-- 2nd Last Digit of NUID: 5, 6, or 7

-- Your NUID:001567554

-- Your Name:Jiangying Zhong

-- Question 1 (4 points)

/\* Rewrite the following query to present the same data in a horizontal format,

as listed below, using the SQL PIVOT command. Use the sample format for

formatting purposes only.

Please use AdventureWorks2008R2 for this question. \*/

select datepart(mm, OrderDate) Month,

TerritoryID,

cast(sum(TotalDue) as int) as TotalSales

from Sales.SalesOrderHeader

where month(OrderDate) in (2,4,6,8,10,12) and TerritoryID between 1 and 5

group by TerritoryID, datepart(mm, OrderDate)

having sum(TotalDue) > 750000;

/\*

Month Territory 1 2 3 4 5

2 1775242 775651 2316676 806813

4 1210705 1817745

6 1333956 780329 2400791

8 2036028 878856 1068611 2527116 1012268

10 1066385 1746462

12 1605966 839737 2638675 790570

\*/

USE AdventureWorks2008R2;

-- Question 2 (5 points)

/\*

Using AdventureWorks2008R2, write a query to return the salesperson id,

number of unique products sold, highest order value, total sales amount,

and top 3 orders for each salesperson. Use TotalDue in SalesOrderHeader

when calculating the highest order value and total sales amount. The

top 3 orders have the 3 highest total order quantities. If there is a tie,

the tie must be retrieved. Exclude orders which don't have a salesperson

for this query. Return only the salespersons whose total sales were

greater than $9800000.

Return the order value and total sales as int. Sort the

returned data by SalesPersonID. The returned data should have a format

displayed below. Use the sample format for formatting purposes only.

\*/

/\*

SalesPersonID TotalUniqueProducts OrderValue TotalSales Orders

275 242 165029 10475367 47395, 46666, 46662

276 244 145742 11695019 51721, 47355, 57046

277 246 132728 11342386 51748, 53560, 47027

\*/

-- Question 3 (6 points)

/\* Given two tables as defined below \*/

CREATE TABLE Department

(DepartmentID INT PRIMARY KEY,

Name VARCHAR(50));

CREATE TABLE Employee

(EmployeeID INT PRIMARY KEY,

LastName VARCHAR(50),

FirsName VARCHAR(50),

Salary DECIMAL(10,2),

DepartmentID INT REFERENCES Department(DepartmentID),

TerminateDate DATE);

/\* There is a business rule that a department cannot have more than 25%

of the company employees in it or a department cannot have more than

60 employees.

Write a SINGLE table-level CHECK constraint based on a SINGLE Function

to implement the rule. \*/