CS669

SiCheng Yi

Lab1

**Section One – Absolute Fundamentals**

1：Create the Movie table. As a reminder, make sure to follow along in the Lab 1 Explanations document as it shows you how to create tables and complete the other steps.

use lab1;

CREATE TABLE MovieTable(

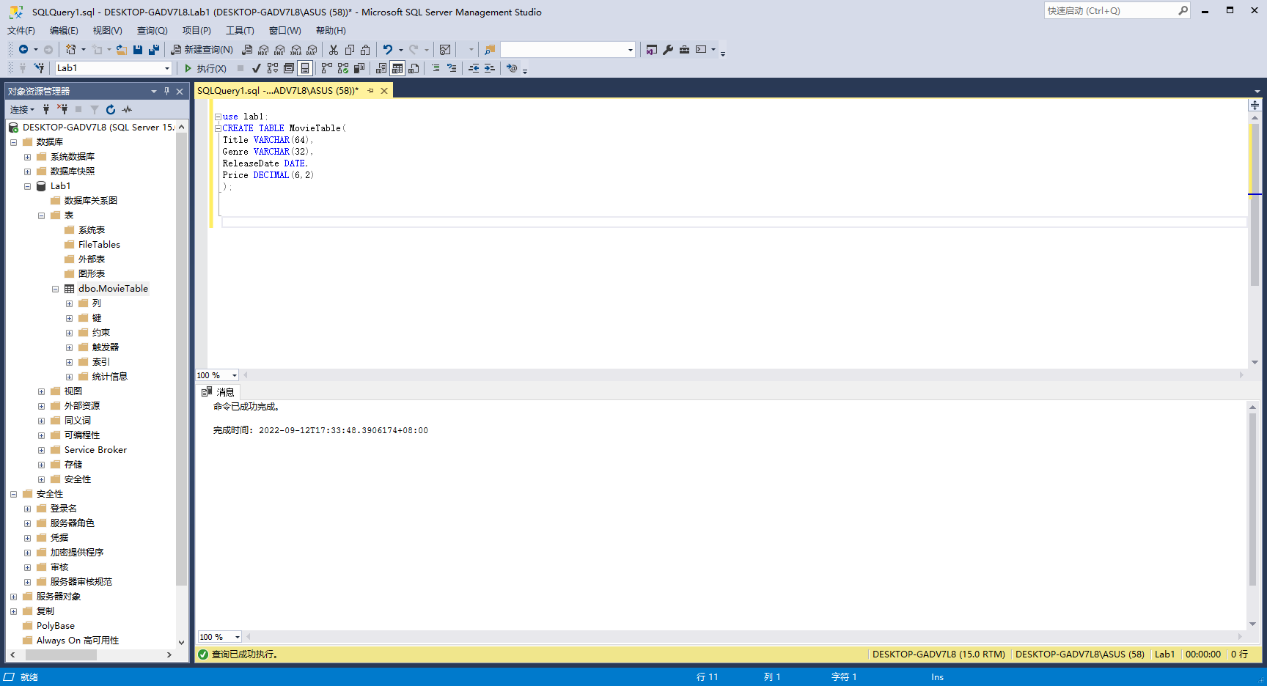
Title VARCHAR(64),

Genre VARCHAR(32),

ReleaseDate DATE,

Price DECIMAL(6,2)

);

