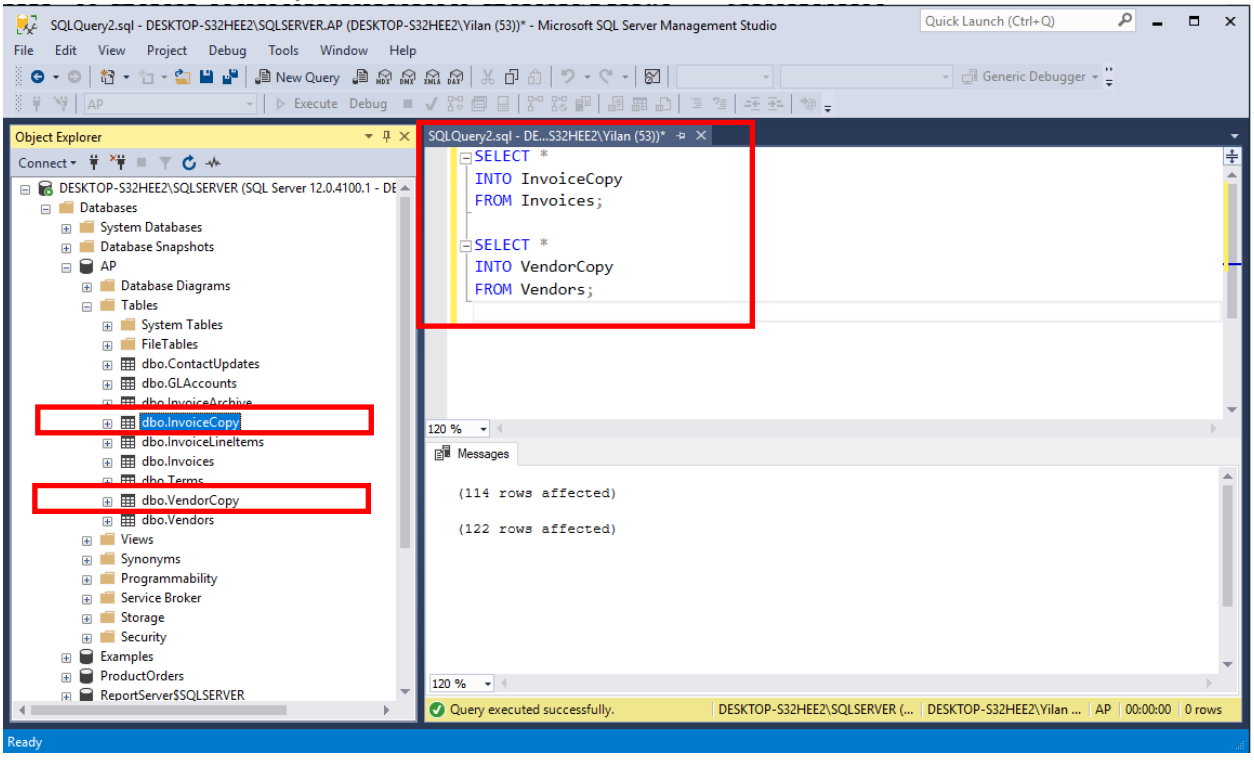
Lab 4: Data manipulation, Datatypes Solution

1. Create VendorCopy table and InvoiceCopy table.

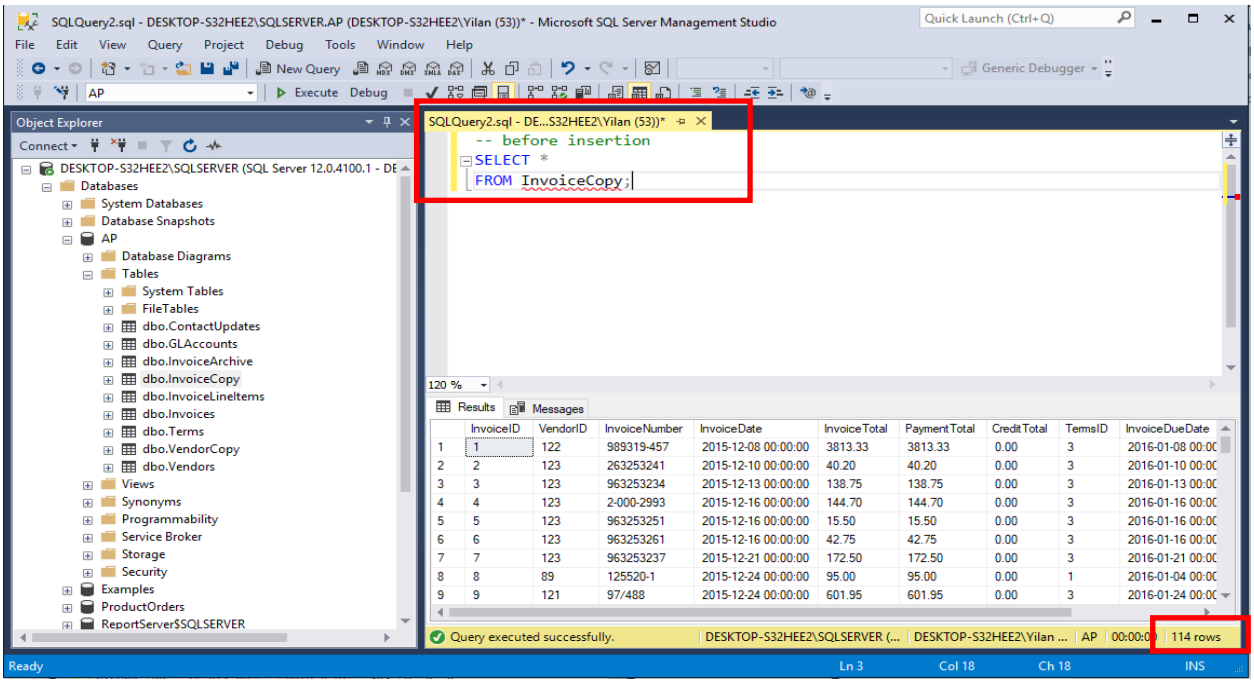
```
SELECT *
INTO InvoiceCopy
FROM Invoices;

SELECT *
INTO VendorCopy
FROM Vendors;
```



2. Write an INSERT statement that adds a row to the InvoiceCopy table with the following values (Use SELECT statement to verify data changes in the table before and after the modification):

