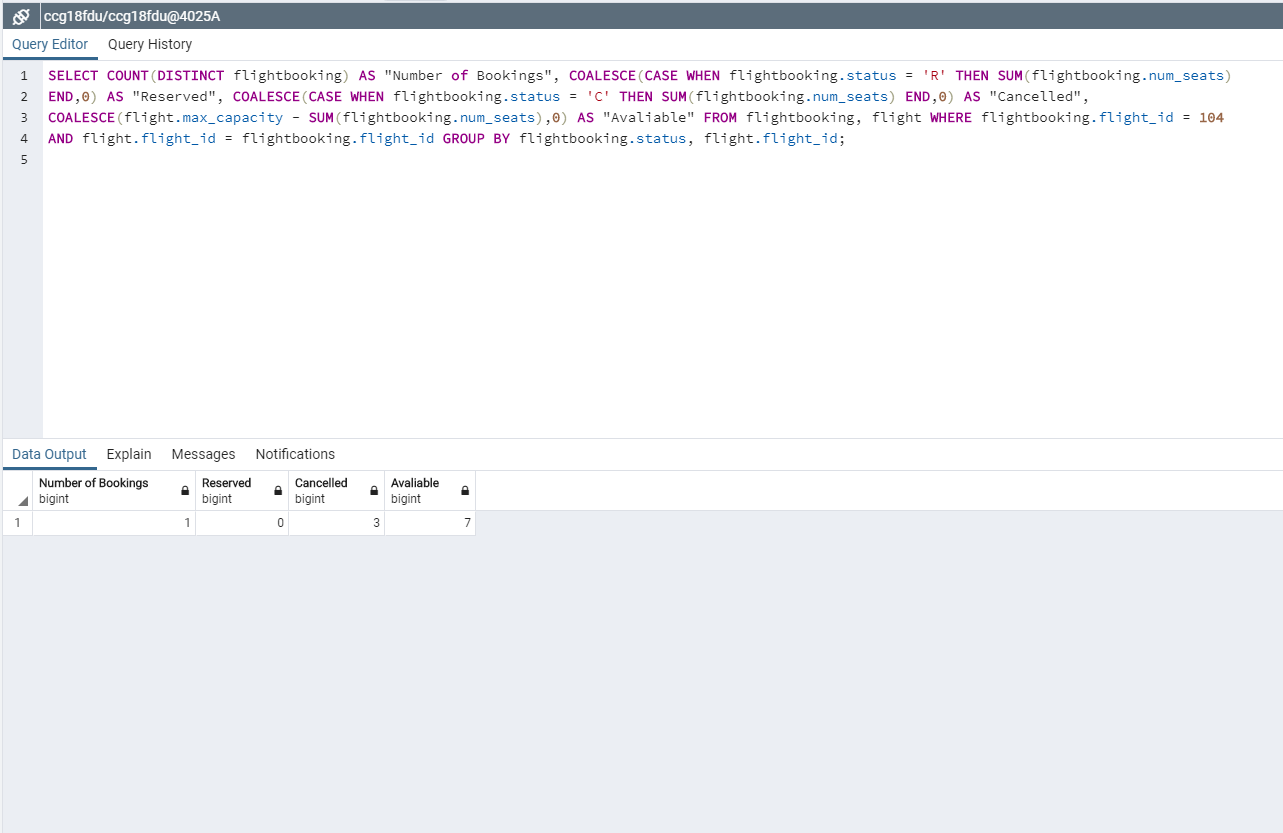


Marks: /2



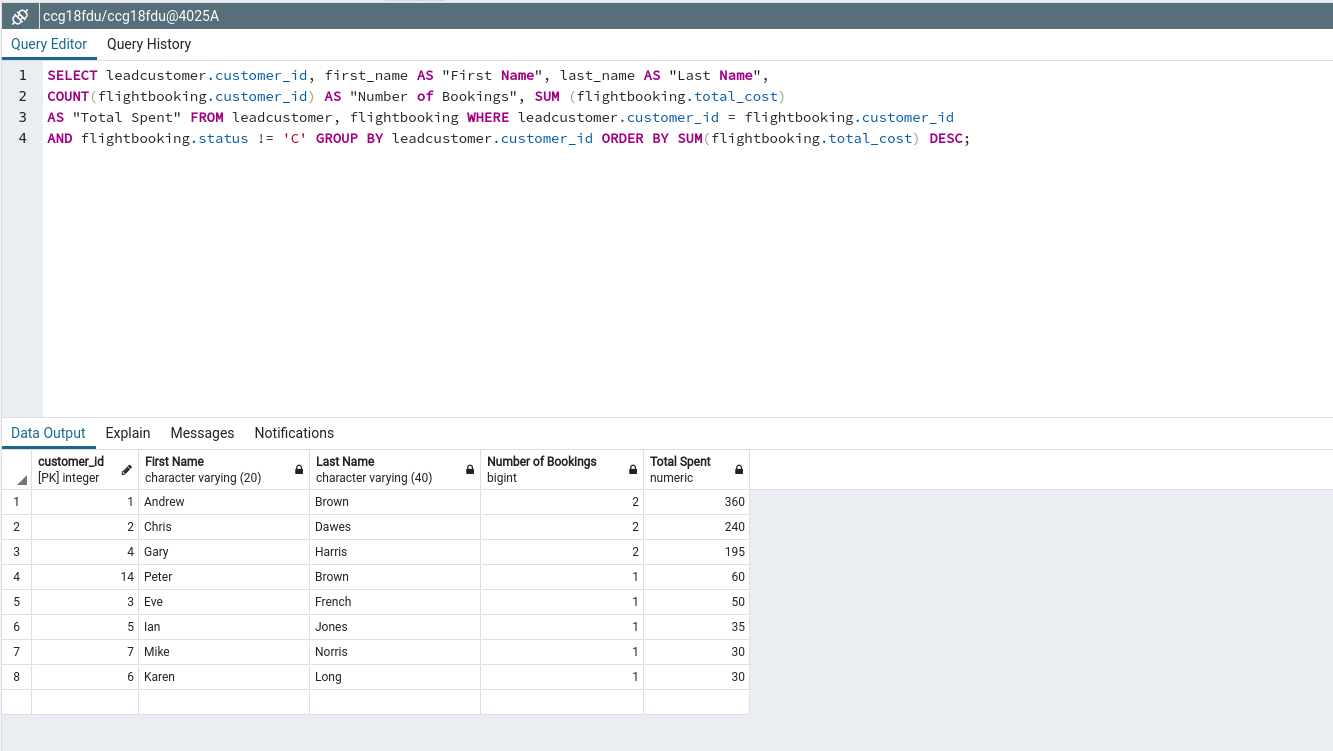
Marks: /1

1. Rerun task 4 (status of flights) for FlightID=104 as that has been affected.



Marks: /1