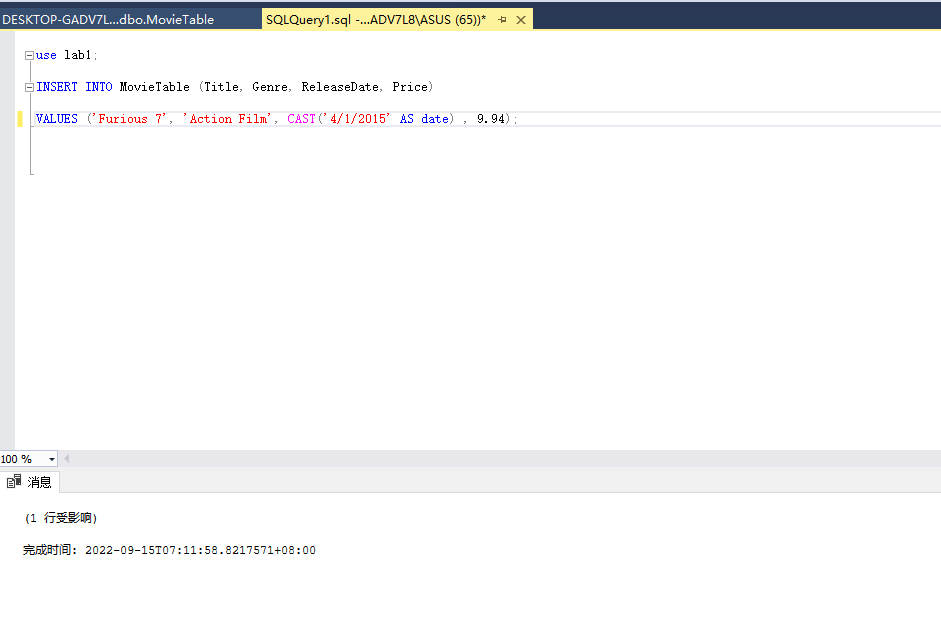


2：Insert the first row where the title is “Furious 7”, the genre is “Action Film”, the release date is 4/1/2015, and the price is $9.94.

use lab1;

INSERT INTO MovieTable (Title, Genre, ReleaseDate, Price)

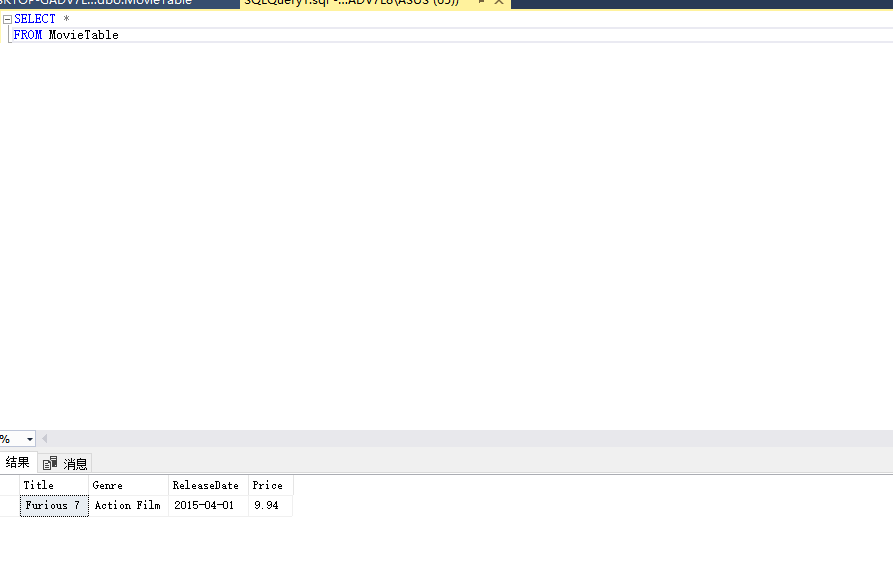
VALUES ('Furious 7', 'Action Film', CAST('4/1/2015' AS date) , 9.94);



3：Select all rows in the table to view the row you inserted.

SELECT \*

FROM MovieTable



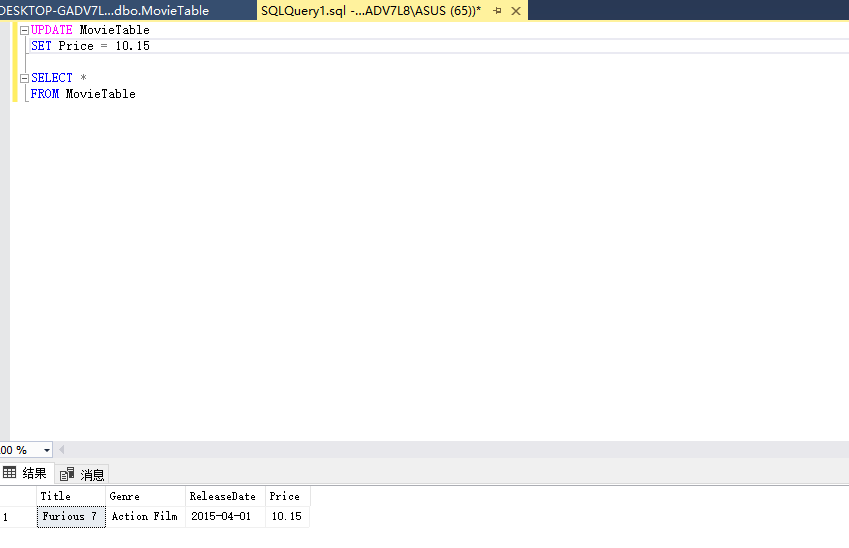
4：Update the price of the row in the table to $10.15, then select all rows in the table to view the row you updated.