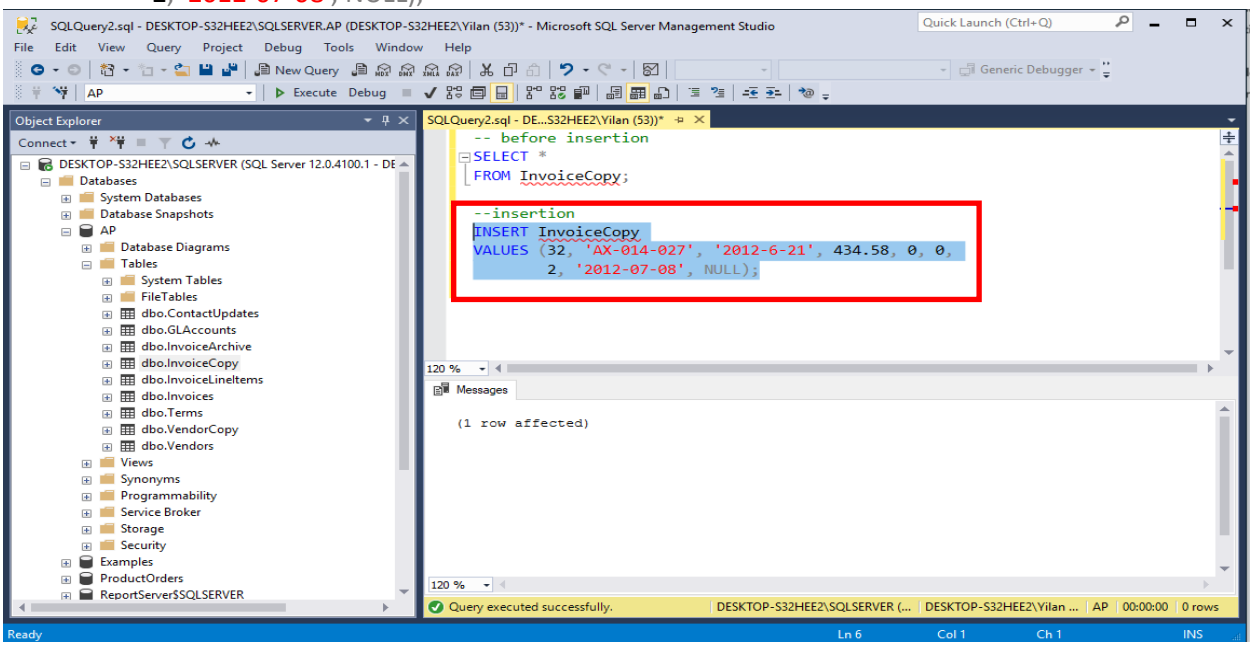
   VendorID: 32 InvoiceTotal: $ 434.58
   TermsID: 2 InvoiceNumber: AX-014-027
   PaymentTotal: $0.00 InvoiceDueDate:07/8/12
   InvoiceDate: 6/21/12 CreditTotal: $0.00

Before insertion:

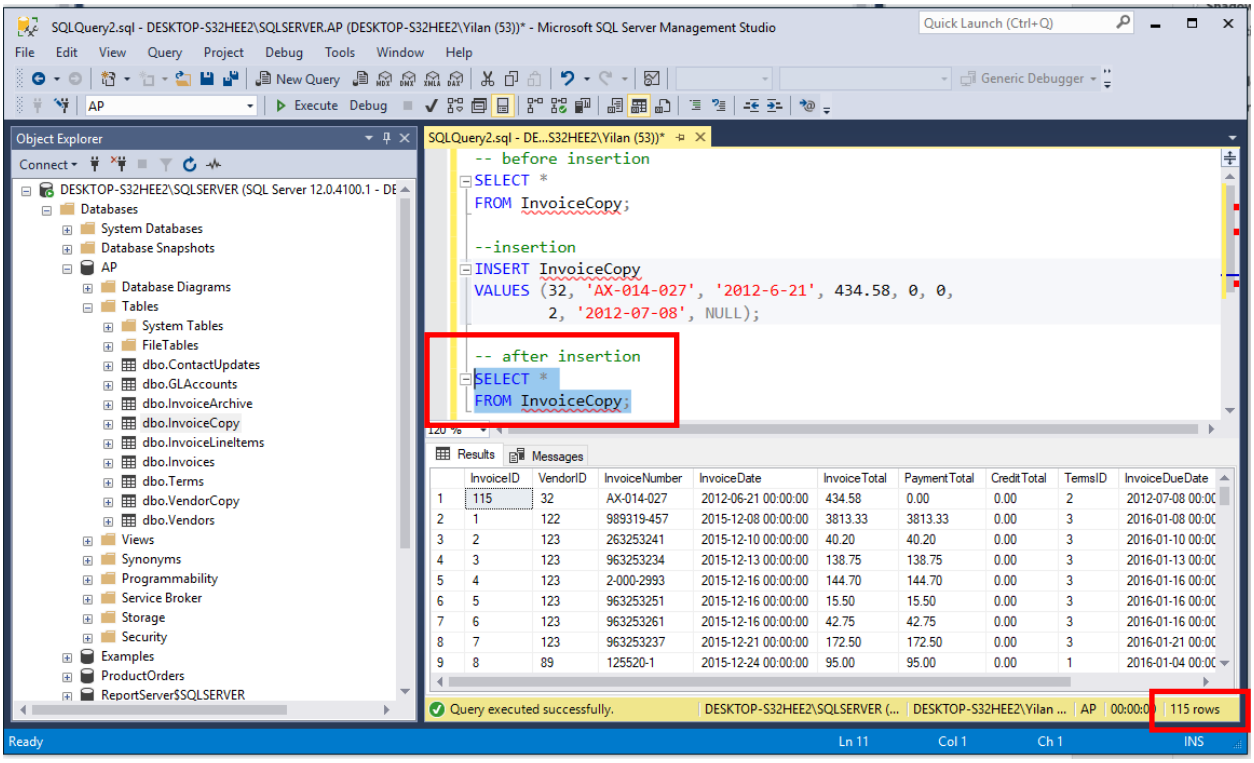


Insertion:

```
INSERT InvoiceCopy
VALUES (32, 'AX-014-027', '2012-6-21', 434.58, 0, 0,
        2, '2012-07-08', NULL);
```