1. Rerun task 5 (ranked list of lead customers) to see what has changed.



Marks: /1

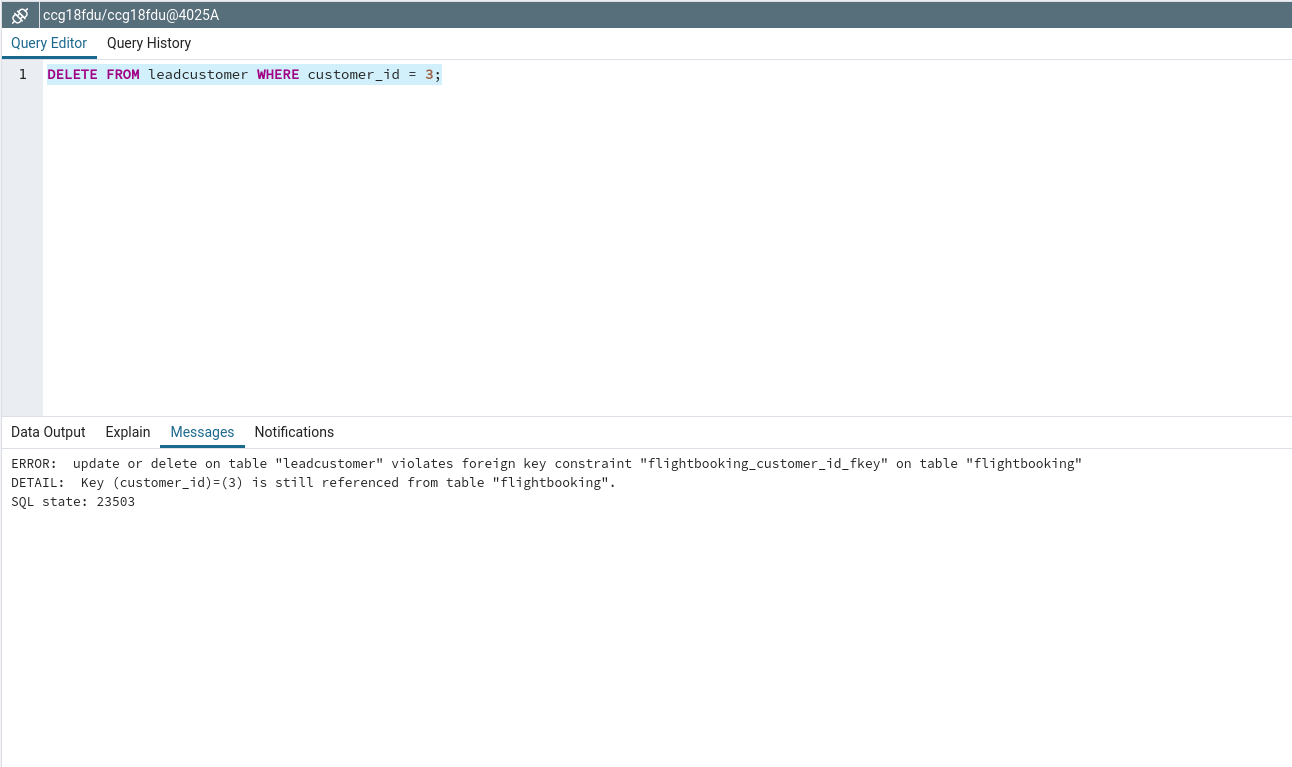
--------------------------------------------------------------------------------------------------------------------------------------------------

**Testing task 2**

--------------------------------------------------------------------------------------------------------------------------------------------------

