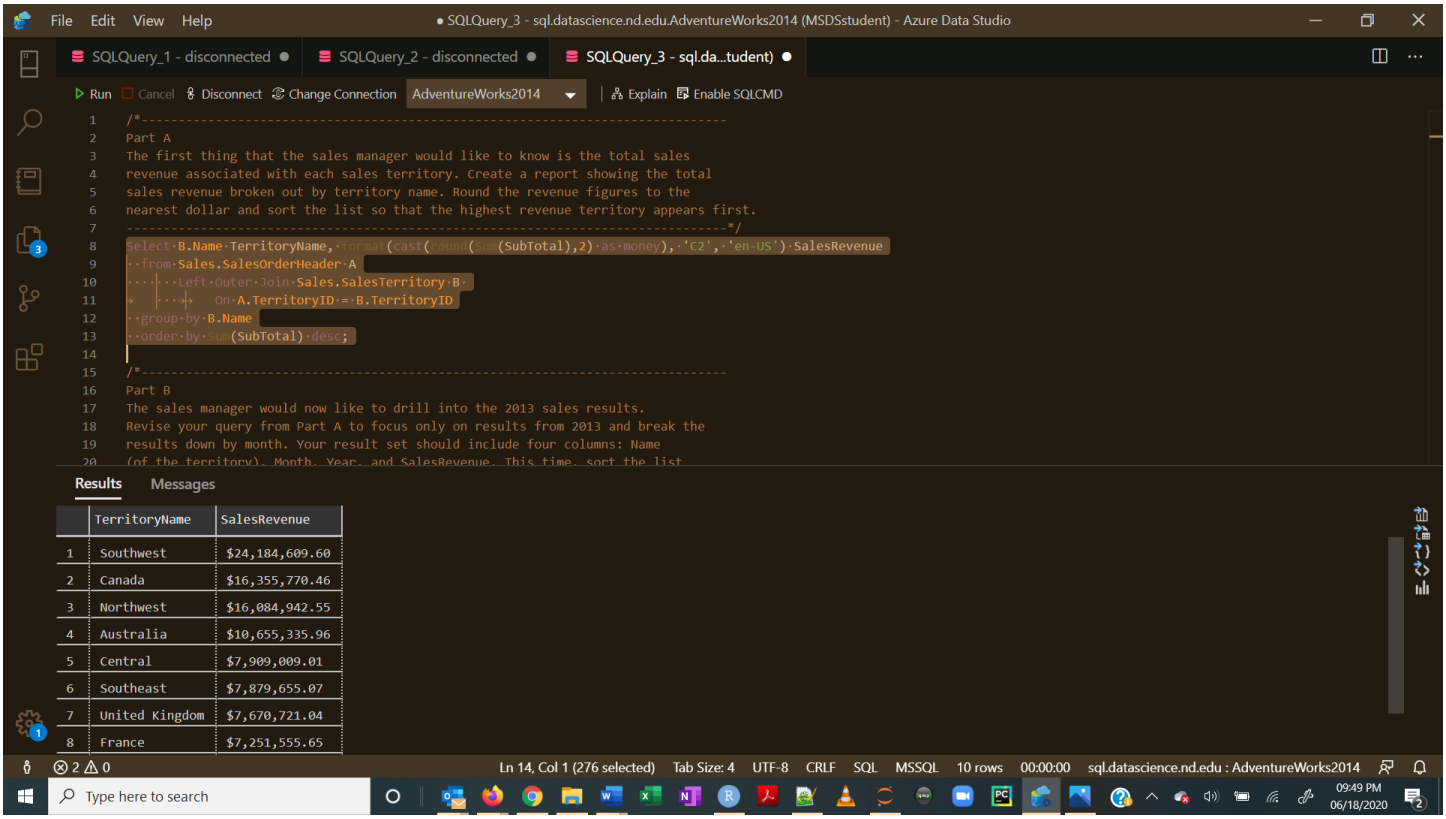
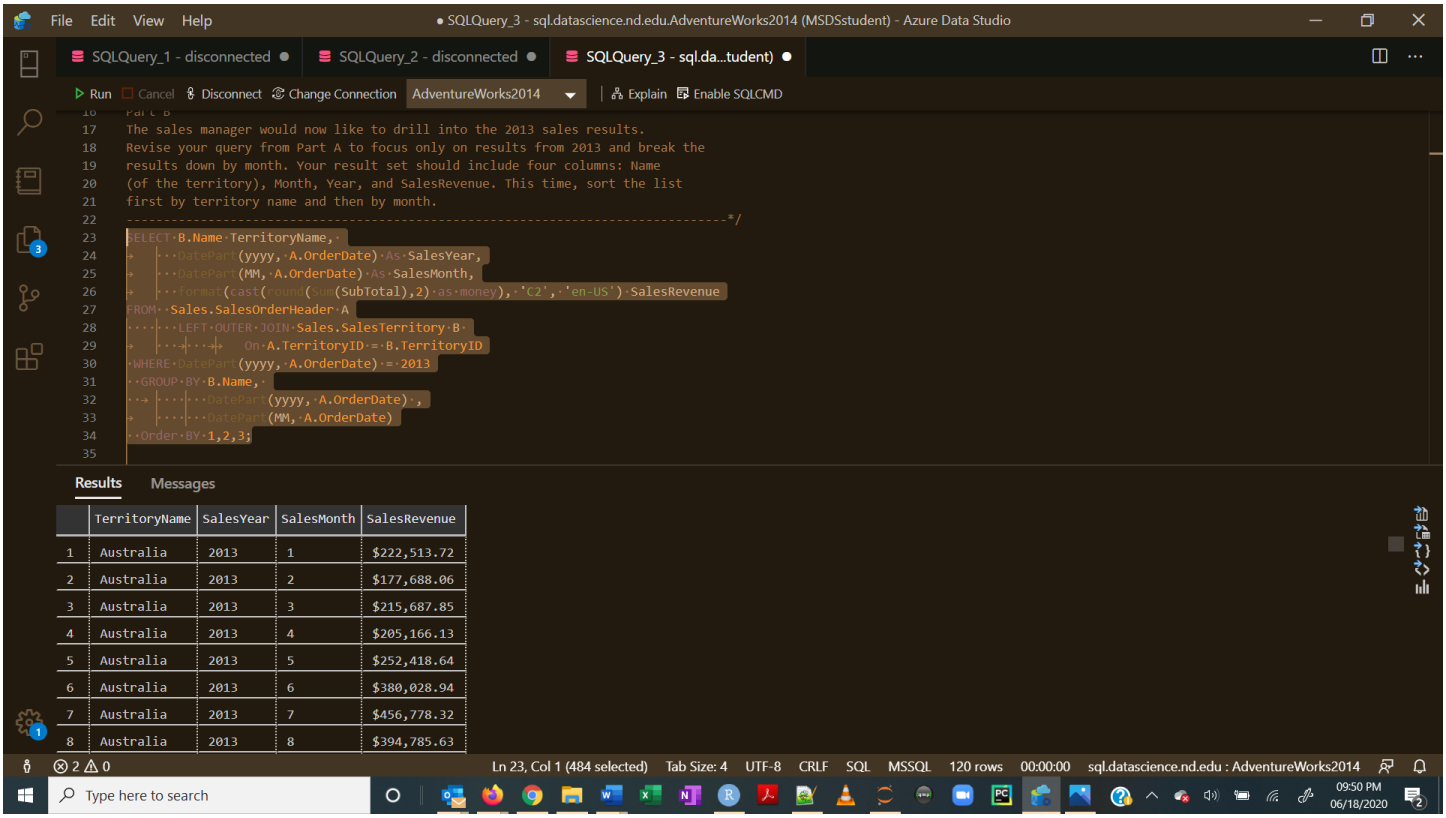
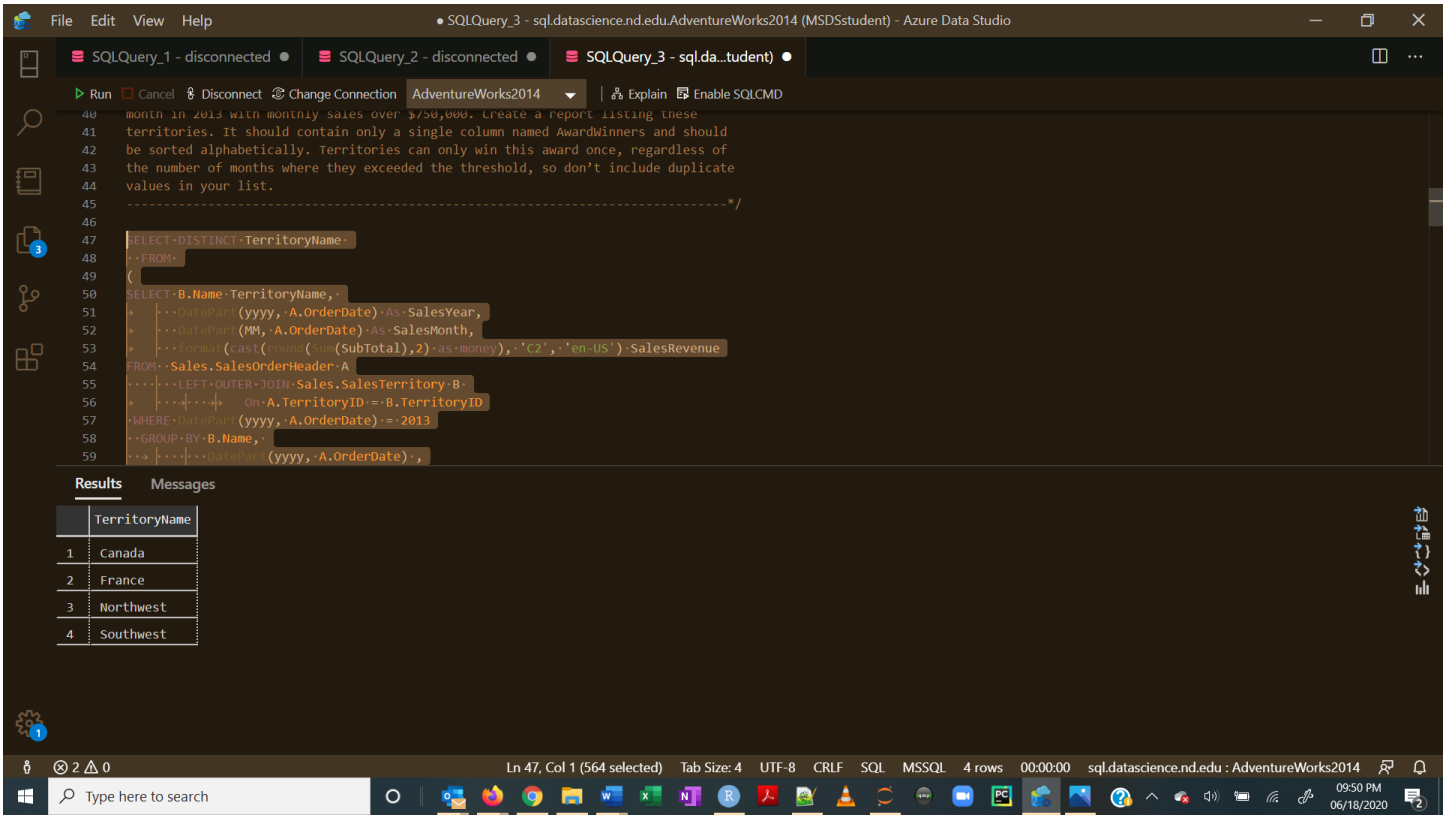
```
/*--------------------------------------------------------------------------------
Part A
The first thing that the sales manager would like to know is the total sales
revenue associated with each sales territory. Create a report showing the total
sales revenue broken out by territory name. Round the revenue figures to the
nearest dollar and sort the list so that the highest revenue territory appears first.