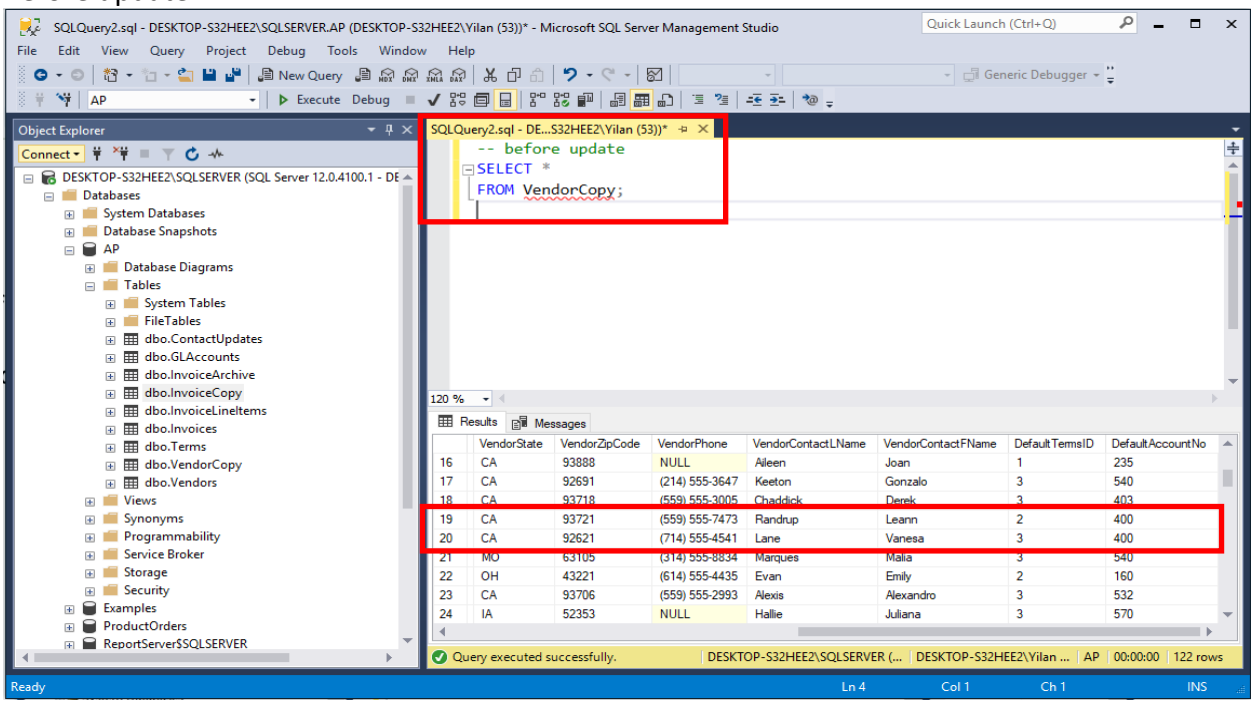


After insertion:



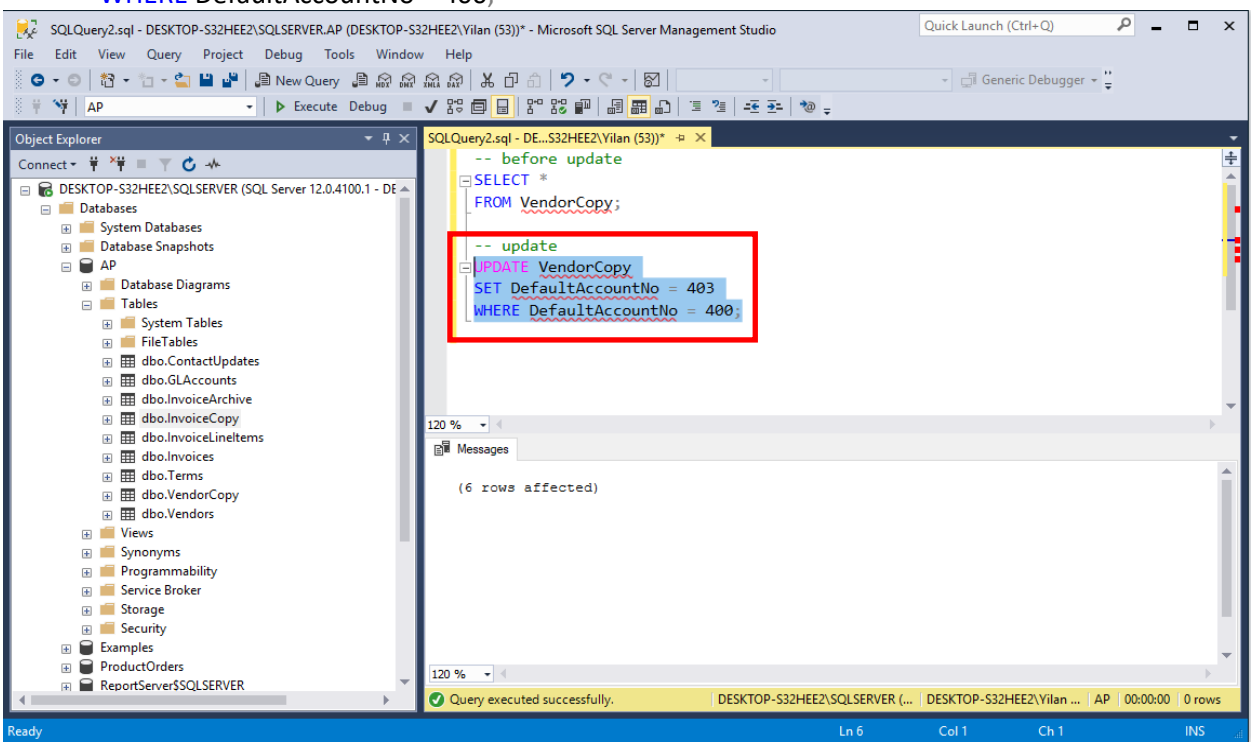
3. Write an UPDATE statement that modifies the VendorCopy table. Change the default account number to 403 for each vendor that has a default account number of 400. (Use SELECT statement to verify data changes in the table before and after the modification)

Before update:

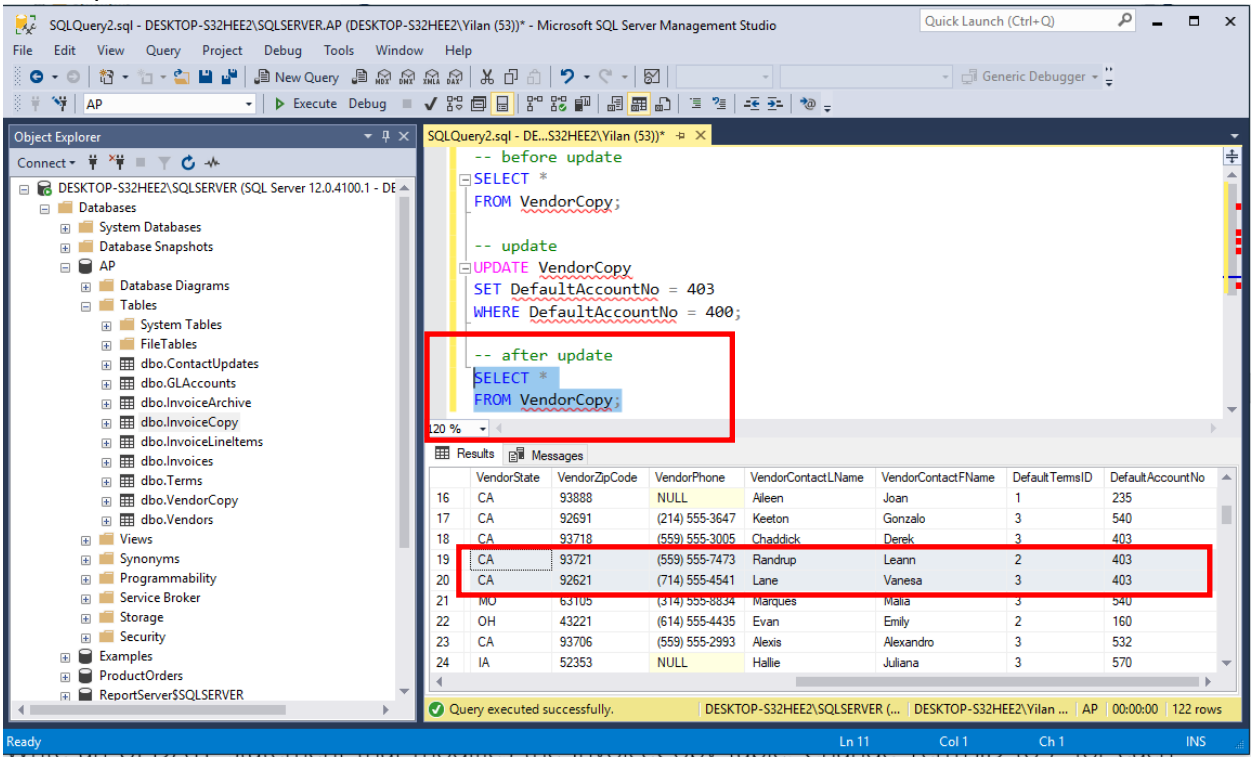


Update:

```
UPDATE VendorCopy
SET DefaultAccountNo = 403
WHERE DefaultAccountNo = 400;
```

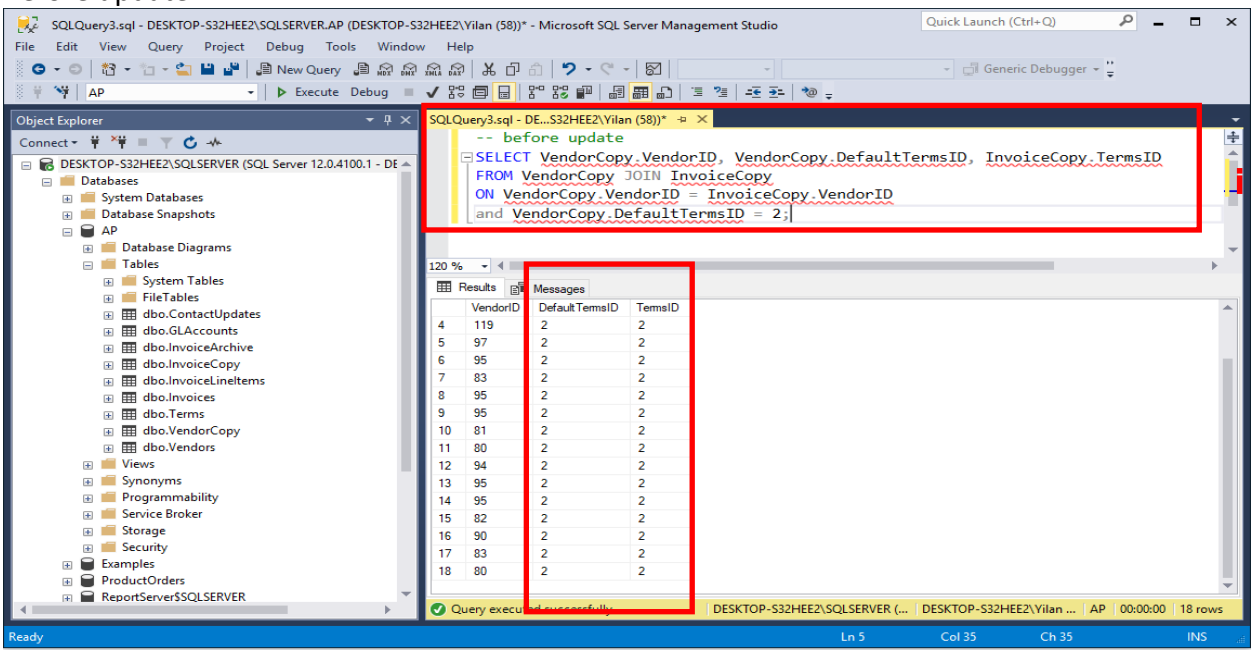


After update:



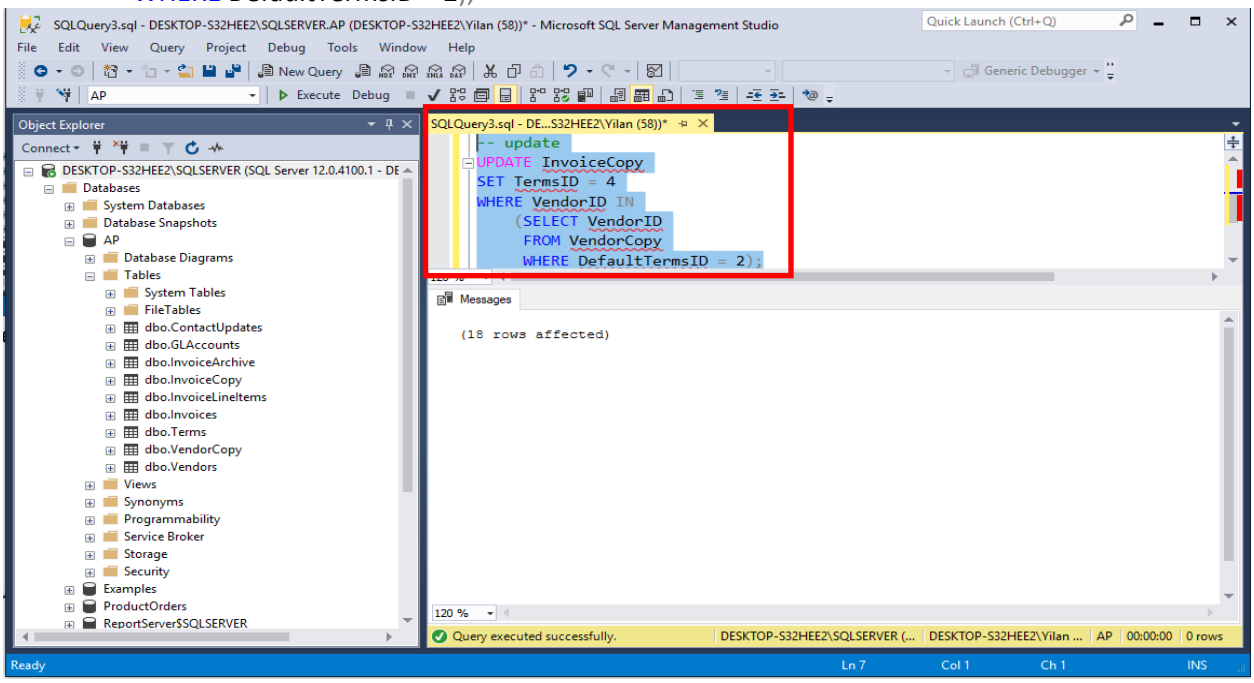
4. Write an UPDATE statement that modifies the InvoiceCopy table. Change TermsID to 4 for each invoice that's from a vendor with a defaultTermsID of 2. Use a subquery. (Use SELECT statement to verify data changes in the table before and after the modification)

