Common SQL

EXTENDED SQL FUNCTIONS

SE0607 – 24/05/2012

TABLE OF CONTENT

1. SOME COMMON FUNCTIONS

The following lists some common functions you can use in SQL Server.

Function Name	Description
CAST	Explicitly converts an expression of one data type to another. CAST and CONVERT provide similar functionality. **Example**: When you want to convert a string to a number: `SELECT cast('1234' as int)`
CONVERT	Explicitly converts an expression of one data type to another. CAST and CONVERT provide similar functionality. **Example**: When you want to convert a string to a number: `SELECT convert(int, '1234')`
GETDATE	Returns the current system date and time in the standard internal format for **datetime** values **Example**: `SELECT GETDATE()`

LEN	Returns the number of characters, instead of the number of bytes, of the given string expression, excluding trailing blanks.
	
	Example: Check to see the length of a constant string:
	
	`SELECT LEN('abcde')`
	
	Example: Get all employees whose lengths are greater than 5
	
	`SELECT FirstName, LEN(FirstName) as [Length Of First Name]`
	`FROM Employees`
	`WHERE LEN(FirstName) > 5`

DATEPART	The DATEPART and DATENAME functions produce the specified part of a **datetime** value such as the year, quarter, day, or hour, as either an integer or an ASCII string. Because **smalldatetime** is accurate only to the minute, when a **smalldatetime** value is used with either of these functions, the seconds and milliseconds returned are always zero
	
	Example:
	
	`SELECT`
	`orderid,`
	`DATEPART(year, OrderDate) as [Order Year],`
	`DATEPART(month, OrderDate) as [Order Month],`
	`DATEPART(day, OrderDate) as [Order Day]`
	
	`FROM Orders`

DATEADD	Returns a new DT_DBTIMESTAMP value after adding a number that represents a date or time interval to the specified datepart in a date
	
	Example:
	
	`SELECT`
	`dateadd(year, 5, GETDATE()) as [5 years after],`
	`dateadd(month, 5, GETDATE()) as [5 months after],`
	`dateadd(day, 5, GETDATE()) as [5 days after]`
	`FROM Orders`

DATEDIFF	Returns the number of date and time boundaries crossed between two specified dates. The *datepart* parameter identifies which date and time boundaries to compare.
	
	Example: How long between '1-jan-10' and '1-jan-11' (by year, month, day)
	
	`SELECT`
	`DATEDIFF(year, '1-jan-10', '1-jan-11'),`
	`DATEDIFF(month, '1-jan-10', '1-jan-11'),`
	`DATEDIFF(day, '1-jan-10', '1-jan-11')`
	`FROM Orders`

REPLACE	Replaces all occurrences of the second specified string expression in the first string expression with a third expression.
	
	Example: Replace all 'd' character by 'x':
	
	`SELECT REPLACE('abcdddd', 'd','x')`
	
	Example: Replace all 'd' character by 'x':
	
	`SELECT REPLACE(FirstName, 'd','x')`
	`FROM Employees`

UPPER	Returns a character expression with lowercase character data converted to uppercase.
	
	Example:
	
	`SELECT UPPER('abcd')`
	
	Example:
	
	`SELECT UPPER(firstName)`
	`From Employees`

LOWER	Returns a character expression after converting uppercase character data to lowercase.
	
	Example:
	
	`SELECT LOWER('abcd')`
	
	Example:
	
	`SELECT LOWER(firstName)`
	`From Employees`

STR	Returns character data converted from numeric data. **Example**: `SELECT 'I love the number: ' + STR(100)`

REVERSE	Returns the reverse of a character expression. **Example**: `SELECT REVERSE('abcd')`
LTRIM	Returns a character expression after it removes leading blanks. **Example**: `SELECT LTRIM('    abcd    ')`
RTRIM	Returns a character string after truncating all trailing blanks. **Example**: `SELECT RTRIM('    abcd    ')`
LEFT	Returns the left part of a character string with the specified number of characters. **Example**: `SELECT LEFT('1234abcd', 4)`
RIGHT	Returns the right part of a character string with the specified number of characters. **Example**: `SELECT RIGHT('1234abcd', 4)`
SUBSTRING	Returns part of a character, binary, text, or image expression **Example**: Return the substring with 6 characters from position 1 (‘I hate’): `SELECT SUBSTRING('I hate studying', 1, 6)`

<table>
<tr>
<td>ISNULL</td>
<td>Replaces NULL with the specified replacement value.