UPDATE MovieTable

SET Price = 10.15

SELECT \*

FROM MovieTable

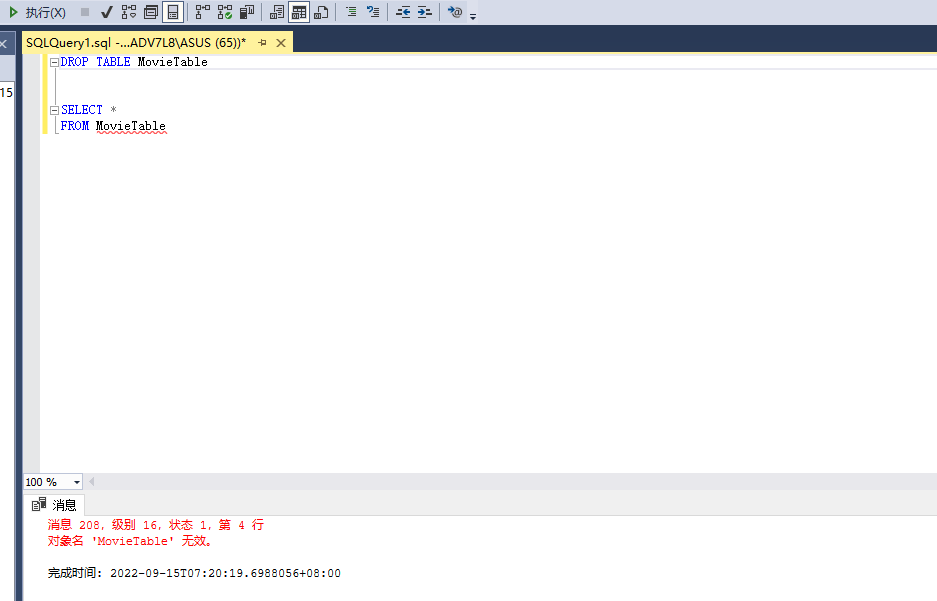


5：Remove all rows from the table, then select all rows in the table to verify there are no rows.

DELETE FROM MovieTable

SELECT \*

FROM MovieTable

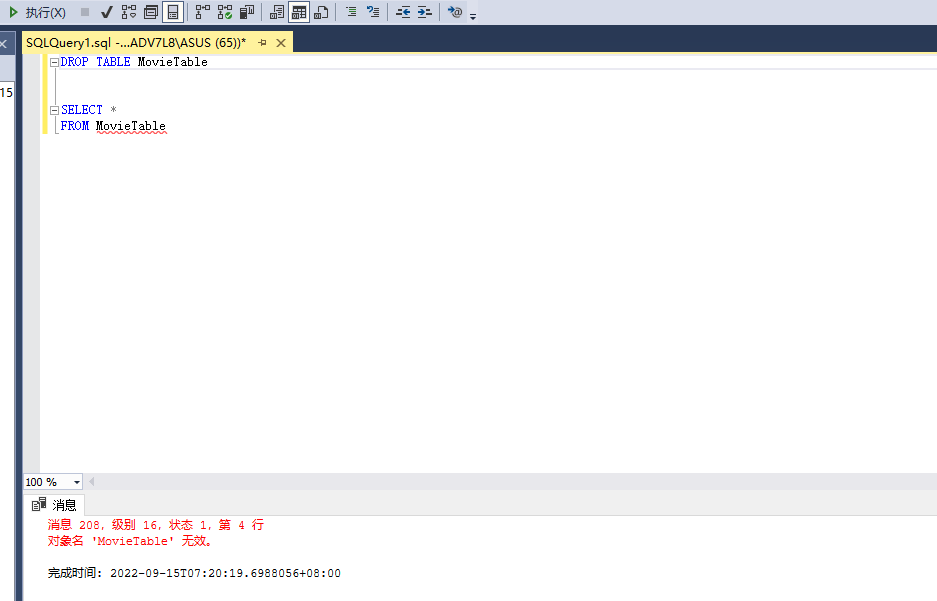


6:Drop the Movie table, then select all rows in the table to verify the table doesn’t exist.

