Basic Databases – Report04

Wroclaw University of Technology, Date: November 23, 2018

Student:	Email:241018@student.pwr.edu.pl	Grade
Identifier	241018	
First name	Siamion	
Last name	Rondzel	

This package consists of 6 tasks. If you cannot solve a complete problem, try to give a partial answer. Remember to write your name, and identifier.

The tasks contain exercises to practice two subjects:

1. Retrieving Information from Many Tables using **common table expressions** **CTE**

Note: CTE facilitates solving of the following tasks

Source data: **Database: AdventureWorks2017**

Task 1

Create reports according to the following definition:

- **City, Number of orders**

Table: `Person.Address, …`**?**

- Create a list of cities with the number of completed orders
- Create a list of cities to which were supplied more than 500 orders

Solution

```
WITH CTE_First_1(City, CompletedOrders) AS(
SELECT P.City, Count(*) AS CompletedOrders
FROM Person.Address AS P, Sales.SalesOrderHeader AS S
WHERE P.AddressID = S.ShipToAddressID
GROUP BY P.City
)
SELECT * FROM CTE_First_1;
```

City	CompletedOrders
Cheltenham	79
Braunschweig	48
Reading	51

Rec:3/539

```
WITH CTE_First_1(City, CompletedOrders) AS(
SELECT P.City, Count(*) AS CompletedOrders
FROM Person.Address AS P, Sales.SalesOrderHeader AS S
WHERE P.AddressID = S.ShipToAddressID
GROUP BY P.City
)
SELECT * FROM CTE_First_1
WHERE CompletedOrders>500;
```

City	CompletedOrders
Paris	**575**
London	**756**

Rec:2/2

Task 2

Table: **?**

Prepare a report which presents a list of customers that have at least four orders in the following months: 5, 6, 7, 8, 9

The structures of reports:

CustId	Year	Month	NoOfOrders
11176	2013	7	6
11330	2014	5	4
11331	2013	7	6
11212	2013	7	4
11276	2014	5	4

Rec: 5/?

Solution

```
WITH CTE_2(CustomerID, Year, Month, NoOfOrders)
AS(
	SELECT CustomerID, YEAR(OrderDate) AS Year, MONTH(OrderDate) AS Month, COUNT(*) AS
NoOfOrders
	FROM Sales.SalesOrderHeader
	WHERE MONTH(OrderDate)>=5 AND MONTH(OrderDate)<=9
	GROUP BY CustomerID, YEAR(OrderDate), MONTH(OrderDate)
	HAVING COUNT(*)>=4
)
SELECT CustomerID, Year, Month, NoOfOrders
FROM CTE_2 AS C1
WHERE Month IN
(
	SELECT MAX(Month)
	FROM CTE_2 AS C2
	WHERE C2.CustomerID=C1.CustomerID
	GROUP BY CustomerID
) AND NoOfOrders IN(
	SELECT MAX(NoOfOrders)
	FROM CTE_2 AS C2
	WHERE C2.CustomerID=C1.CustomerID
	GROUP BY CustomerID
);
```

CustomerID	Year	Month	NoOfOrders
11176	2013	7	6
11330	2014	5	4
11331	2013	7	6
11212	2013	7	4
11142	2014	5	4

Rec:5/8

Task 3

Calculate the monthly number of orders based on order date for each customer separately from 2012 to 2014

The structures of reports:

- **custID, Name (Last Name, First name), Year, Month, Number of orders**

Solution

```
WITH CTE_Second(CustomerID, Year, Month, NoOfOrders)
AS(
SELECT CustomerID, YEAR(OrderDate) as Year, MONTH(OrderDate) AS Month, COUNT(*) AS
NoOfOrders
FROM Sales.SalesOrderHeader
WHERE YEAR(OrderDate)>=2012 AND YEAR(OrderDate)<=2014
GROUP BY CustomerID, YEAR(OrderDate), MONTH(OrderDate)
)
SELECT C2.CustomerID, P.FirstName + ', ' + P.LastName AS Name, C2.Year, C2.Month,
C2.NoOfOrders
FROM CTE_Second AS C2, Sales.Customer AS C, Person.Person AS P
WHERE C2.CustomerID=C.CustomerID AND C.PersonID=P.BusinessEntityID
ORDER BY C2.CustomerID ASC, Year ASC, Month ASC;
```

CustomerID	Name	Year	Month	NoOfOrders
11000	Jon, Yang	2013	6	1
11000	Jon, Yang	2013	10	1
11001	Eugene, Huang	2013	6	1

Rec: 3/29129

Task 4

Table: ?