<u>Example</u>: For all employees: if their region is null, replace with ‘Has no region’

<pre>SELECT FirstName, LastName,
      ISNULL(Region, 'Has no region')
FROM Employees</pre>
</td>
</tr>
<tr>
<td>CASE</td>
<td>
<pre>CASE input_expression
     WHEN when_expression THEN
result_expression
          [...n]
     [
          ELSE else_result_expression
     ]
END</pre>
<u>Example</u>: For all employees: if their region is null, replace with ‘Has no region’:

<pre>SELECT FirstName, LastName,
      case when Region is not null then Region
           when Region is null then 'Has not region'
      end
FROM Employees</pre>
</td>
</tr>
</table>

@@IDENTITY

Example: Look at the Employees table:

```
CREATE TABLE Employees
(
      EmployeeID int IDENTITY(1,1) NOT NULL,
      LastName nvarchar(20) NOT NULL,
      FirstName nvarchar(10) NOT NULL
)
```

The column EmployeeID is set to "IDENTITY", this means that: the ID is automatically increased.

So, SQL Server will raise an error if you run the following statement:

```
INSERT INTO Employees(EmployeeID, LastName, FirstName)
VALUES (100, 'do', 'long')
```

To make the above 'insert' statement run successfully, you must rewrite:

```
INSERT INTO Employees(LastName, FirstName)
VALUES ('do', 'long')
```

But now, you dont know what EmployeeId is inserted into Employees table. To solve this problem, use @@IDENTITY as following (note that: you must run the following query in the same session with 'Insert' statement):

```
SELECT @@IDENTITY
```

2. EXERCISES

Exercise 1

Write a SELECT query to display Full name of all employees in lower-case as following:

Full name	titleOfCourtesy
davolio nancy	Ms.
fuller andrew	Dr.
leverling janet	Ms.
peacock margaret	Mrs.
buchanan steven	Mr.
suyama michael	Mr.
king robert	Mr.
callahan laura	Ms.
dodsworth anne	Ms.

Exercise 2

Write a SELECT query to display Full name of all employees in upper-case as following:

Full name
DAVOLIO NANCY
FULLER ANDREW
LEVERLING JANET
PEACOCK MARGARET
BUCHANAN STEVEN
SUYAMA MICHAEL
KING ROBERT
CALLAHAN LAURA
DODSWORTH ANNE

Exercise 3

Write a SELECT query to display all employees that are from United States as following:

EmployeeID	LastName	FirstName	Title	City	Country
1	Davolio	Nancy	Sales Representative	Seattle	USA
2	Fuller	Andrew	Vice President, Sales	Tacoma	USA
3	Leverling	Janet	Sales Representative	Kirkland	USA
4	Peacock	Margaret	Sales Representative	Redmond	USA
8	Callahan	Laura	Inside Sales Coordinator	Seattle	USA

Exercise 4

Write a SELECT query to display all customers that are from UK as following:

CustomerID	CompanyName	ContactName	ContactTitle	Country
AROUT	Around the Horn	Thomas Hardy	Sales Representative	UK
BSBEV	B's Beverages	Victoria Ashworth	Sales Representative	UK
CONSH	Consolidated Holdings	Elizabeth Brown	Sales Representative	UK
EASTC	Eastern Connection	Ann Devon	Sales Agent	UK
ISLAT	Island Trading	Helen Bennett	Marketing Manager	UK
NORTS	North/South	Simon Crowther	Sales Associate	UK
SEVES	Seven Seas Imports	Hari Kumar	Sales Manager	UK