DROP TABLE MovieTable

SELECT \*

FROM MovieTable



7:Explain how you would use the error message resulting from step #6, in conjunction with the SELECT command, to diagnose the error.

I will see the error said that line 4 has a problem and it said that: 'Invalid object name 'MovieTable''. Then I can see I write a 'FROM MovieTable' , there is only one 'MovieTable'. Combine these, it is means that the 'MovieTable' table does not exist, because we previously dropped this table. So if I want to use this table, I need to make a new project which is named 'MovieTable'.

**Section Two – More Precise Data Handling**

8. Create the Vacation table with its columns, datatypes, and constraints. As a reminder, make sure to follow along in the Lab 1 Explanations document as it shows you how to complete the steps.

Create Table Vacationtable(

Vacationld DECIMAL(12) Primary Key,

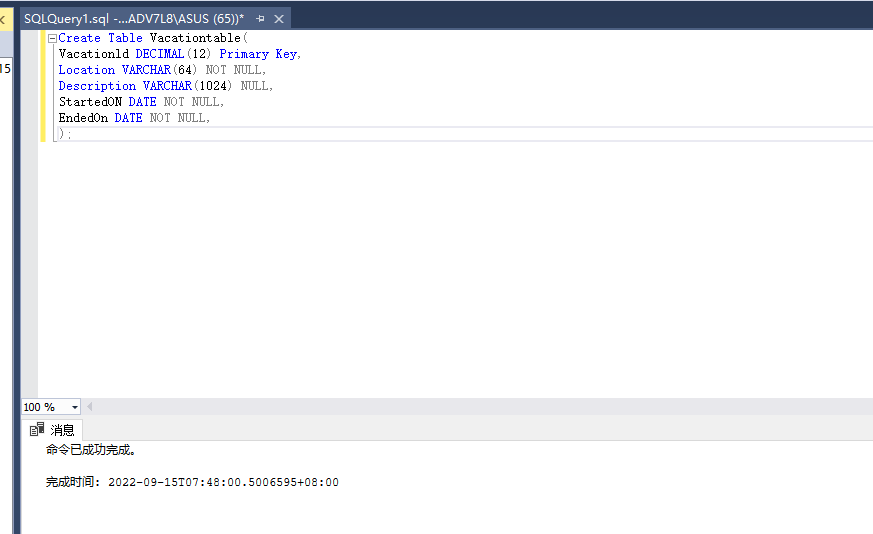
Location VARCHAR(64) NOT NULL,

Description VARCHAR(1024) NULL,

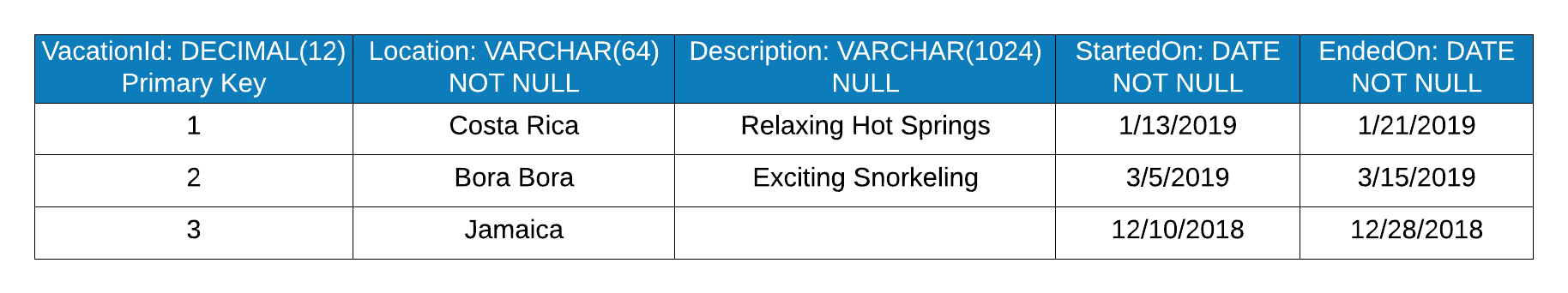
StartedON DATE NOT NULL,

EndedOn DATE NOT NULL,

);



9: Insert the rows illustrated in the figure below.

**

Note that the description for Jamaica is null.

