

HURTOWNIE DANYCH

Laboratorium 3

Maciej Kopiński 254578

Zad. 1. Wykorzystanie funkcji grupujących (rollup, cube, grouping sets)

1. Przygotować zestawienie przedstawiające, ile pieniędzy wydali klienci na zamówienia na przestrzeni poszczególnych lat.
Wykonaj zestawienie przy użyciu poleceń rollup, cube, grouping sets.
2. Przygotować zestawienie przedstawiające łączną kwotę zniżek z podziałem na kategorie, produkty oraz lata.

Zad. 2. Wykorzystanie funkcji okienkowych (over, over partition by, row_number, rank, dense_rank, ntile)

1. Dla kategorii 'Bikes' przygotuj zestawienie prezentujące procentowy udział kwot sprzedaży produktów tej kategorii w poszczególnych latach w stosunku do łącznej kwoty sprzedaży dla tej kategorii. W zadaniu wykorzystaj funkcje okna.

Wykonaj podobne zestawienia dla pozostałych kategorii.
2. Przygotuj zestawienie prezentujące narastającą liczbę zamówień wykonanych przez klienta w poszczególnych latach. W zadaniu wykorzystaj funkcje okna.
Ogranicz wynik do 10 najlepszych klientów.
3. Przygotuj zestawienie dla sprzedawców z podziałem na lata i miesiące prezentujące liczbę obsłużonych przez nich zamówień w ciągu roku, w ciągu roku narastająco oraz sumarycznie w obecnym i poprzednim miesiącu. W zadaniu wykorzystaj funkcje okna.
4. Przygotuj zestawienie dla kategorii produktów prezentujące sumę maksymalnych cen produktów w poszczególnych podkategoriach należących danej tej kategorii. W zadaniu wykorzystaj funkcje okna.
5. Przygotuj ranking klientów w zależności od liczby zakupionych produktów. Porównaj rozwiązania uzyskane przez funkcje rank i dense_rank.

6. Przygotuj ranking produktów w zależności od średniej liczby sprzedanych sztuk. Wyróżnij 3 (prawie równoliczne) grupy produktów: sprzedających się najlepiej, średnio i najsłabiej.

Rozwiązania:

Ad.1

1)

ROLLUP:

```
SELECT p.FirstName + ' ' + p.LastName AS Klient, YEAR(soh.OrderDate) AS Rok,
SUM(soh.TotalDue) AS Kwota
FROM Person.Person p
JOIN Sales.Customer c ON c.PersonID = p.BusinessEntityID
JOIN Sales.SalesOrderHeader soh ON soh.CustomerID = c.CustomerID
GROUP BY ROLLUP(p.FirstName + ' ' + p.LastName, YEAR(soh.OrderDate))
ORDER BY 1;
```

CUBE:

```
SELECT p.FirstName + ' ' + p.LastName AS Klient, YEAR(soh.OrderDate) AS Rok,
SUM(soh.TotalDue) AS Kwota
FROM Person.Person p
JOIN Sales.Customer c ON c.PersonID = p.BusinessEntityID
JOIN Sales.SalesOrderHeader soh ON soh.CustomerID = c.CustomerID
GROUP BY CUBE(p.FirstName + ' ' + p.LastName, YEAR(soh.OrderDate))
ORDER BY 1, 2, 3;
```

GROUPING SETS:

```
SELECT p.FirstName + ' ' + p.LastName AS Klient, YEAR(soh.OrderDate) AS Rok,
SUM(soh.TotalDue) AS Kwota
FROM Person.Person p
JOIN Sales.Customer c ON c.PersonID = p.BusinessEntityID
JOIN Sales.SalesOrderHeader soh ON soh.CustomerID = c.CustomerID
GROUP BY GROUPING SETS(
    (),
    (YEAR(soh.OrderDate)),
    (p.FirstName + ' ' + p.LastName),
    (p.FirstName + ' ' + p.LastName, YEAR(soh.OrderDate))
)
ORDER BY 1, 2, 3;
```