Exercise 5

Write a SELECT query to display all customers that are from Mexico as following:

CustomerID	CompanyName	address	city	Country
ANATR	Ana Trujillo Emparedados y helados	Avda. de la Constitución 2222	México D.F.	Mexico
ANTON	Antonio Moreno Taquería	Mataderos 2312	México D.F.	Mexico
CENTC	Centro comercial Moctezuma	Sierras de Granada 9993	México D.F.	Mexico
PERIC	Pericles Comidas clásicas	Calle Dr. Jorge Cash 321	México D.F.	Mexico
TORTU	Tortuga Restaurante	Avda. Azteca 123	México D.F.	Mexico

Exercise 6

Write a SELECT query to display all customers that are from Sweden as following:

CustomerID	CompanyName	phone	address	city	Country
BERGS	Berglunds snabbköp	0921-12 34 65	Berguvsvägen 8	Luleå	Sweden
FOLKO	Folk och fä HB	0695-34 67 21	Åkergatan 24	Bräcke	Sweden

Exercise 7

In Products table, values of UnitsInStock tell you the total units in the inventory of every product. Write a SELECT query to display product id, product name, unit price of all products such that their total units in the inventory is between 5 and 10 as following:

productid	productName	Unitprice	UnitsInStock
8	Northwoods Cranberry Sauce	40.00	6
30	Nord-Ost Matjeshering	25.89	10
32	Mascarpone Fabioli	32.00	9
45	Rogede sild	9.50	5
49	Maxilaku	20.00	10
68	Scottish Longbreads	12.50	6

Exercise 8

In Products table, the value of UnitsOnOrder tells you the total ordered units for every product. Write a SELECT query to display product id, product name, unit price, re-order level… of all

products that have total ordered units between 60 and 100 as following:

ProductID	ProductName	unitprice	reorderlevel	UnitsOnOrder
3	Aniseed Syrup	10.00	25	70
31	Gorgonzola Telino	12.50	20	70
45	Rogede sild	9.50	15	70
48	Chocolade	12.75	25	70
49	Maxilaku	20.00	15	60
64	Wimmers gute Semmelknödel	33.25	30	80
66	Louisiana Hot Spiced Okra	17.00	20	100

Exercise 9

Write a SELECT query to display total orders of every employee in 1996 as following:

EmployeeID	LastName	FirstName	Title	year	total orders
1	Davolio	Nancy	Sales Representative	1996	26
2	Fuller	Andrew	Vice President, Sales	1996	16
3	Leverling	Janet	Sales Representative	1996	18
4	Peacock	Margaret	Sales Representative	1996	31
5	Buchanan	Steven	Sales Manager	1996	11
6	Suyama	Michael	Sales Representative	1996	15
7	King	Robert	Sales Representative	1996	11
8	Callahan	Laura	Inside Sales Coordinator	1996	19
9	Dodsworth	Anne	Sales Representative	1996	5

Exercise 10

Write a SELECT query to display total orders of every employee in 1996 as following:

EmployeeID	LastName	FirstName	Title	year	total orders
1	Davolio	Nancy	Sales Representative	1997	55
2	Fuller	Andrew	Vice President, Sales	1997	41
3	Leverling	Janet	Sales Representative	1997	71
4	Peacock	Margaret	Sales Representative	1997	81
5	Buchanan	Steven	Sales Manager	1997	18
6	Suyama	Michael	Sales Representative	1997	33
7	King	Robert	Sales Representative	1997	36
8	Callahan	Laura	Inside Sales Coordinator	1997	54
9	Dodsworth	Anne	Sales Representative	1997	19

Exercise 11

Write a SELECT query to display total orders of every employee in 1998 as following:

EmployeeID	LastName	FirstName	City	Country	total orders
1	Davolio	Nancy	Seattle	USA	42
2	Fuller	Andrew	Tacoma	USA	39
3	Leverling	Janet	Kirkland	USA	38
4	Peacock	Margaret	Redmond	USA	44
5	Buchanan	Steven	London	UK	13
6	Suyama	Michael	London	UK	19
7	King	Robert	London	UK	25
8	Callahan	Laura	Seattle	USA	31
9	Dodsworth	Anne	London	UK	19

Exercise 12

Write a SELECT query to display total orders of every employee from 1/1/1998 to 31/7/1998 as following:

EmployeeID	LastName	FirstName	hiredate	total orders
1	Davolio	Nancy	1992-05-01 00:00:00.000	42
2	Fuller	Andrew	1992-08-14 00:00:00.000	39
3	Leverling	Janet	1992-04-01 00:00:00.000	38
4	Peacock	Margaret	1993-05-03 00:00:00.000	44
5	Buchanan	Steven	1993-10-17 00:00:00.000	13
6	Suyama	Michael	1993-10-17 00:00:00.000	19
7	King	Robert	1994-01-02 00:00:00.000	25
8	Callahan	Laura	1994-03-05 00:00:00.000	31
9	Dodsworth	Anne	1994-11-15 00:00:00.000	19

Exercise 13

Write a SELECT query to display total orders of every employee from 1/1/1997 to 30/6/1997 as following:

EmployeeID	LastName	FirstName	hiredate	homephone	total orders
1	Davolio	Nancy	1992-05-01 00:00:00.000	(206) 555-9857	20
2	Fuller	Andrew	1992-08-14 00:00:00.000	(206) 555-9482	19
3	Leverling	Janet	1992-04-01 00:00:00.000	(206) 555-3412	35
4	Peacock	Margaret	1993-05-03 00:00:00.000	(206) 555-8122	36
5	Buchanan	Steven	1993-10-17 00:00:00.000	(71) 555-4848	7
6	Suyama	Michael	1993-10-17 00:00:00.000	(71) 555-7773	14
7	King	Robert	1994-01-02 00:00:00.000	(71) 555-5598	18
8	Callahan	Laura	1994-03-05 00:00:00.000	(206) 555-1189	28
9	Dodsworth	Anne	1994-11-15 00:00:00.000	(71) 555-4444	8

Exercise 14

Write a SELECT query to display total orders of every employee from 1/1/1997 to 30/6/1997 as following:

EmployeeID	LastName	FirstName	birthdate	homephone	total orders
1	Davolio	Nancy	1948-12-08 00:00:00.000	(206) 555-9857	20
2	Fuller	Andrew	1952-02-19 00:00:00.000	(206) 555-9482	19
3	Leverling	Janet	1963-08-30 00:00:00.000	(206) 555-3412	35
4	Peacock	Margaret	1937-09-19 00:00:00.000	(206) 555-8122	36
5	Buchanan	Steven	1955-03-04 00:00:00.000	(71) 555-4848	7
6	Suyama	Michael	1963-07-02 00:00:00.000	(71) 555-7773	14
7	King	Robert	1960-05-29 00:00:00.000	(71) 555-5598	18
8	Callahan	Laura	1958-01-09 00:00:00.000	(206) 555-1189	28
9	Dodsworth	Anne	1966-01-27 00:00:00.000	(71) 555-4444	8

Exercise 15

An order will be taxed 10% if its freight cost is larger than or equal to 100$.

Otherwise, an order will be taxed 5% if its freight cost is smaller than 100$.