- Create a list of customers and for each specifies a value Y or N depending on whether the number of his orders is ten times greater than the average number of customer orders

CustID	Valuable	NumberOfOrders
11019	Y	17
11078	Y	17
11091	Y	28
11142	Y	17
…	…	…

Solution

```
DECLARE @totalNoOfReports DECIMAL(10,2);
SET @totalNoOfReports = (SELECT COUNT(*) FROM Sales.SalesOrderHeader);
DECLARE @totalNoOfCustomers DECIMAL(10,2);
SET @totalNoOfCustomers = (SELECT COUNT(*) FROM Sales.Customer);
DECLARE @avgNoOfReports DECIMAL(10,2);
SET @avgNoOfReports = (@totalNoOfReports / @totalNoOfCustomers);
SELECT CustomerID,
```

```
	CASE
		WHEN COUNT(*)>10*@avgNoOfReports
		THEN 'Y'
		ELSE 'N'
	END AS Valuable, COUNT(*) AS NoOfOrders
FROM Sales.SalesOrderHeader
GROUP BY CustomerID;
```

CustomerID	Valuable	NoOfOrders
11090	**N**	**3**
11091	**Y**	**28**
11092	**N**	**3**

Rec: 3/19119

Task 5

Create report according to the following definition:
Table: ?

EmpID	Order year	CustID	Last name, first name	Total sum
279	2011	29624	Cantoni, Joseph	279655.51
…			…	
282	2011	29722	D'sa, Reuben	183329.02
…			…	
276	2011	29710	Diaz, Brenda	170660.07
…			…	

Solution

```
WITH CTE_4(SalesPersonID, CustomerID, Year, TotalSum)
AS(
	SELECT SalesPersonID,CustomerID, YEAR(OrderDate) AS Year, SUM(TotalDue) AS TotalSum
	FROM Sales.SalesOrderHeader
	WHERE YEAR(OrderDate) = 2011 AND SalesPersonID IS NOT NULL
	GROUP BY CustomerID, SalesPersonID, YEAR(OrderDate)
)
SELECT SalesPersonID, Year, C.CustomerID, P.LastName + ', ' + P.FirstName AS 'Last name,
first name', TotalSum AS 'Total sum'
FROM CTE_4, Sales.Customer AS C, Person.Person AS P
WHERE CTE_4.CustomerID = C.CustomerID AND C.PersonID = P.BusinessEntityID;
```

EmpID	Year	CustomerID	Last name, first name	Total sum
277	2011	29486	Abercrombie, Kim	94633,3601
275	2011	29487	Acevedo, Humberto	42360,8949
279	2011	29484	Achong, Gustavo	4560,2864

Rec:3/209

Task 6

Count for each customer the difference of his/her orders between June and July 2014
The structures of reports:

- **custID, Name (Last Name, First name), Gender, Difference of orders June/July 2014**

Solution

```
CREATE VIEW JuneOrdersTable AS
SELECT CustomerID, COUNT(*) AS JuneOrders
FROM Sales.SalesOrderHeader
WHERE YEAR(OrderDate)=2014 AND MONTH(OrderDate)=6
GROUP BY CustomerID

CREATE VIEW JulyOrdersTable AS
SELECT CustomerID, COUNT(*) AS JulyOrders
FROM Sales.SalesOrderHeader
WHERE YEAR(OrderDate)=2014 AND MONTH(OrderDate)=7
GROUP BY CustomerID

SELECT T1.CustomerID, P.LastName + ', ' + P.FirstName AS 'Name (Last Name, First name)',
              COALESCE(T1.JuneOrders-T2.JulyOrders, T1.JuneOrders) AS 'Difference in
orders'
FROM JuneOrdersTable AS T1
LEFT JOIN JulyOrdersTable AS T2 ON T1.CustomerID=T2.CustomerID
JOIN Sales.Customer AS C ON C.CustomerID = T1.CustomerID
JOIN Person.Person AS P ON P.BusinessEntityID = C.PersonID
ORDER BY CustomerID;
```

CustomerID	Name (Last Name, First name)	Difference in orders
11013	Jenkins, Ian	1
11019	Lal, Luke	2
11023	Edwards, Seth	1

Rec:3/898

Please do not forget about the conclusions being a summary of considered problems and proposed solutions!

Conclusion:

In this lab we were presented with 6 problems, regarding building SQL queries, and most of these problems were aimed at practicing Common Table Expressions (CTEs) as a way to reference the resulting table multiple times in the same statement and as a way to substitute views.