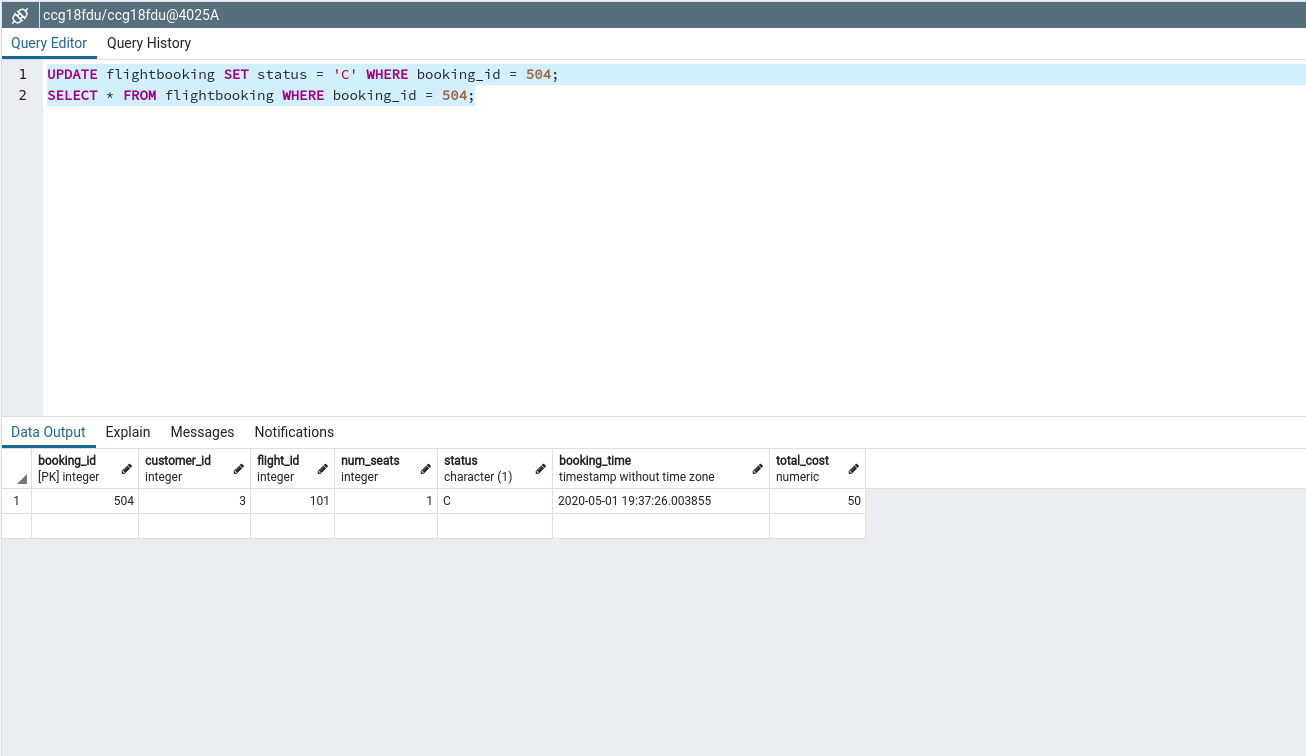
**2. Delete a Lead Customer.**

1. Delete lead customer with CustomerIDINSERT YOUR SQL QUERY AND OUTPUT HERE=3.



Marks: /1

1. Run task 8 (cancel a booking) for bookingID 504.

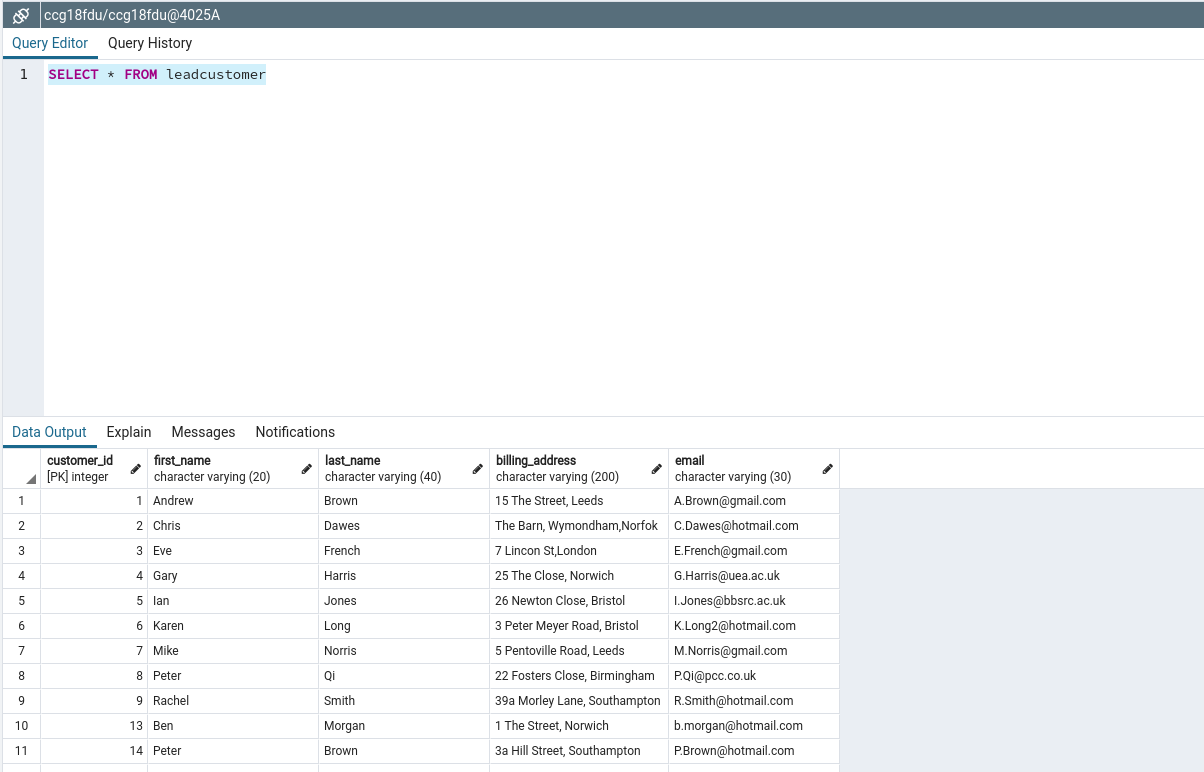