Write a SELECT query to show the freight with taxes of orders placed between 1/8/1996 and 5/8/1996 as following:

orderid	OrderDay	OrderMonth	OrderYear	freight	tax	Freight with tax
10270	1	8	1996	136.54	10%	150.194000
10271	1	8	1996	4.54	10%	4.994000
10272	2	8	1996	98.03	10%	107.833000
10273	5	8	1996	76.07	10%	83.677000
10270	1	8	1996	136.54	5%	143.367000
10271	1	8	1996	4.54	5%	4.767000
10272	2	8	1996	98.03	5%	102.931500
10273	5	8	1996	76.07	5%	79.873500

Exercise 16

Write a SELECT query to display the full name, title of courtesy and sex for all employees such that:

- If title of courtesy is ‘Mr.’ then sex is set to ‘Male’
- If title of courtesy is ‘Ms.’ or ‘Mrs.’ then sex is set to ‘Female’

Full name	titleOfCourtesy	Sex
Buchanan Steven	Mr.	Male
Suyama Michael	Mr.	Male
King Robert	Mr.	Male
Davolio Nancy	Ms.	Female
Leverling Janet	Ms.	Female
Peacock Margaret	Mrs.	Female
Callahan Laura	Ms.	Female
Dodsworth Anne	Ms.	Female

Exercise 17

Write a SELECT query to display the full name, title of courtesy and sex for all employees such that:

- If title of courtesy is ‘Mr.’ or ‘Dr.’ then sex is set to ‘M’
- If title of courtesy is ‘Ms.’ or ‘Mrs.’ then sex is set to ‘F’

	full name	titleofcourtesy	sex
1	Fuller Andrew	Dr.	M
2	Buchanan Steven	Mr.	M
3	Suyama Michael	Mr.	M
4	King Robert	Mr.	M
5	Davolio Nancy	Ms.	F
6	Leverling Janet	Ms.	F
7	Peacock Margaret	Mrs.	F
8	Callahan Laura	Ms.	F
9	Dodsworth Anne	Ms.	F

Exercise 18

Write a SELECT query to display the full name, title of courtesy and sex for all employees such that:

- If title of courtesy is ‘Mr.’ then sex is set to ‘Male’
- If title of courtesy is ‘Ms.’ or ‘Mrs.’ then sex is set to ‘Female’
- If title of courtesy is not ‘Mr.’ and not ‘Mrs.’ and not ‘Ms.’ then sex is set to ‘Unknown’

	full name	titleofcourtesy	sex
1	Buchanan Steven	Mr.	Male
2	Suyama Michael	Mr.	Male
3	King Robert	Mr.	Male
4	Davolio Nancy	Ms.	Female
5	Leverling Janet	Ms.	Female
6	Peacock Margaret	Mrs.	Female
7	Callahan Laura	Ms.	Female
8	Dodsworth Anne	Ms.	Female
9	Fuller Andrew	Dr.	Unknown

Exercise 19

Write a SELECT query to display the full name, title of courtesy and sex for all employees such that:

- If title of courtesy is ‘Mr.’ then sex is set to 1
- If title of courtesy is ‘Ms.’ or ‘Mrs.’ then sex is set to 0
- If title of courtesy is not ‘Mr.’ and not ‘Mrs.’ and not ‘Ms.’ then sex is set to 2

	full name	titleofcourtesy	sex
1	Buchanan Steven	Mr.	1
2	Suyama Michael	Mr.	1
3	King Robert	Mr.	1
4	Davolio Nancy	Ms.	0
5	Leverling Janet	Ms.	0
6	Peacock Margaret	Mrs.	0
7	Callahan Laura	Ms.	0
8	Dodsworth Anne	Ms.	0
9	Fuller Andrew	Dr.	2

Exercise 20

Write a SELECT query to display the full name, title of courtesy and sex for all employees such that:

- If title of courtesy is ‘Mr.’ then sex is set to ‘M’
- If title of courtesy is ‘Ms.’ or ‘Mrs.’ then sex is set to ‘F’
- If title of courtesy is not ‘Mr.’ and not ‘Mrs.’ and not ‘Ms.’ then sex is set to ‘N/A’

	full name	titleofcourtesy	sex
1	Buchanan Steven	Mr.	M
2	Suyama Michael	Mr.	M
3	King Robert	Mr.	M
4	Davolio Nancy	Ms.	F
5	Leverling Janet	Ms.	F
6	Peacock Margaret	Mrs.	F
7	Callahan Laura	Ms.	F
8	Dodsworth Anne	Ms.	F
9	Fuller Andrew	Dr.	N/A

Exercise 21

Write a query to determine the revenues for all products from 1/7/1996 to 5/7/1996 as following (Revenue = Quantity * Unitprice). **Note**: The output must be order by category id and then product id.