INSERT INTO Vacationtable(Vacationld, Location, Description, StartedOn, EndedOn)

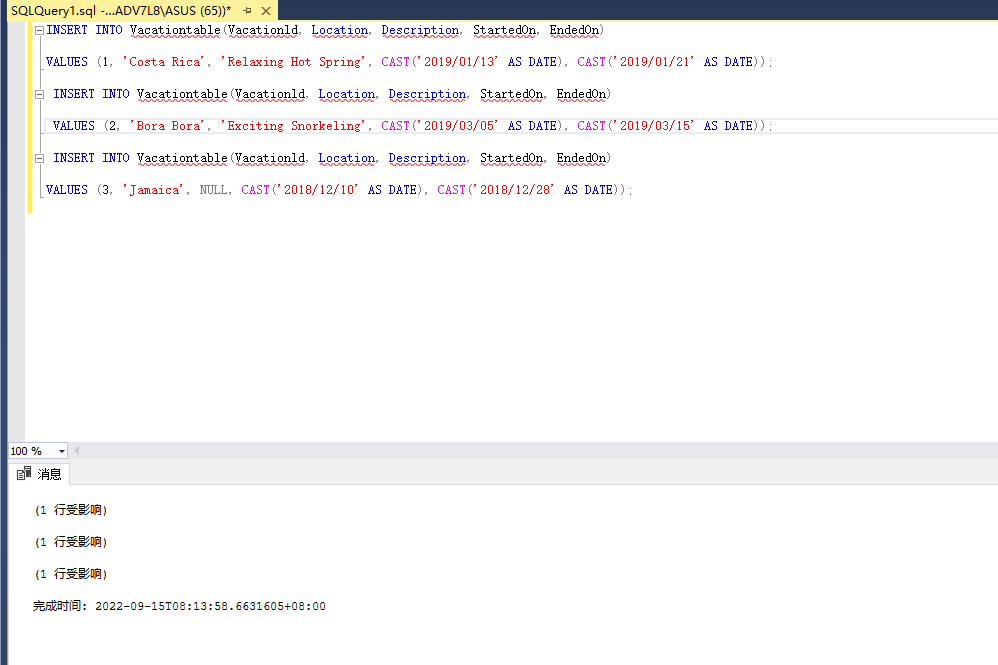
VALUES (1, 'Costa Rica', 'Relaxing Hot Spring', CAST('2019/01/13' AS DATE), CAST('2019/01/21' AS DATE));

INSERT INTO Vacationtable(Vacationld, Location, Description, StartedOn, EndedOn)

VALUES (2, 'Bora Bora', 'Exciting Snorkeling', CAST('2019/03/05' AS DATE), CAST('2019/03/15' AS DATE));

INSERT INTO Vacationtable(Vacationld, Location, Description, StartedOn, EndedOn)

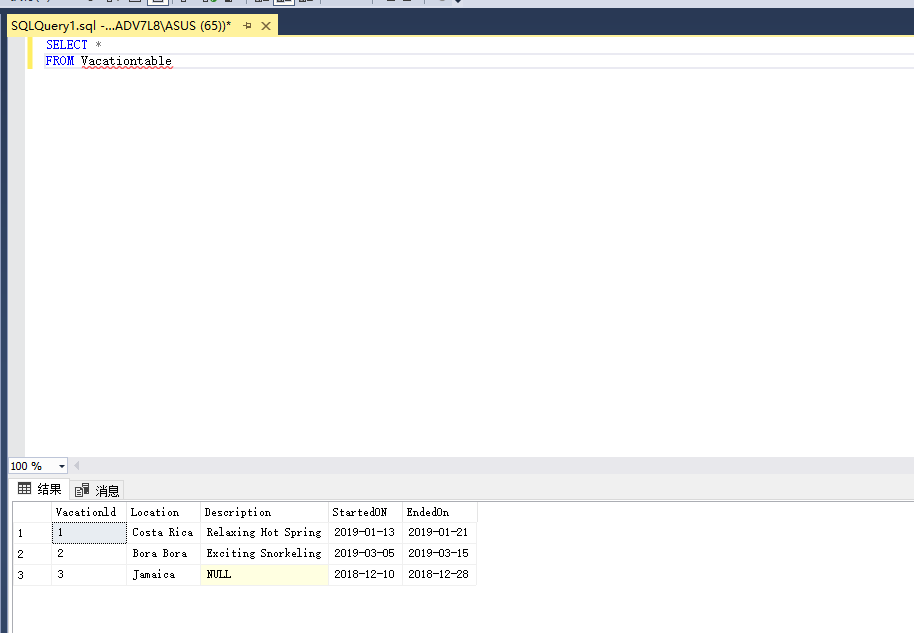
VALUES (3, 'Jamaica', NULL, CAST('2018/12/10' AS DATE), CAST('2018/12/28' AS DATE));



10:Select all rows from the Vacation table to show that the inserts were successful.