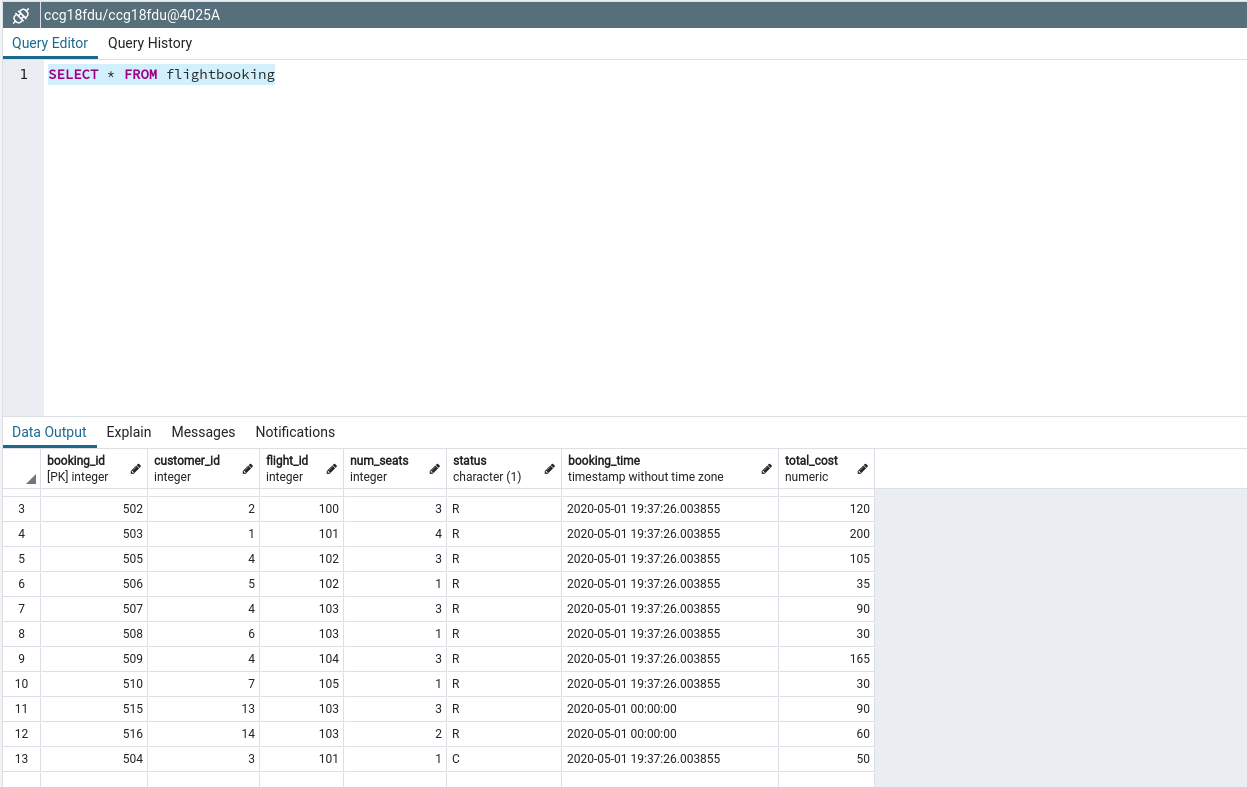
Marks: /1

1. Delete lead customer with CustomerID=3.



Marks: /1





Marks: /2

--------------------------------------------------------------------------------------------------------------------------------------------------