--------------------------------------------------------------------------------*/
SELECT B.Name TerritoryName,
       FORMAT(CAST(ROUND(SUM(SubTotal),2) AS MONEY), 'C2') SalesRevenue
  FROM Sales.SalesOrderHeader A
       LEFT OUTER JOIN Sales.SalesTerritory B
                    On A.TerritoryID = B.TerritoryID
 GROUP BY B.Name
 ORDER BY SUM(SubTotal) DESC;
```



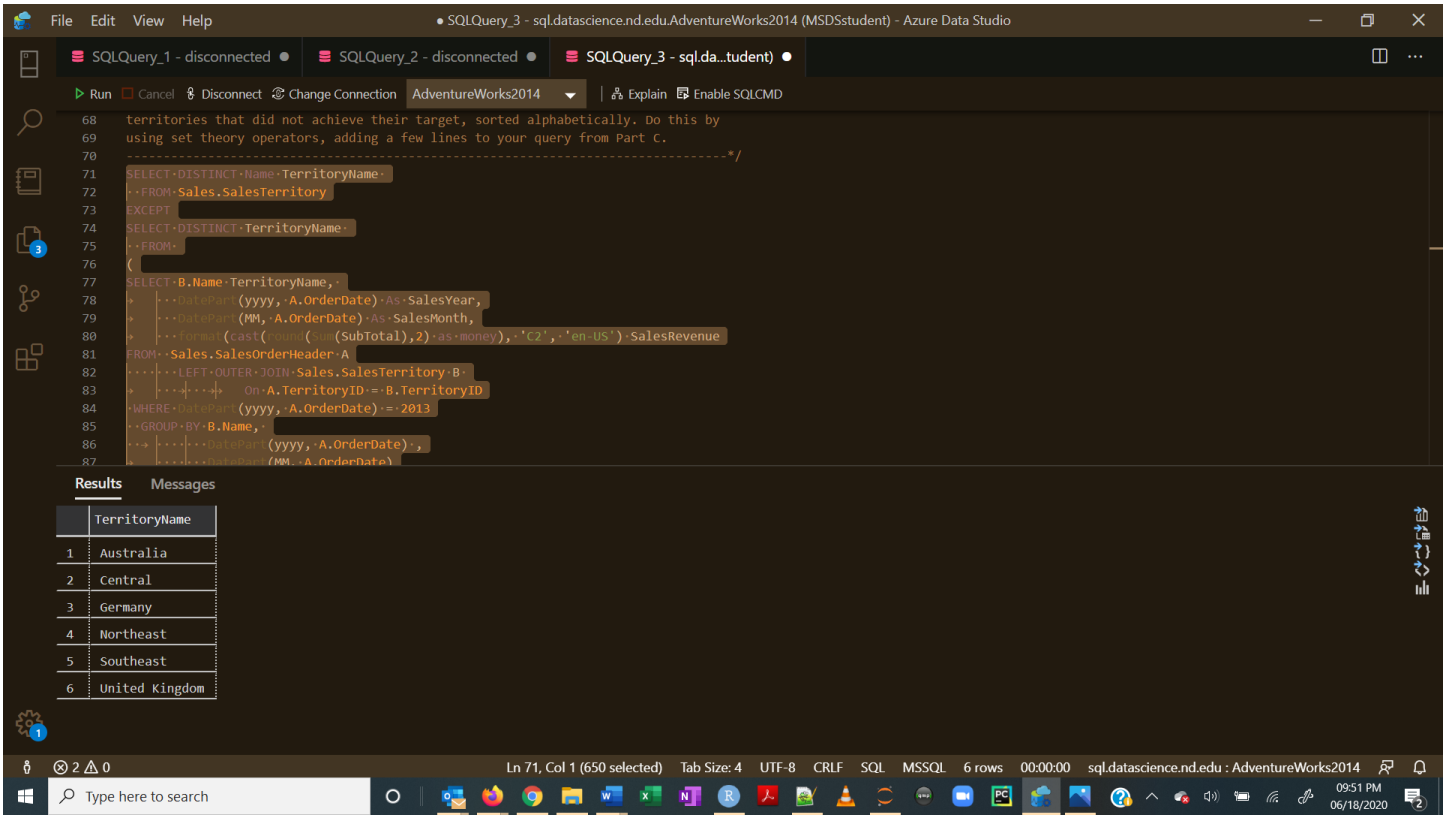
```
/*--------------------------------------------------------------------------
Part B
The sales manager would now like to drill into the 2013 sales results.