Before update:

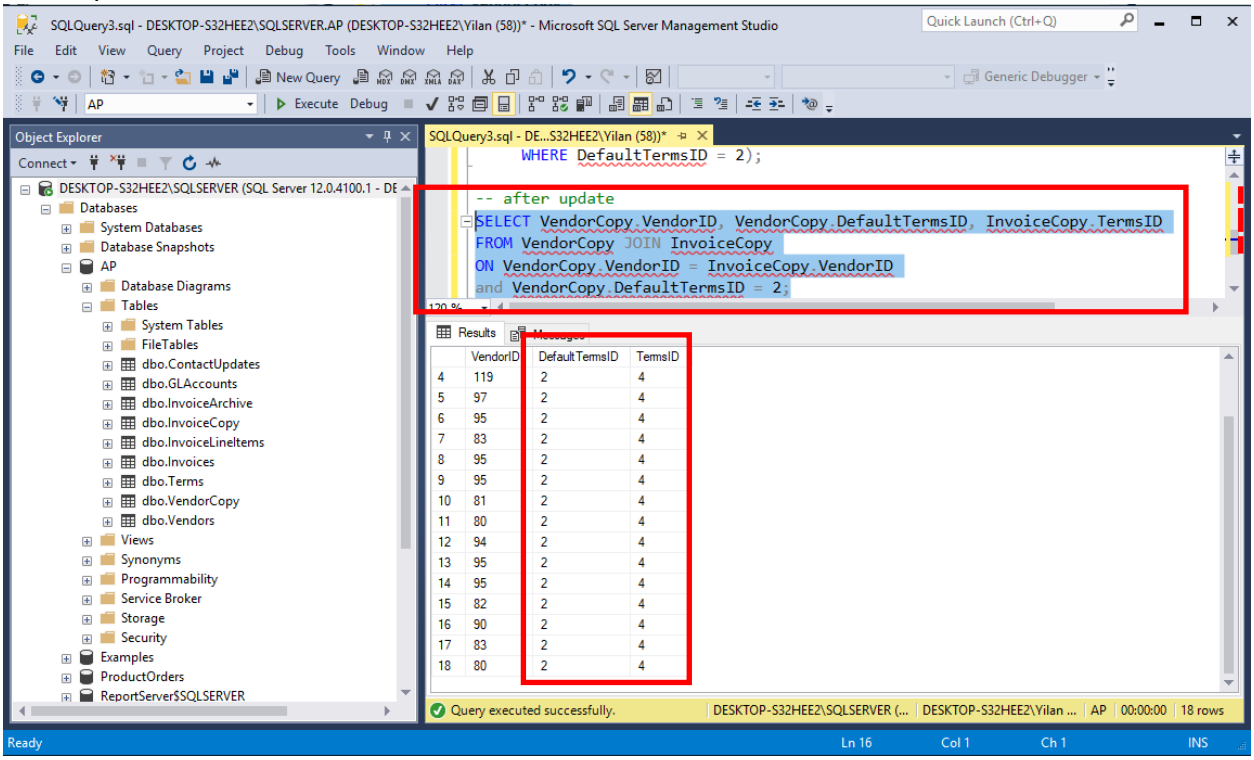


Update:

```
UPDATE InvoiceCopy
SET TermsID = 4
WHERE VendorID IN
   (SELECT VendorID
    FROM VendorCopy
    WHERE DefaultTermsID = 2);
```

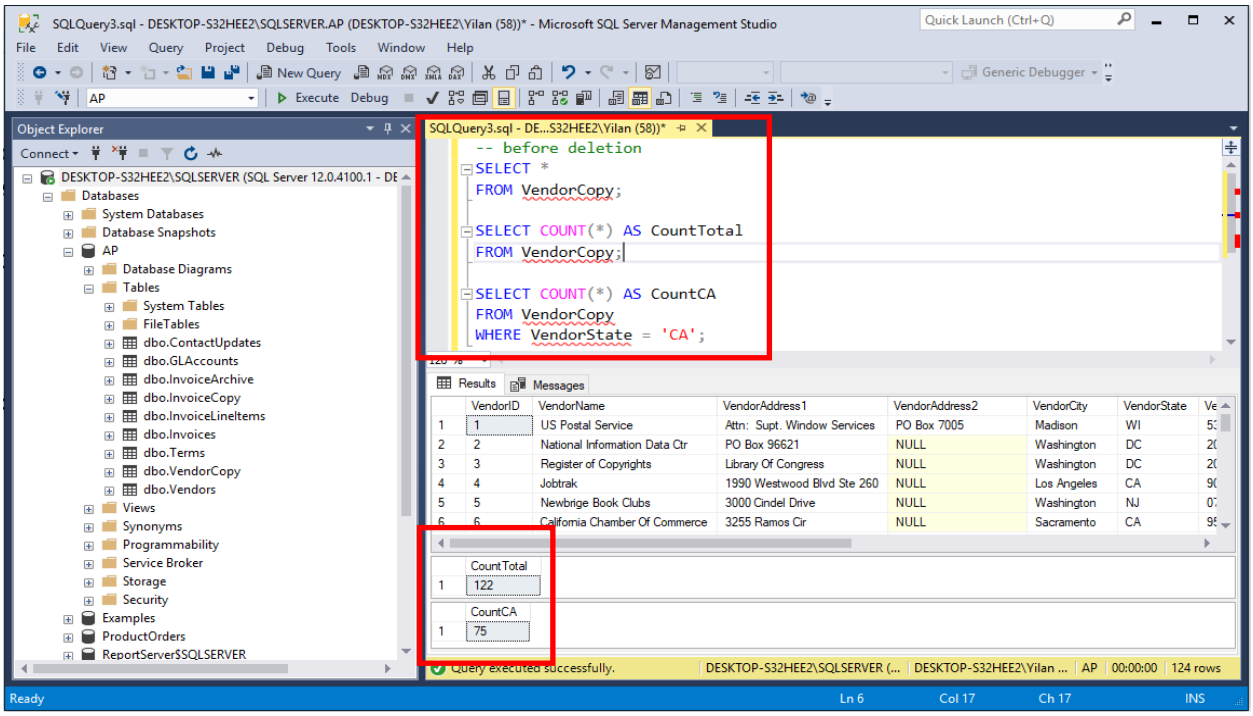


After update:



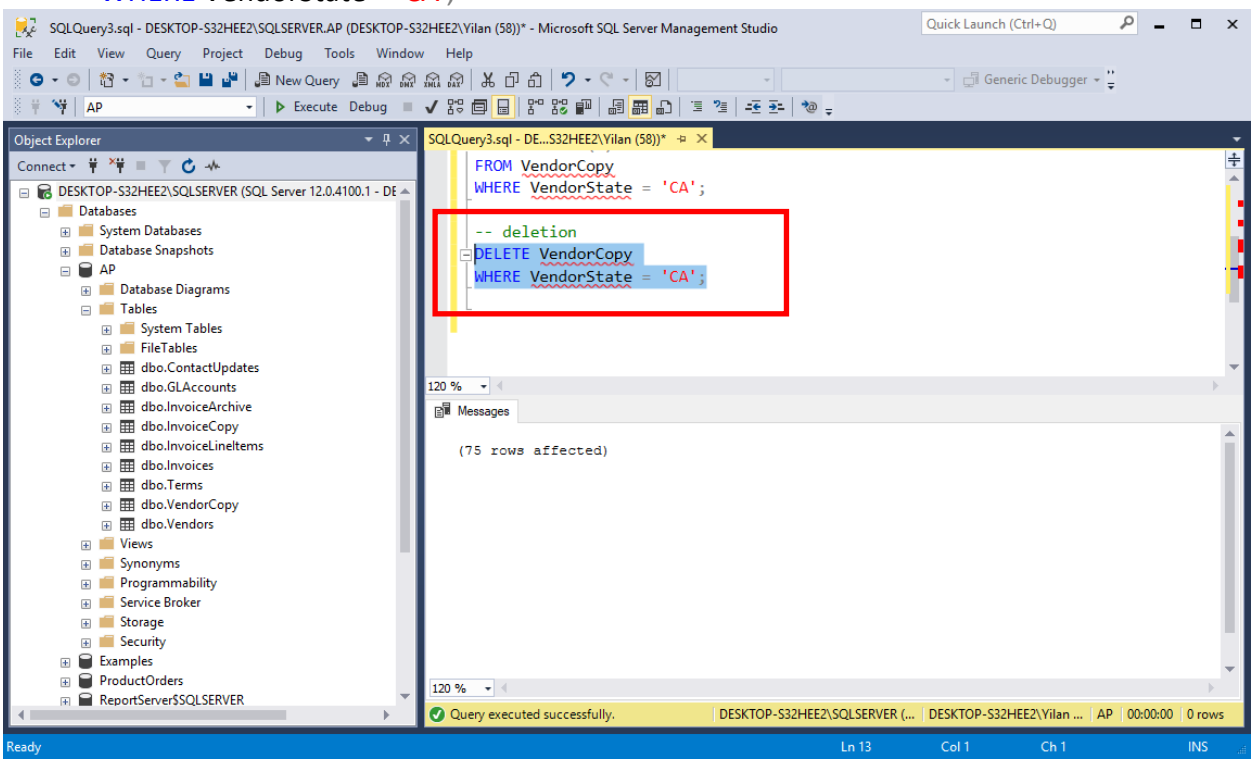
5. Write a DELETE statement that deletes all vendors in the state of California from the VendorCopy table. (Use SELECT statement to verify data changes in the table before and after the modification)

Before deletion:

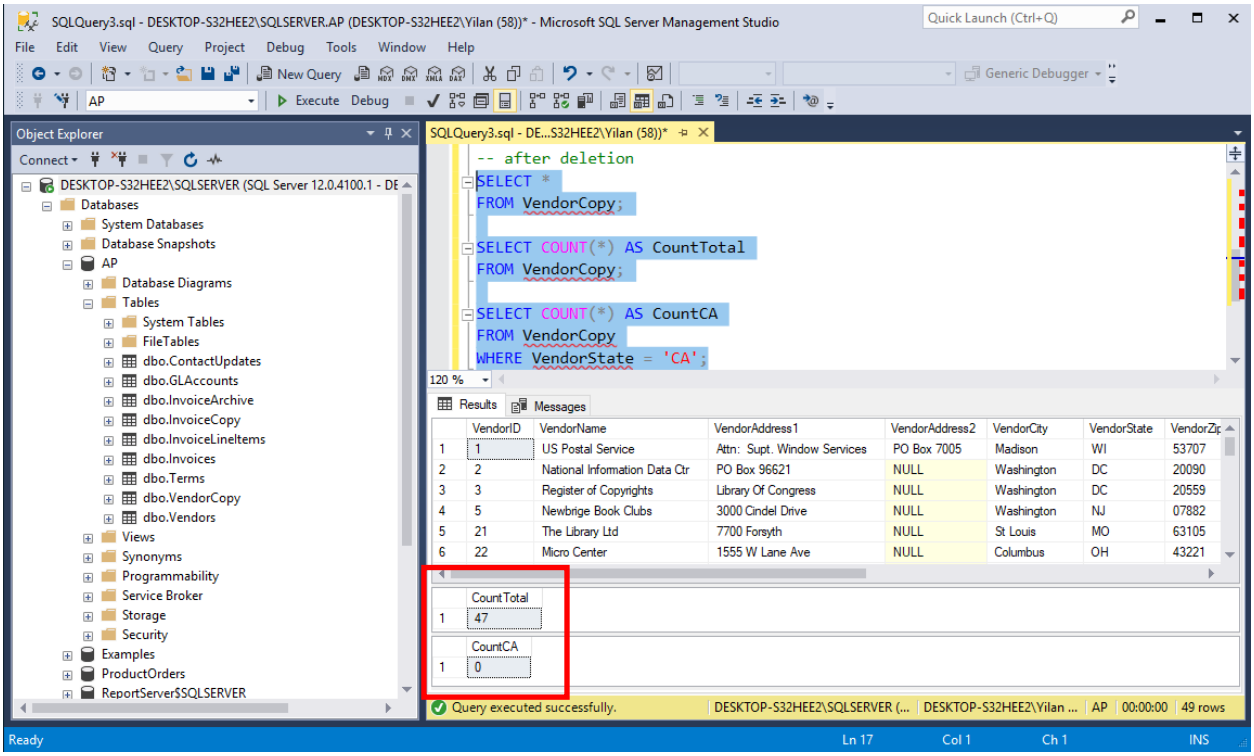


Deletion:

```
DELETE VendorCopy
WHERE VendorState = 'CA';
```

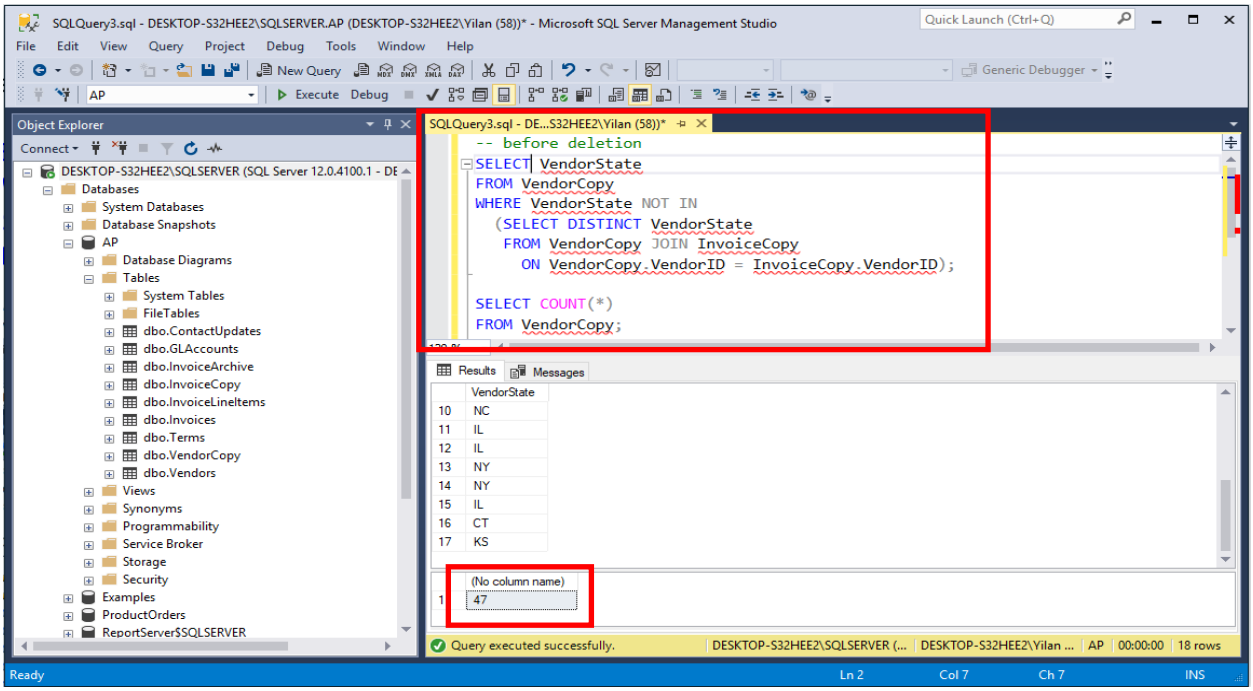


After deletion:

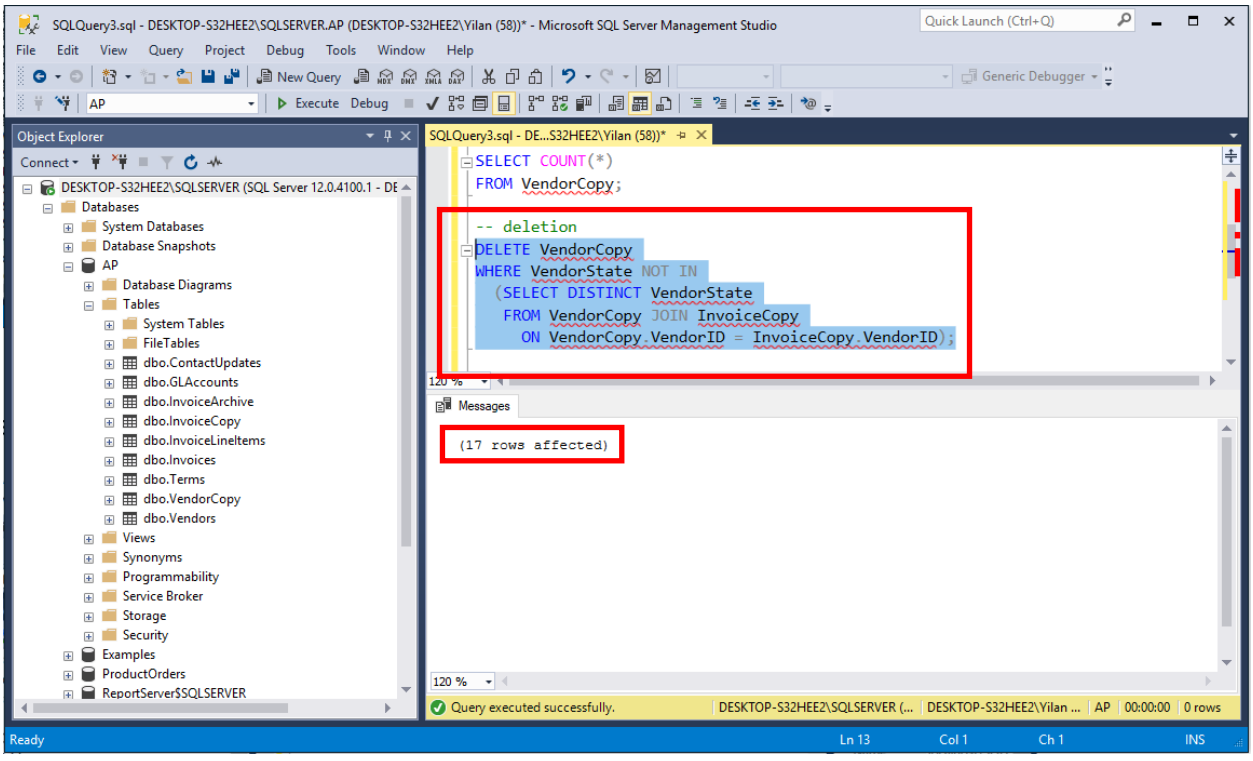


6. Write a DELETE statement for the VendorCopy table. Delete the vendors that are located in states from which no vendor has ever sent an invoice. (Use SELECT statement to verify data changes in the table before and after the modification)
   Hint: Use a subquery coded with "SELECT DISTINCT VendorState" introduced with the NOT IN operator.