**END OF TESTING**

--------------------------------------------------------------------------------------------------------------------------------------------------

**APPENDIX:**  EXAMPLE OF HOW TO FILL THIS FORM/DOCUMENT

--------------------------------------------------------------------------------------------------------------------------------------------------

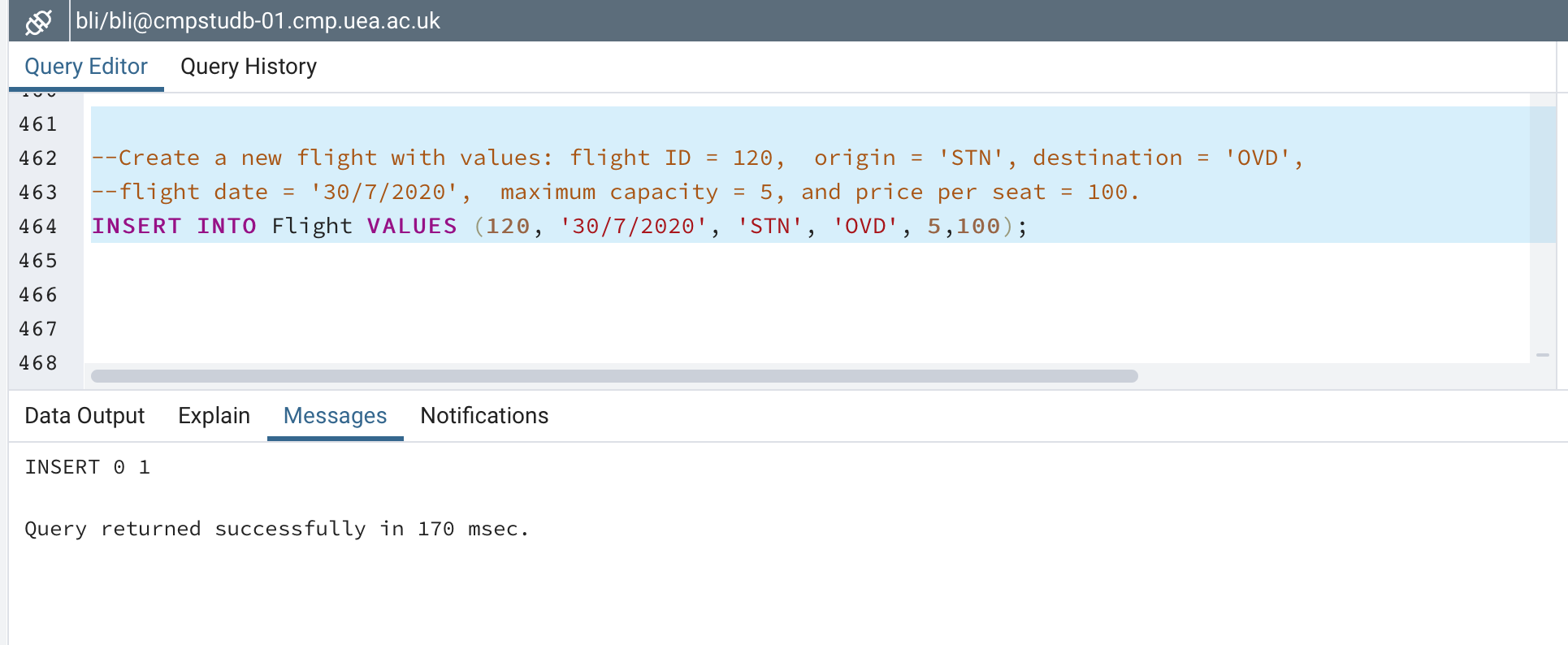
**Testing task 1**

--------------------------------------------------------------------------------------------------------------------------------------------------