CategoryID	CategoryName	ProductID	ProductName	day	month	year	Revenue
4	Dairy Products	11	Queso Cabrales	4	7	1996	168.00
4	Dairy Products	72	Mozzarella di Giovanni	4	7	1996	174.00
5	Grains/Cereals	42	Singaporean Hokkien Fried Mee	4	7	1996	98.00
7	Produce	14	Tofu	5	7	1996	167.40
7	Produce	51	Manjimup Dried Apples	5	7	1996	1696.00

Exercise 22

Write a query to display information about 7-days late orders and their employees (an order is called late if it is shipped after required date):

EmployeeID	LastName	FirstName	OrderID	OrderDate	RequiredDate	ShippedDate
1	Davolio	Nancy	10827	1998-01-12 00:00:00.000	1998-01-26 00:00:00.000	1998-02-06 00:00:00.000
2	Fuller	Andrew	10515	1997-04-23 00:00:00.000	1997-05-07 00:00:00.000	1997-05-23 00:00:00.000
2	Fuller	Andrew	10663	1997-09-10 00:00:00.000	1997-09-24 00:00:00.000	1997-10-03 00:00:00.000
4	Peacock	Margaret	10726	1997-11-03 00:00:00.000	1997-11-17 00:00:00.000	1997-12-05 00:00:00.000
6	Suyama	Michael	10423	1997-01-23 00:00:00.000	1997-02-06 00:00:00.000	1997-02-24 00:00:00.000
7	King	Robert	10777	1997-12-15 00:00:00.000	1997-12-29 00:00:00.000	1998-01-21 00:00:00.000
8	Callahan	Laura	10660	1997-09-08 00:00:00.000	1997-10-06 00:00:00.000	1997-10-15 00:00:00.000
9	Dodsworth	Anne	10828	1998-01-13 00:00:00.000	1998-01-27 00:00:00.000	1998-02-04 00:00:00.000
9	Dodsworth	Anne	10970	1998-03-24 00:00:00.000	1998-04-07 00:00:00.000	1998-04-24 00:00:00.000

Exercise 23

Write a query to display the names and telephone numbers of all employees and all customers satisfy the condition: all customers have names start with 'W'.

CompanyName	Phone
Wartian Herkku	981-443655
Wellington Importadora	(14) 555-8122
White Clover Markets	(206) 555-4112
Wilman Kala	90-224 8858
Wolski Zajazd	(26) 642-7012
Davolio Nancy	(206) 555-9857
Fuller Andrew	(206) 555-9482
Leverling Janet	(206) 555-3412
Peacock Margaret	(206) 555-8122
Buchanan Steven	(71) 555-4848
Suyama Michael	(71) 555-7773
King Robert	(71) 555-5598
Callahan Laura	(206) 555-1189
Dodsworth Anne	(71) 555-4444

Exercise 24

Write a query to display information about the customer that placed the order with Id equal to 10643 as following:

CustomerID	CompanyName	ContactName	ContactTitle
ALFKI	Alfreds Futterkiste	Maria Anders	Sales Representative

Exercise 25

Write a query to display the product ids, product names and total units ordered of all products that satisfy: the total units ordered must be greater than or equal to 1200 units.