SELECT \*

FROM Vacationtable



11: Attempt to insert a row with the following values. The insert command will fail because the location column must have a value.

VacationId = 4

Location = NULL

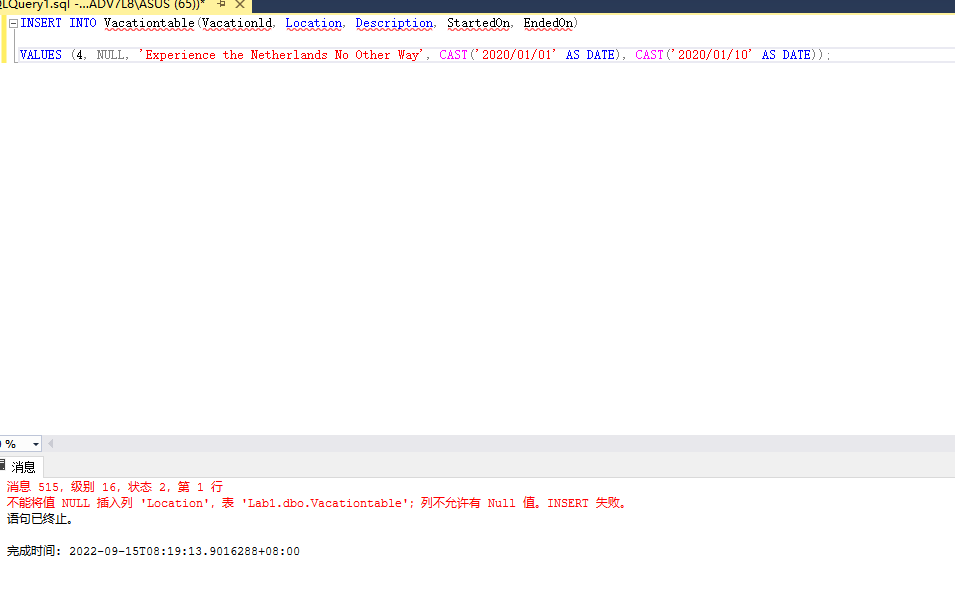
Description = Experience the Netherlands No Other Way

StartedOn = 1/1/2020

EndedOn = 1/10/2020

Explain how you would interpret the error message conclude that the location column is missing a required value.

The MS SQL error message directly tells us that it cannot insert NULL into 'Location', because we set 'Location' could not be NULL before. So we need to replace it with another value or reset 'Location' to allow NULL to exist.



12: Now insert the row with a location intact, with the following values.

VacationId = 4

Location = Netherlands

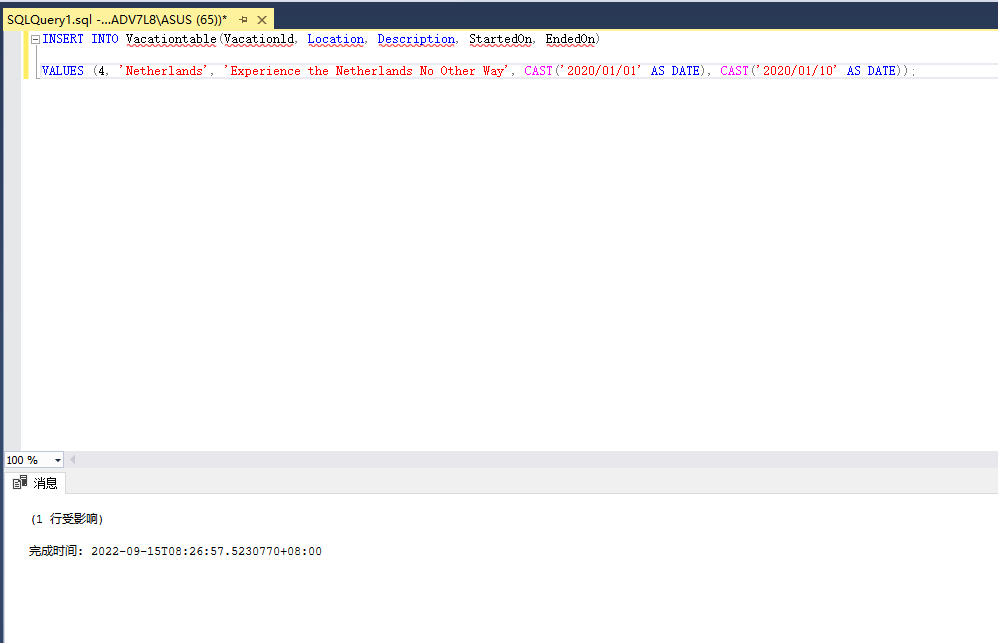
Description = Experience the Netherlands No Other Way