Revise your query from Part A to focus only on results from 2013 and break the
results down by month. Your result set should include four columns: Name
(of the territory), Month, Year, and SalesRevenue. This time, sort the list
first by territory name and then by month.
--------------------------------------------------------------------------
SELECT B.Name TerritoryName,
        DATEPART(yyyy, A.OrderDate) AS SalesYear,
        DATEPART(MM, A.OrderDate)   AS SalesMonth,
        FORMAT(CAST(ROUND(SUM(SubTotal),2) AS money), 'C2') SalesRevenue
  FROM Sales.SalesOrderHeader A
       LEFT OUTER JOIN Sales.SalesTerritory B
                    ON A.TerritoryID = B.TerritoryID
 WHERE DATEPART(yyyy, A.OrderDate) = 2013
 GROUP BY  B.Name,
           DATEPART(yyyy, A.OrderDate) ,
           DATEPART(MM,   A.OrderDate)
 ORDER BY 1,2,3;
```



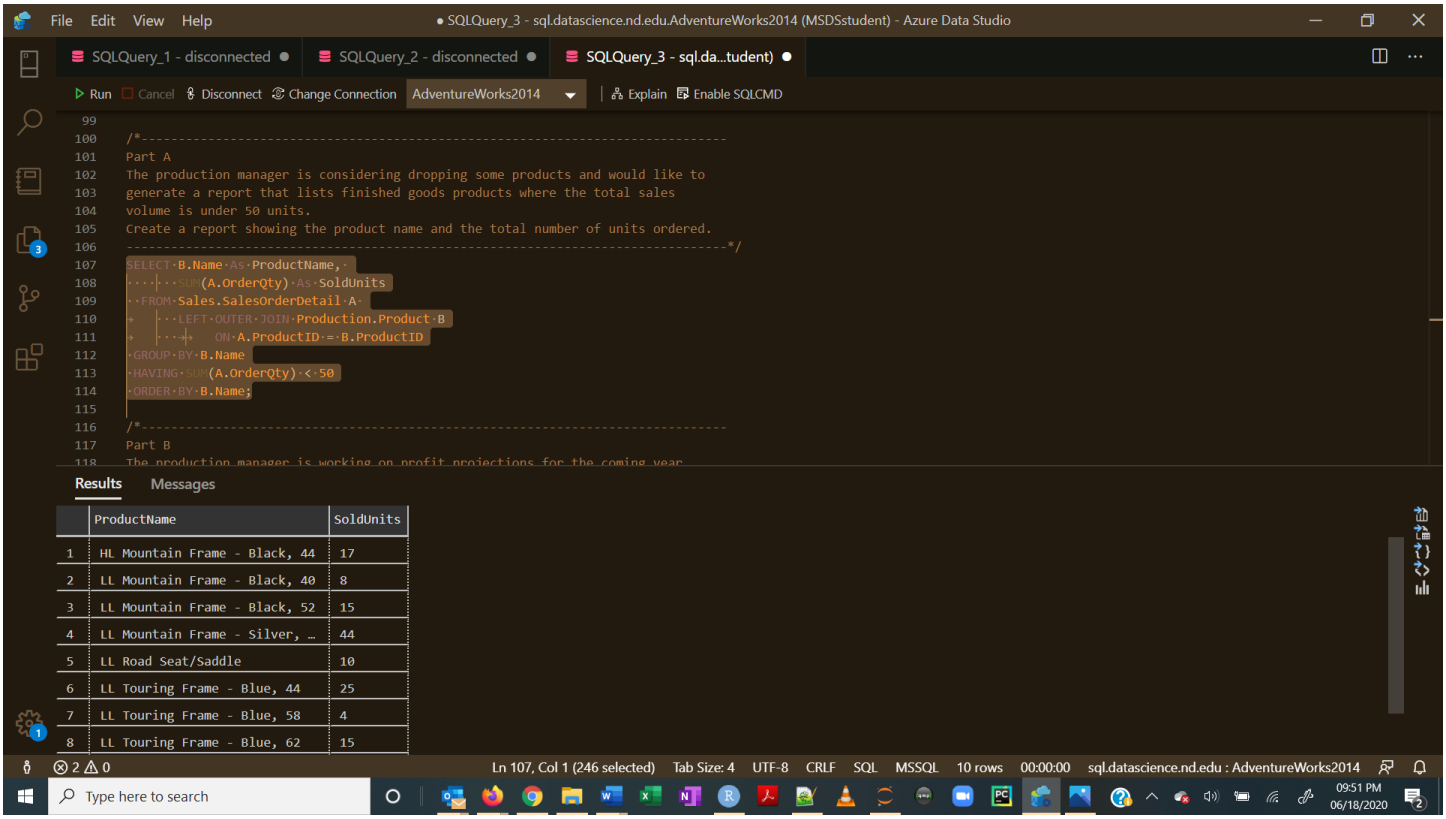
```
/*--------------------------------------------------------------------------------
Part C
The sales manager would like to recognize territories that had at least one
month in 2013 with monthly sales over $750,000. Create a report listing these
territories. It should contain only a single column named AwardWinners and should be sorted
alphabetically. Territories can only win this award once, regardless of the number of months
where they exceeded the threshold, so don't include duplicate values in your list.
--------------------------------------------------------------------------------
SELECT DISTINCT TerritoryName
  FROM (
        SELECT B.Name TerritoryName,
               DATEPART(yyyy, A.OrderDate) AS SalesYear,
               DATEPART(MM, A.OrderDate)   AS SalesMonth,
               FORMAT(CAST(ROUND(SUM(SubTotal),2) AS money), 'C') SalesRevenue