2)

```
SELECT pc.Name, p.Name, YEAR(soh.OrderDate), SUM(sod.UnitPriceDiscount * sod.UnitPrice
* sod.OrderQty)
FROM Production.ProductCategory pc
JOIN Production.ProductSubcategory ps ON ps.ProductCategoryID = pc.ProductCategoryID
JOIN Production.Product p ON p.ProductSubcategoryID = ps.ProductSubcategoryID
JOIN Sales.SalesOrderDetail sod ON sod.ProductID = p.ProductID
JOIN Sales.SalesOrderHeader soh ON soh.SalesOrderID = sod.SalesOrderID
GROUP BY GROUPING SETS(
    (pc.Name, p.Name, YEAR(soh.OrderDate)),
    (pc.Name, p.Name)
)
ORDER BY 1, 2, 3;
```

Ad.2

1)

```
SELECT DISTINCT pc.Name AS Nazwa, YEAR(soh.OrderDate) AS Rok, 100 * SUM(sod.LineTotal)
OVER(PARTITION BY YEAR(soh.OrderDate))/SUM(sod.LineTotal) OVER() AS Procent
FROM Production.ProductCategory pc
JOIN Production.ProductSubcategory ps ON ps.ProductCategoryID = pc.ProductCategoryID
JOIN Production.Product p ON p.ProductSubcategoryID = ps.ProductSubcategoryID
JOIN Sales.SalesOrderDetail sod ON sod.ProductID = p.ProductID
JOIN Sales.SalesOrderHeader soh ON soh.SalesOrderID = sod.SalesOrderID
WHERE pc.Name='Bikes'
ORDER BY 2;
```

```
SELECT DISTINCT pc.Name AS Nazwa, YEAR(soh.OrderDate) AS Rok, 100 * SUM(sod.LineTotal)
OVER(PARTITION BY YEAR(soh.OrderDate))/SUM(sod.LineTotal) OVER() AS Procent
FROM Production.ProductCategory pc
JOIN Production.ProductSubcategory ps ON ps.ProductCategoryID = pc.ProductCategoryID
JOIN Production.Product p ON p.ProductSubcategoryID = ps.ProductSubcategoryID
JOIN Sales.SalesOrderDetail sod ON sod.ProductID = p.ProductID
JOIN Sales.SalesOrderHeader soh ON soh.SalesOrderID = sod.SalesOrderID
WHERE pc.Name='Accessories'
ORDER BY 2;
```

```
SELECT DISTINCT pc.Name AS Nazwa, YEAR(soh.OrderDate) AS Rok, 100* SUM(sod.LineTotal)
OVER(PARTITION BY YEAR(soh.OrderDate))/SUM(sod.LineTotal) OVER() AS Procent
FROM Production.ProductCategory pc
JOIN Production.ProductSubcategory ps ON ps.ProductCategoryID = pc.ProductCategoryID
JOIN Production.Product p ON p.ProductSubcategoryID = ps.ProductSubcategoryID
JOIN Sales.SalesOrderDetail sod ON sod.ProductID = p.ProductID
JOIN Sales.SalesOrderHeader soh ON soh.SalesOrderID = sod.SalesOrderID
WHERE pc.Name='Clothing'
ORDER BY 2;
```

```
SELECT DISTINCT pc.Name AS Nazwa, YEAR(soh.OrderDate) AS Rok, 100 * SUM(sod.LineTotal)
OVER(PARTITION BY YEAR(soh.OrderDate))/SUM(sod.LineTotal) OVER() AS Procent
FROM Production.ProductCategory pc
JOIN Production.ProductSubcategory ps ON ps.ProductCategoryID = pc.ProductCategoryID
JOIN Production.Product p ON p.ProductSubcategoryID = ps.ProductSubcategoryID
JOIN Sales.SalesOrderDetail sod ON sod.ProductID = p.ProductID
JOIN Sales.SalesOrderHeader soh ON soh.SalesOrderID = sod.SalesOrderID
WHERE pc.Name='Components'
ORDER BY 2;
```