StartedOn = 1/1/2020

EndedOn = 1/10/2020

INSERT INTO Vacationtable(Vacationld, Location, Description, StartedOn, EndedOn)

VALUES (4, 'Netherlands', 'Experience the Netherlands No Other Way', CAST('2020/01/01' AS DATE), CAST('2020/01/10' AS DATE));



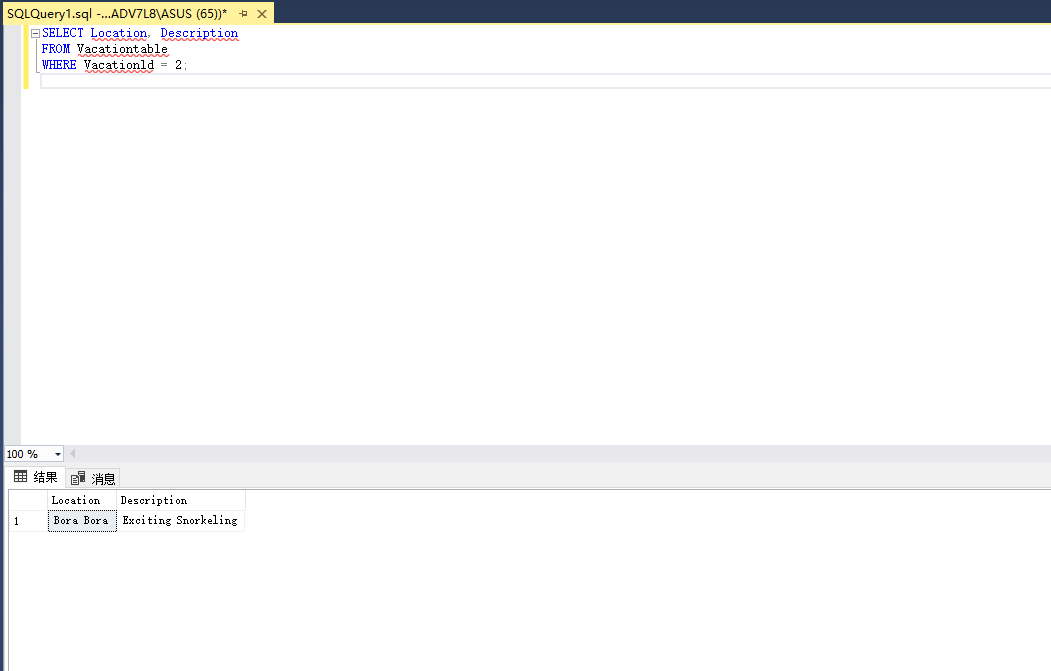
13: Retrieve only the location and description for the Bora Bora vacation, using the primary key as the column that determines which row is retrieved. Explain why it is useful to limit the number of rows and columns returned from a SELECT statement.

SELECT Location, Description

FROM Vacationtable

WHERE Vacationld = 2;

By limit the number of rows and columns returned from a SELECT statement. When the amount of data in the database is very large, we can quickly find the data we need and also speed up the operation.



14: The Jamaica vacation has no description. Update the row so that its description is “Aquatic Wonders”. Select all rows in the table to show that the update was successful.