**1. Given flight details create new flight record**

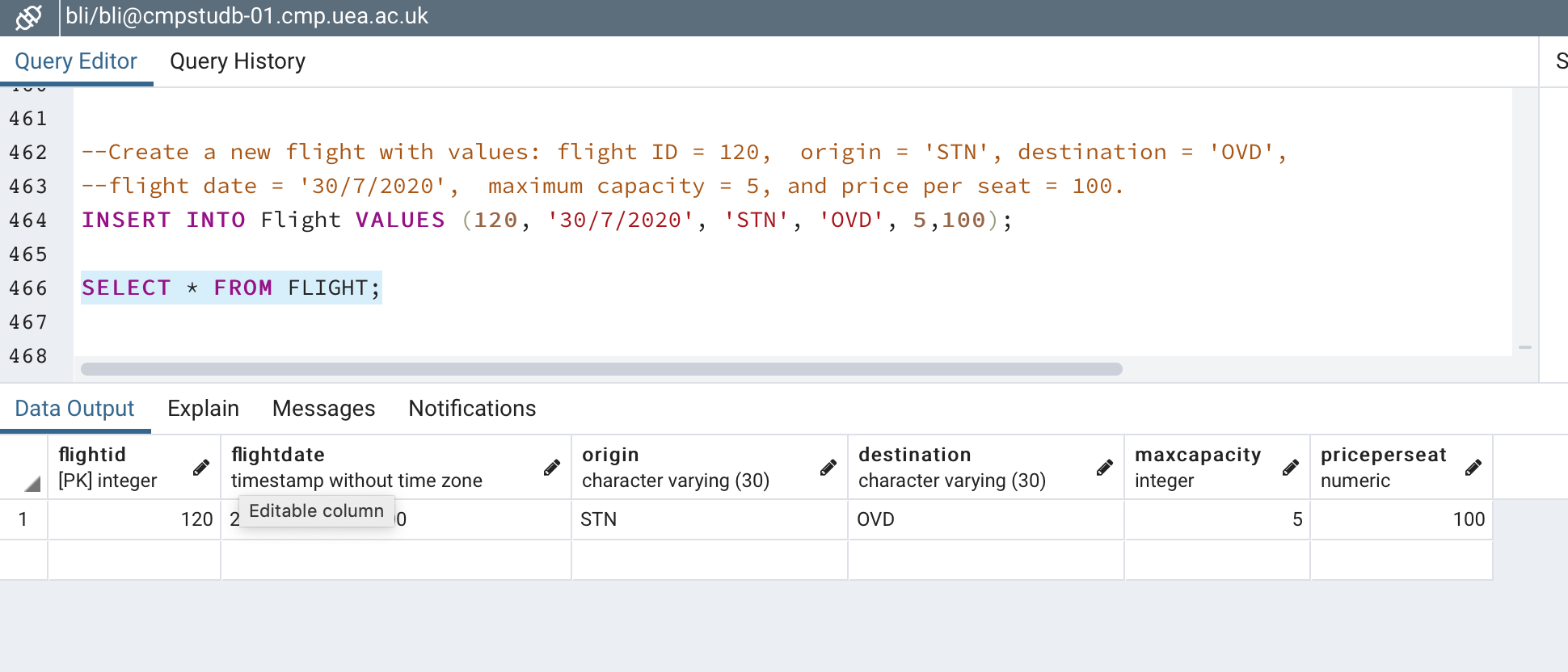
1. Create a new flight with values: flight ID = 120, origin = 'STN', destination = 'OVD', flight date = '30/7/2020', maximum capacity = 5, and price per seat = 100.

INSERT YOUR SQL QUERY AND OUTPUT HERE



Marks: /1

SHOW THE CONTENTS OF FLIGHT TABLE AT THE END OF THE TASK



Marks: /1

**\*\*\* Note: if you wish to submit evidence of running a Node.JS program, include it after the SQL output as an additional screenshot so show you achieved the task in Node.JS.\*\*\***