          FROM Sales.SalesOrderHeader A
               LEFT OUTER JOIN Sales.SalesTerritory B
                    ON A.TerritoryID = B.TerritoryID
         WHERE DATEPART(yyyy, A.OrderDate) = 2013
         GROUP BY  B.Name,
                   DATEPART(yyyy, A.OrderDate) ,
                   DATEPART(MM, A.OrderDate)
        HAVING ROUND(SUM(A.SubTotal),2) >= 750000
       ) X;
```



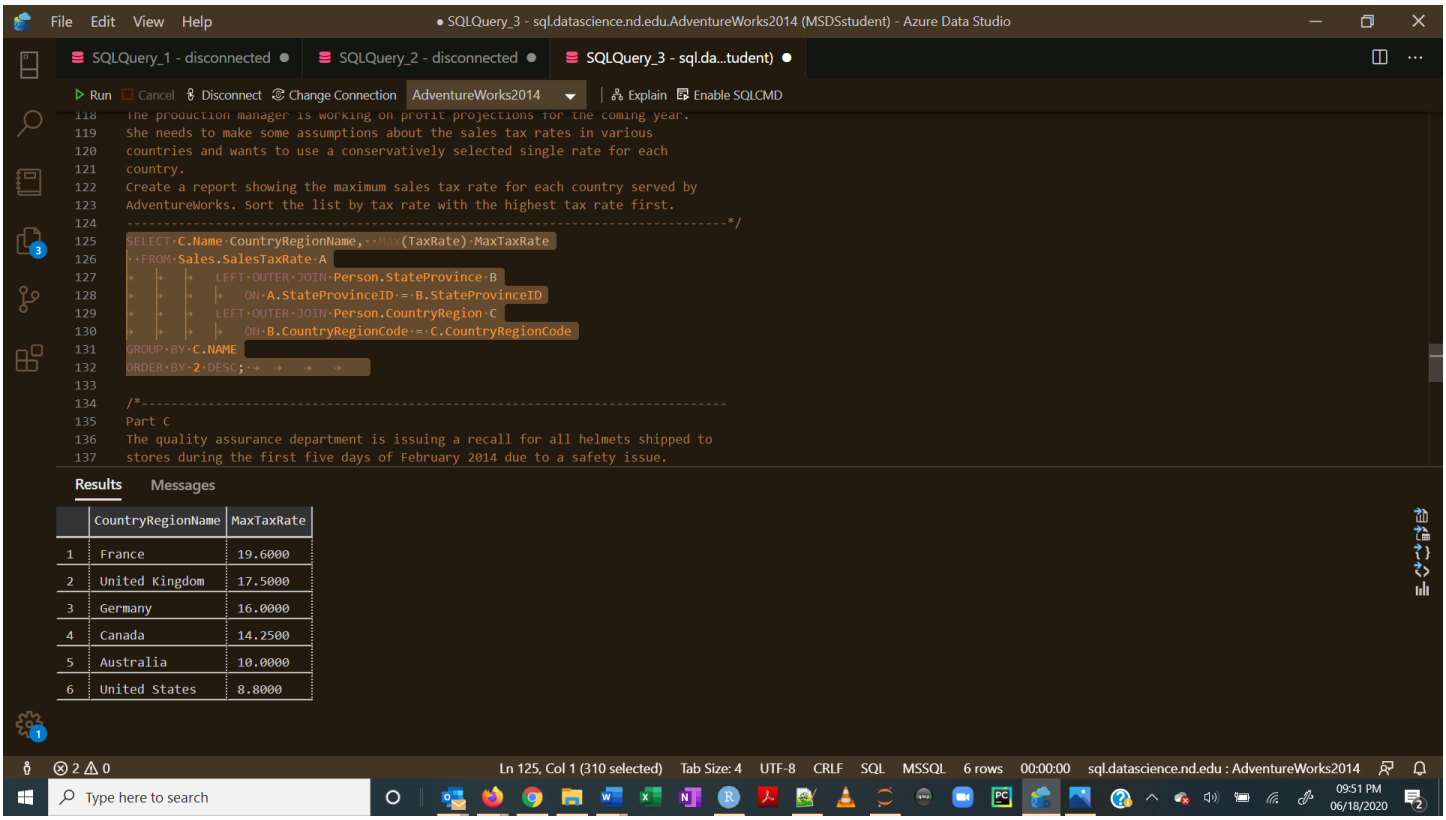
```
/*--------------------------------------------------------------------------
Part D
The sales manager would like to offer additional training to those territories
that did not achieve the $750,000 at least once in 2013. Create a list of
territories that did not achieve their target, sorted alphabetically. Do this by using set
theory operators, adding a few lines to your query from Part C.