2)

```
SELECT DISTINCT p.FirstName + ' ' + p.LastName AS Klient, YEAR(soh.OrderDate) AS Rok,
COUNT(soh.SalesOrderID) OVER(PARTITION BY c.CustomerID ORDER BY YEAR(soh.OrderDate))
AS Liczba
FROM Person.Person p
JOIN Sales.Customer c ON c.PersonID = p.BusinessEntityID
JOIN Sales.SalesOrderHeader soh ON soh.CustomerID = c.CustomerID
WHERE c.CustomerID IN (
SELECT CustomerID FROM(
SELECT DISTINCT TOP 10 x.CustomerID, COUNT(x.CustomerID) OVER(PARTITION BY
x.CustomerID) y
FROM Sales.SalesOrderHeader x
ORDER BY 2 DESC) P)
ORDER BY 1, 2;
```

3)

```
SELECT "Imię i nazwisko", Rok, Miesiąc, "W miesiącu", "W roku", "W roku narastająco",
SUM("W miesiącu") OVER (PARTITION BY "Imię i nazwisko", Rok ORDER BY Miesiąc ROWS
BETWEEN 1 PRECEDING AND CURRENT ROW)
AS "Obecny i poprzedni miesiąc"
FROM(
SELECT DISTINCT p.FirstName + ' ' + p.LastName AS "Imię i nazwisko",
YEAR(soh.OrderDate) AS Rok, MONTH(soh.OrderDate) AS Miesiąc,
COUNT(soh.SalesOrderID) OVER (PARTITION BY p.FirstName + ' ' + p.LastName,
YEAR(soh.OrderDate), MONTH(soh.OrderDate)) AS "W miesiącu",
COUNT(soh.SalesOrderID) OVER (PARTITION BY p.FirstName + ' ' + p.LastName,
YEAR(soh.OrderDate)) AS "W roku",
COUNT(soh.SalesOrderID) OVER (PARTITION BY p.FirstName + ' ' + p.LastName,
YEAR(soh.OrderDate) ORDER BY MONTH(soh.OrderDate)) AS "W roku narastająco"
FROM Person.Person p
JOIN HumanResources.Employee e ON e.BusinessEntityID = p.BusinessEntityID
JOIN Sales.SalesPerson sp ON sp.BusinessEntityID = e.BusinessEntityID
JOIN Sales.SalesOrderHeader soh ON soh.SalesPersonID = sp.BusinessEntityID
) P;
```

4)

```
SELECT DISTINCT Kategoria, SUM("Cena max") OVER (PARTITION BY Kategoria) Suma
FROM(
SELECT DISTINCT cat.Name AS Kategoria, subcat.Name AS Podkategoria,
MAX(p.ListPrice) OVER (PARTITION BY cat.Name, subcat.Name) AS "Cena max"
FROM Production.ProductCategory cat
JOIN Production.ProductSubcategory subcat ON subcat.ProductCategoryID =
cat.ProductCategoryID
JOIN Production.Product p ON p.ProductSubcategoryID = subcat.ProductSubcategoryID
) P;
```

5)

RANK:

```
SELECT Klient, RANK() OVER(ORDER BY Liczba DESC) AS Pozycja
FROM (
SELECT DISTINCT p.FirstName + ' ' + p.LastName AS Klient, SUM(sod.OrderQty) OVER
(PARTITION BY p.FirstName + ' ' + p.LastName) Liczba
FROM Person.Person p
JOIN Sales.Customer c ON c.PersonID = p.BusinessEntityID
JOIN Sales.SalesOrderHeader soh ON soh.CustomerID = c.CustomerID
JOIN Sales.SalesOrderDetail sod ON sod.SalesOrderID = soh.SalesOrderID
) P
ORDER BY 2;
```