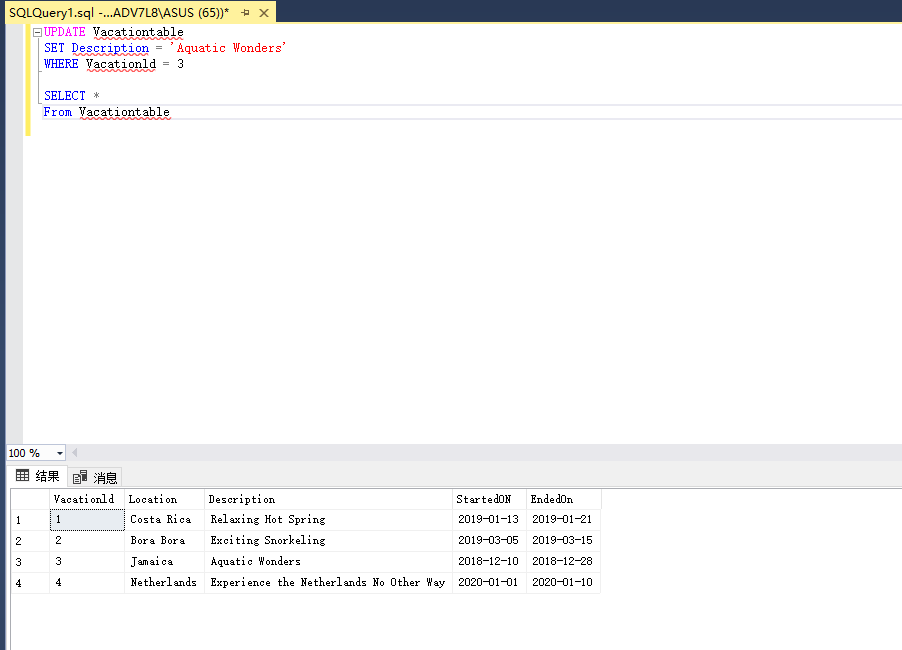
UPDATE Vacationtable

SET Description = 'Aquatic Wonders'

WHERE Vacationld = 3

SELECT \*

From Vacationtable



15: Update the Jamaica vacation so that it no longer has the description (i.e. its description is null). Select all rows in the table to show that the update was successful.

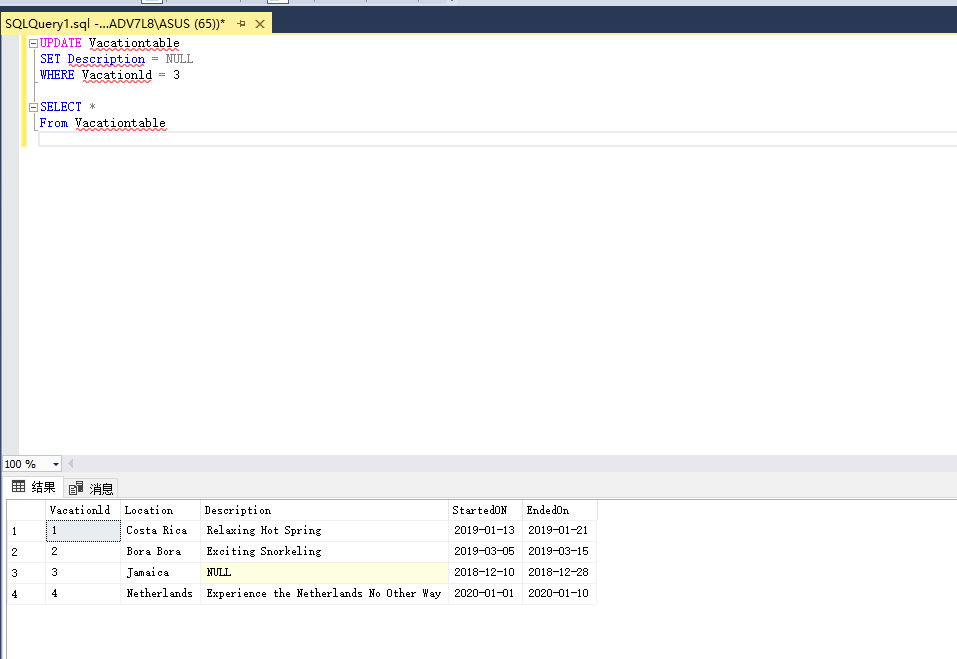
UPDATE Vacationtable

SET Description = NULL

WHERE Vacationld = 3

SELECT \*

From Vacationtable



16: Delete all rows where the vacation started on a date greater than June 1st, 2019, by using the StartedOn column as the determinant of which rows are deleted. Select all rows in the table to show the delete was successful.

DELETE FROM Vacationtable

WHERE StartedON<CAST('2019-06-01' AS DATE)

SELECT \*

From Vacationtable