--------------------------------------------------------------------------*/
SELECT DISTINCT
       Name TerritoryName
  FROM Sales.SalesTerritory
EXCEPT
SELECT DISTINCT TerritoryName
  FROM
     (
       SELECT B.Name TerritoryName,
              DATEPART(yyyy, A.OrderDate)        AS SalesYear,
              DATEPART(MM, A.OrderDate)    AS SalesMonth,
              FORMAT(CAST(ROUND(SUM(SubTotal),2) AS money), 'C2') SalesRevenue
         FROM Sales.SalesOrderHeader A
              LEFT OUTER JOIN Sales.SalesTerritory B
                   ON A.TerritoryID = B.TerritoryID
        WHERE DATEPART(yyyy, A.OrderDate) = 2013
        GROUP BY  B.Name,
                  DATEPART(yyyy, A.OrderDate) ,
                  DATEPART(MM, A.OrderDate)
       HAVING ROUND(SUM(A.SubTotal),2) >= 750000
      ) X
 ORDER BY 1;
```



```
/*-------------------------------------------------------------------------------
Problem 2
Your next assignment is to work with the production manager on a series of
questions that he has about your product catalog.
-------------------------------------------------------------------------------
/*-------------------------------------------------------------------------------
Part A
The production manager is considering dropping some products and would like to
generate a report that lists finished goods products where the total sales
volume is under 50 units.
Create a report showing the product name and the total number of units ordered.
-------------------------------------------------------------------------------
SELECT B.Name            AS ProductName,
       SUM(A.OrderQty)   AS SoldUnits
  FROM Sales.SalesOrderDetail A
         LEFT OUTER JOIN Production.Product B ON A.ProductID = B.ProductID
 GROUP BY B.Name
HAVING SUM(A.OrderQty) < 50
 ORDER BY B.Name;
```



```
/*-------------------------------------------------------------------------------
Part B
The production manager is working on profit projections for the coming year.
She needs to make some assumptions about the sales tax rates in various
countries and wants to use a conservatively selected single rate for each
country.
Create a report showing the maximum sales tax rate for each country served by
AdventureWorks. Sort the list by tax rate with the highest tax rate first.
-------------------------------------------------------------------------------*/
SELECT C.Name        AS CountryRegionName,
       MAX(TaxRate) AS MaxTaxRate
  FROM Sales.SalesTaxRate A
          LEFT OUTER JOIN Person.StateProvince B
               ON A.StateProvinceID    = B.StateProvinceID
          LEFT OUTER JOIN Person.CountryRegion C
               ON B.CountryRegionCode = C.CountryRegionCode
 GROUP BY C.NAME
 ORDER BY 2 DESC;
```



```
/*-------------------------------------------------------------------------------
Part C
The quality assurance department is issuing a recall for all helmets shipped to
stores during the first five days of February 2014 due to a safety issue.
They are already handling the notification of individual consumers where records
are available. The sales department needs your help to follow up with stores that
received the recalled helmets. Produce a listing of stores that received helmets
shipped during the recall period. Your result should have two columns:
        StoreName
        TerritoryName
Sort the list in alphabetical order, first by territory name and then by store
name. Each store should only appear once in the list, regardless
of the number of shipments they received.
-------------------------------------------------------------------------------*/
SELECT DISTINCT
         C.Name TerritoryName,
         D.Name StoreName
  FROM Sales.SalesOrderHeader A
                  INNER JOIN Sales.Customer         B ON A.CustomerID    = B.CustomerID
                  INNER JOIN Sales.SalesTerritory   C ON B.TerritoryID   = C.TerritoryID
                  INNER JOIN Sales.Store            D ON B.StoreID       = D.BusinessEntityID
            INNER JOIN Sales.SalesOrderDetail       E ON A.SalesOrderID = E.SalesOrderID
            INNER JOIN Production.Product           F ON E.ProductID     = F.ProductID
WHERE A.ShipDate  BETWEEN '2014-02-01' AND '2014-02-05'
      AND UPPER(F.Name) LIKE '%HELMET%'
ORDER BY 1, 2;
```