Run the following script to crete tables in tempdb that will be used for exercises 1-5:

--------------------------------------------------------------------------------------------USE tempdb;

IF OBJECT\_ID('dbo.Orders', 'U') IS NOT NULL DROP TABLE dbo.Orders;

CREATE TABLE dbo.Orders

(

orderid INT NOT NULL,

orderdate DATETIME NOT NULL,

empid INT NOT NULL,

custid VARCHAR(5) NOT NULL,

qty INT NOT NULL,

CONSTRAINT PK\_Orders PRIMARY KEY(orderid)

);

INSERT INTO dbo.Orders(orderid, orderdate, empid, custid, qty)

SELECT 30001, '20070802', 3, 'A', 10 UNION ALL

SELECT 10001, '20071224', 2, 'A', 12 UNION ALL

SELECT 10005, '20071224', 1, 'B', 20 UNION ALL

SELECT 40001, '20080109', 2, 'A', 40 UNION ALL

SELECT 10006, '20080118', 1, 'C', 14 UNION ALL

SELECT 20001, '20080212', 2, 'B', 12 UNION ALL

SELECT 40005, '20090212', 3, 'A', 10 UNION ALL

SELECT 20002, '20090216', 1, 'C', 20 UNION ALL

SELECT 30003, '20090418', 2, 'B', 15 UNION ALL

SELECT 30004, '20070418', 3, 'C', 22 UNION ALL

SELECT 30007, '20090907', 3, 'D', 30

--------------------------------------------------------------------------------------------

IF OBJECT\_ID('dbo.EmpCustOrders', 'U') IS NOT NULL DROP TABLE dbo.EmpCustOrders;

CREATE TABLE dbo.EmpCustOrders(empid INT,A VARCHAR(5),B VARCHAR(5),C VARCHAR(5),D VARCHAR(5))

INSERT INTO dbo.EmpCustOrders(empid,A,B,C,D)

SELECT 1, NULL, 20, 34, NULL UNION ALL

SELECT 2, 52, 27, NULL, NULL UNION ALL

SELECT 3, 20, NULL, 22, 30

--------------------------------------------------------------------------------------------

Task#1.

Write a query pivoting order data, returning the total quantity for each employee (on rows) and customer (on cols)

Use MS SQL 2000 compatible query.

tables to use: tempdb.dbo.Orders

Output:

empid A B C D

----- ---- --- --- ----

1 NULL 20 34 NULL

...

Task#2.

The same task as in #1, but use MS SQL 2005 compatible query.

tables to use: tempdb.dbo.Orders

Output:

empid A B C D

----- ---- --- -- ----

1 NULL 20 34 NULL

...

Task#3.

Write a query to unpivot the data, returning a row for each employee and customer, along with the order quantity.

Use MS SQL 2000 compatible query.

tables involved: tempdb.dbo.EmpCustOrders

Output:

empid custid qty

------ ------ ------

1 B 20

1 C 34

2 A 52

2 B 27

3 A 20

3 C 22

3 D 30

Task#4.

The same task as in #3, but use MS SQL 2005 compatible query.

tables involved: tempdb.dbo.EmpCustOrders

Output:

empid custid qty

------- ------- -----

1 B 20

1 C 34

2 A 52

2 B 27

3 A 20

3 C 22

3 D 30

Task#5.

Write a query against the Orders table that returns a row for each employee, a column for each order year, and the count of orders for each employee and order year.

tables involved: tempdb.dbo.Orders

Output:

empid cnt2007 cnt2008 cnt2009

------ --------- -------- -----------

Task#6.

Write a query against the Sales.SalesOrderHeader table that computes for each customer order both a rank and a

dense rank, partitioned by CustomerID and ordered by SubTotal

Output:

CustomerID SalesOrderID SubTotal rnk drnk

----------- ------------ ----------- ----- -----

11000 51522 2341.97 1 1

11000 57418 2507.03 2 2

11000 43793 3399.99 3 3

.......

Task#7.

Using Sales.SalesOrderHeader table create a query that computes for each customer order both the difference

between the current SubTotal and the customer’s previous SubTotal and the

difference between the current SubTotal and the customer’s next SubTotal.

Output:

CustomerID SalesOrderID SubTotal diffprev diffnext

----------- ------------ ----------- --------- ----------

11000 43793 3399.99 3399.99 1058.02

11000 51522 2341.97 -1058.02 -165.06

11000 57418 2507.03 165.06 2507.03