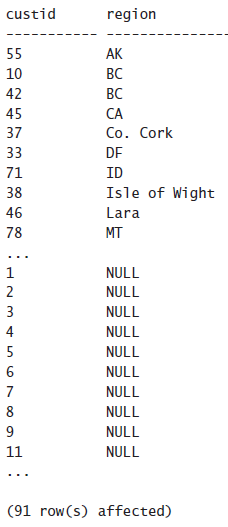
**Exercise 10**

By default, SQL Server sorts *NULLs* before non-*NULL* values. To get *NULLs* to sort last, you can use a *CASE* expression that returns 1 when the *region* column is *NULL* and 0 when it is not *NULL*. Specify this *CASE* expression as the first sort column and the *region* column as the second. This way, non-*NULLs* sort correctly among themselves first followed by *NULLs*. Here’s the complete solution query:



SELECT custid, region

FROM Sales.Customers

ORDER BY

CASE

WHEN region IS NULL THEN 1 ELSE 0

END

The query does the following

-It assigns a value of 1 if the region is a NULL

Else if it is not a NULL then it is assigned a value of 0

-By doing this we can sort the query in order

-By default, SQL orders in DESC (Low - High), So since we assigned 1(High) to NULL and 0(Low) to NOT NULL the table is displayed from NOT NULL(Low) to NULL(High)

-Thus, all the NULL’s are displayed last

If we wanted to display NULL’s first we could have done the following

SELECT custid, region

FROM Sales.Customers

ORDER BY

CASE

WHEN region IS NULL THEN 1 ELSE 0

END DESC

OR this, which would have been equivalent to the following code since SQL by

default sorts in ASC order

SELECT custid, region

FROM Sales.Customers