DENSE_RANK:

```
SELECT Klient, DENSE_RANK() OVER(ORDER BY Liczba DESC) AS Pozycja
FROM (
SELECT DISTINCT p.FirstName + ' ' + p.LastName AS Klient, SUM(sod.OrderQty) OVER
(PARTITION BY p.FirstName + ' ' + p.LastName) Liczba
FROM Person.Person p
JOIN Sales.Customer c ON c.PersonID = p.BusinessEntityID
JOIN Sales.SalesOrderHeader soh ON soh.CustomerID = c.CustomerID
JOIN Sales.SalesOrderDetail sod ON sod.SalesOrderID = soh.SalesOrderID
) P
ORDER BY 2;
```

6)

```
SELECT Produkt, Tile AS Grupa, RANK() OVER(PARTITION BY Tile ORDER BY Srednia DESC)
Pozycja, "Śreania liczba sztuk"
FROM (
SELECT Produkt, NTILE(3) OVER(ORDER BY Srednia DESC) AS Tile, Srednia, SUM(Srednia)
OVER (PARTITION BY Produkt) "Śreania liczba sztuk"
FROM(
SELECT DISTINCT p.Name Produkt, AVG(sod.OrderQty) OVER (PARTITION BY p.Name) AS
Srednia
FROM Production.Product p
JOIN Sales.SalesOrderDetail sod ON sod.ProductID = p.ProductID
) A) b;
```

Wnioski:

Ad.1

1)

	Klient	Rok	Kwota
1	NULL	NULL	123216786,1159
2	NULL	2011	14155699,525
3	NULL	2012	37675700,312
4	NULL	2013	48965887,9632
5	NULL	2014	22419498,3157
6	A. Leonetti	NULL	3400,8402
7	A. Leonetti	2013	1814,1819
8	A. Leonetti	2014	1586,6583
9	Aaron Adams	NULL	130,3458
10	Aaron Adams	2013	130,3458
11	Aaron Alexander	NULL	77,339
12	Aaron Alexander	2014	77,339
13	Aaron Allen	NULL	3756,989
14	Aaron Allen	2012	3756,989
15	Aaron Baker	NULL	1934,8329
16	Aaron Baker	2014	1934,8329
17	Aaron Bryant	NULL	148,0258
18	Aaron Bryant	2013	148,0258
19	Aaron Butler	NULL	16,5529

Z powyższego zrzutu ekranu możemy wyraźnie stwierdzić, że największa sprzedaż miała miejsce w 2013 roku. Trzeba oczywiście wziąć pod uwagę, że dane z 2014 roku są niekompletne.

2)

	Name	Name	Rok	Suma zniżek
1	Accessories	All-Purpose Bike Stand	NULL	0,00
2	Accessories	All-Purpose Bike Stand	2013	0,00
3	Accessories	All-Purpose Bike Stand	2014	0,00
4	Accessories	Bike Wash - Dissolver	NULL	111,5827
5	Accessories	Bike Wash - Dissolver	2013	83,8663
6	Accessories	Bike Wash - Dissolver	2014	27,7164
7	Accessories	Cable Lock	NULL	23,78
8	Accessories	Cable Lock	2012	20,30
9	Accessories	Cable Lock	2013	3,48
10	Accessories	Fender Set - Mountain	NULL	0,00
11	Accessories	Fender Set - Mountain	2013	0,00
12	Accessories	Fender Set - Mountain	2014	0,00
13	Accessories	Hitch Rack - 4-Bike	NULL	2341,044
14	Accessories	Hitch Rack - 4-Bike	2013	1950,444
15	Accessories	Hitch Rack - 4-Bike	2014	390,60
16	Accessories	HL Mountain Tire	NULL	0,00
17	Accessories	HL Mountain Tire	2013	0,00
18	Accessories	HL Mountain Tire	2014	0,00
19	Accessories	HL Road Tire	NULL	0,00

Możemy zauważyć, że największe sumy udzielonych zniżek wystąpiły w 2013 roku – wynika to z faktu najwyższej sprzedaży właśnie w tym roku.