	ProductID	ProductName	Total Ordered
1	56	Gnocchi di nonna Alice	1263
2	31	Gorgonzola Telino	1397
3	59	Raclette Courdavault	1496
4	60	Camembert Pierrot	1577

Exercise 26

Write a query to display the product ids, product names, supplier id, category id and total units ordered of all products that satisfy: the total units ordered must be greater than or equal to 1400 units.

ProductID	ProductName	SupplierID	CategoryID	Total ordered
60	Camembert Pierrot	28	4	1577
59	Raclette Courdavault	28	4	1496

Exercise 27

Write a query to display the categories that have maximum total product as following:

	CategoryID	CategoryName	Total products
	3	Confections	13

Exercise 28

Write a query to display the categories that have minimum total product as following:

	CategoryID	CategoryName	Total products
1	9	Demo1	0
2	10	Demo2	0

Exercise 29

Write a query to display the total record in Customer and Employees tables:

	Total records
1	100

Exercise 30

Write a query to display information about employees who have minimum total orders as following:

EmployeeID	LastName	FirstName	Title	total orders
5	Buchanan	Steven	Sales Manager	42

Exercise 31

Write a query to display information about employees who have maximum total orders as following:

	EmployeeID	LastName	FirstName	Title	Total_Orders
1	4	Peacock	Margaret	Sales Representative	156

Exercise 32

In Products table, the value of UnitsInStock tells you the total unit in the inventory for every product. Write a query to display information about products that have maximum total unit in inventory as following:

ProductID	ProductName	SupplierID	CategoryID	UnitsInStock
75	Rhönbräu Klosterbier	12	1	125

Exercise 33

In Products table, the value of UnitsInStock tells you the total unit in the inventory for every product. Write a query to display information about products that have minimum total unit in inventory as following:

	ProductID	ProductName	SupplierID	CategoryID	UnitsInStock
1	5	Chef Anton's Gumbo Mix	2	2	0
2	17	Alice Mutton	7	6	0
3	29	Thüringer Rostbratwurst	12	6	0
4	31	Gorgonzola Telino	14	4	0
5	53	Perth Pasties	24	6	0

Exercise 34

In Products table, the value of UnitsOnOrder tells you the total ordered unit for every product. Write a query to display information about products that have maximum total ordered unit as following:

ProductID	ProductName	SupplierID	CategoryID	UnitsOnOrder
66	Louisiana Hot Spiced Okra	2	2	100

Exercise 35

In Products table, the value of ReOrderLevel tells you the re-order level for every product. Write a query to display information about products that have maximum re-order level as following:

ProductID	ProductName	SupplierID	CategoryID	reorderlevel
11	Queso Cabrales	5	4	30
25	NuNuCa Nuß-Nougat-Creme	11	3	30
27	Schoggi Schokolade	11	3	30
40	Boston Crab Meat	19	8	30
50	Valkoinen suklaa	23	3	30
56	Gnocchi di nonna Alice	26	5	30
64	Wimmers gute Semmelknödel	12	5	30
70	Outback Lager	7	1	30

Exercise 36

Write a query to display the information about employees who have maximum total delayed-orders as following:

EmployeeID	LastName	FirstName	Delayed Orders
4	Peacock	Margaret	10

Exercise 37

Write a query to display the information about employees who have at least one delayed-order and have minimum total delayed-orders as following:

EmployeeID	LastName	FirstName	Delayed Orders
5	Buchanan	Steven	1

Exercise 38

Write a query to display product ids and product names of all products that have total ordered-units in three-highest level (top 3 to top 1) as following:

ProductID	ProductName	Total Ordered
31	Gorgonzola Telino	1397
59	Raclette Courdavault	1496
60	Camembert Pierrot	1577

Exercise 39

Write a query to display product ids and product names of all products that have total ordered-units in five-highest level (top 5 to top 1) as following:

ProductID	ProductName	Total Ordered
16	Pavlova	1158
56	Gnocchi di nonna Alice	1263
31	Gorgonzola Telino	1397
59	Raclette Courdavault	1496
60	Camembert Pierrot	1577