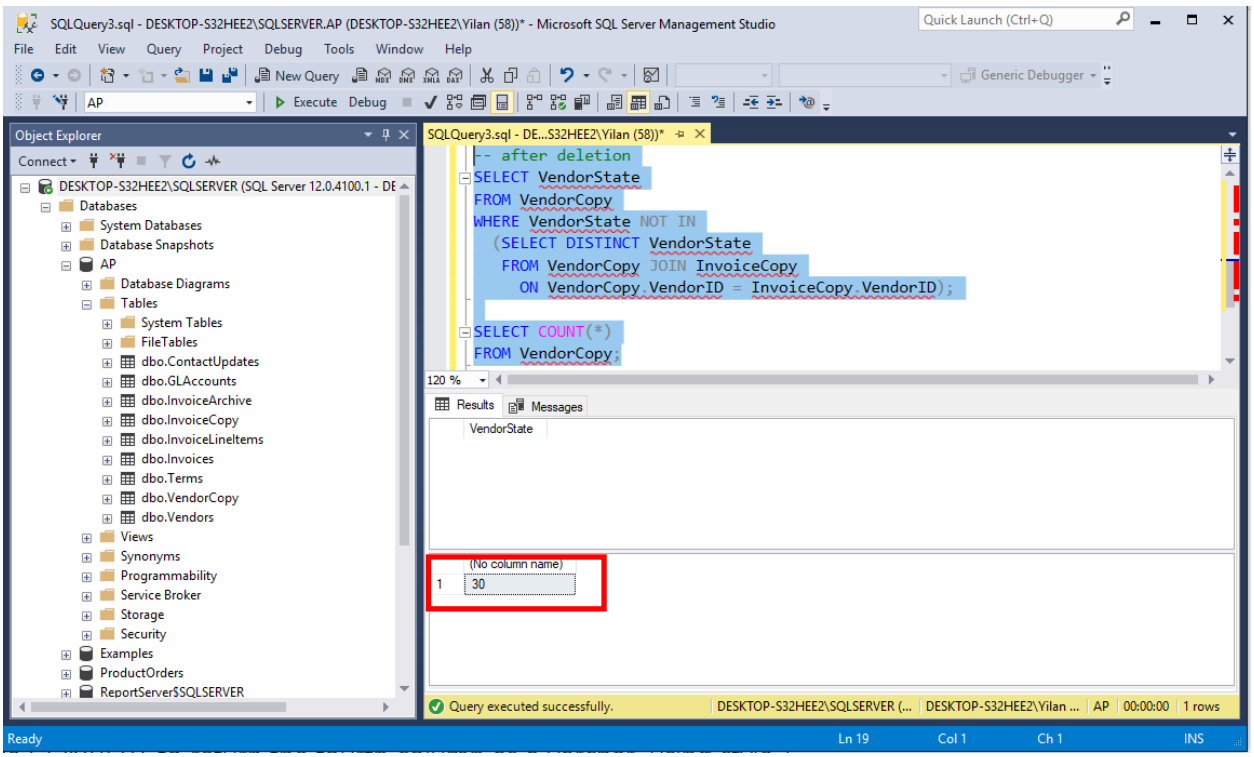
Before deletion:



Deletion:

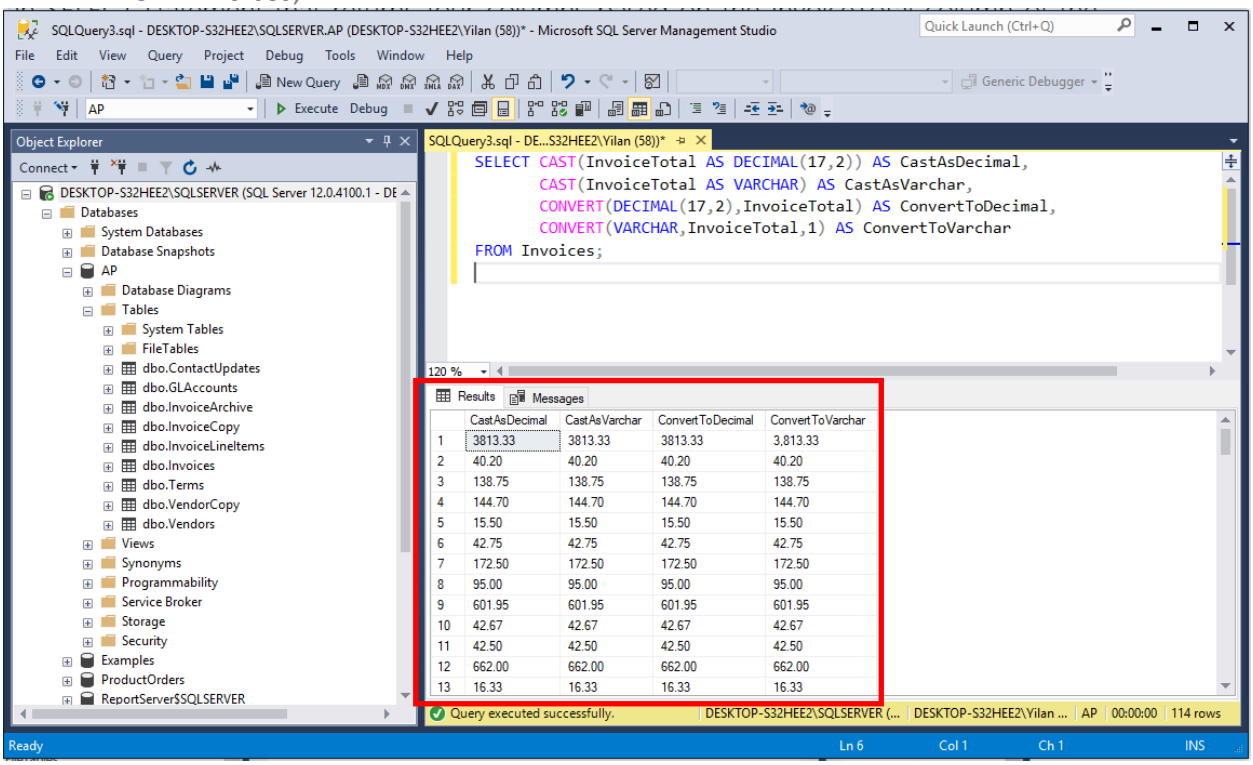


After deletion:



7. Write an SELECT statement that returns four columns based on the InvoiceTotal column of the Invoices table:
   - Use CAST function to return the first column as data type decimal with 2 digits to the right of the decimal point.
   - Use CAST to return the second column as a varchar
   - Use CONVERT function to return the third column as the same data type as the first column.
   - Use CONVERT to return the fourth column as a varchar, using style 1.

```
SELECT CAST(InvoiceTotal AS DECIMAL(17,2)) AS CastAsDecimal,
    CAST(InvoiceTotal AS VARCHAR) AS CastAsVarchar,
    CONVERT(DECIMAL(17,2),InvoiceTotal) AS ConvertToDecimal,
    CONVERT(VARCHAR,InvoiceTotal,1) AS ConvertToVarchar
FROM Invoices;
```



8. Write a SELECT statement that returns four columns based on the InvoiceDate column of the Invoices table:
   - Use the CAST function to return the first column as data type varchar.
   - Use the CONVERT function to return the second and third columns as a varchar, using style 1 and style 10, respectively.
   - Use the CAST function to return the fourth column as data type real.

```
SELECT CAST(InvoiceDate AS varchar) AS CastAsVarchar,
       CONVERT(varchar,InvoiceDate,1) AS ConvertToStyle1,
       CONVERT(varchar,InvoiceDate,10) AS ConvertToStyle10,
       CAST(InvoiceDate AS real) AS ConvertToReal
FROM Invoices;
```