Ad.2

1)

	Nazwa	Rok	Procent		Nazwa	Rok	Procent		Nazwa	Rok	Procent		Nazwa	Rok	Procent
1	Bikes	2011	12.620706	1	Accessories	2011	1.636759	1	Clothing	2011	1.699163	1	Components	2011	5.415530
2	Bikes	2012	30.623514	2	Accessories	2012	8.052983	2	Clothing	2012	26.200262	2	Components	2012	32.880553
3	Bikes	2013	38.316302	3	Accessories	2013	53.064935	3	Clothing	2013	50.349836	3	Components	2013	47.556797
4	Bikes	2014	18.439476	4	Accessories	2014	37.245321	4	Clothing	2014	21.750737	4	Components	2014	14.147119

Najwyższy udział w sprzedaży wyglądał tak samo w przypadku każdej kategorii – najwyższy wynik w roku 2013, najniższy w 2011, następnie w 2014 i 2012 (jedyna różnica jest widoczna w przypadku akcesoriów – udział w roku 2014 był większy niż w roku 2011 i 2012).

2)

	Klient	Rok	Liczba
1	Ashley Henderson	2013	9
2	Ashley Henderson	2014	27
3	Charles Jackson	2013	14
4	Charles Jackson	2014	27
5	Dalton Perez	2013	15
6	Dalton Perez	2014	28
7	Fernando Bames	2013	13
8	Fernando Bames	2014	27
9	Hailey Patterson	2013	15
10	Hailey Patterson	2014	27
11	Henry Garcia	2013	16
12	Henry Garcia	2014	27
13	Jason Griffin	2013	15
14	Jason Griffin	2014	27
15	Jennifer Simmons	2013	11
16	Jennifer Simmons	2014	27
17	Mason Roberts	2013	15
18	Mason Roberts	2014	28
19	Nancy Chapman	2013	9

Tutaj także widzimy, że większa część zakupionych przez klienta produktów była zakupiona w 2013 roku. Widzimy także, że klienci, którzy kupowali najwięcej, kupowali głównie w latach 2013 oraz 2014.

3)

	Imię i nazwisko	Rok	Miesiąc	W miesiącu	W roku	W roku narastająco	Obecny i poprzedni miesiąc
31	David Campbell	2012	8	6	63	40	9
32	David Campbell	2012	9	7	63	47	13
33	David Campbell	2012	10	4	63	51	11
34	David Campbell	2012	11	6	63	57	10
35	David Campbell	2012	12	6	63	63	12
36	David Campbell	2013	1	3	72	3	3
37	David Campbell	2013	2	6	72	9	9
38	David Campbell	2013	3	7	72	16	13
39	David Campbell	2013	4	1	72	17	8
40	David Campbell	2013	5	6	72	23	7
41	David Campbell	2013	6	7	72	30	13
42	David Campbell	2013	7	6	72	36	13
43	David Campbell	2013	8	7	72	43	13
44	David Campbell	2013	9	7	72	50	14
45	David Campbell	2013	10	7	72	57	14
46	David Campbell	2013	11	8	72	65	15
47	David Campbell	2013	12	7	72	72	15
48	David Campbell	2014	1	7	26	7	7
49	David Campbell	2014	3	14	26	21	21

Widzimy, że pracownicy wykazywali największą sprzedaż w roku 2013. Dla powyższego przykładu można także powiedzieć, że pracownik sprzedał w 2013 roku większość produktów z 2013 w drugiej połowie roku.

4)

	Kategoria	Suma
1	Accessories	663,88
2	Bikes	9362,33
3	Clothing	408,94
4	Components	5539,77

Suma maksymalnych cen w poszczególnych kategoriach pokazuje nam, że najwyższe ceny mają produkty należące do kategorii 'Bikes', następnie 'Components', 'Accessories', a na końcu 'Clothing'.

5)

	Klient	Pozycja
40	Lindsey Camacho	40
41	Stacey Cereghino	40
42	Thomas Armstrong	42
43	Andrew Dixon	43
44	Michael Graff	43
45	Lowell Graham	45
46	Jauna Elson	46
47	Mike Choi	47
48	Min Su	48
49	James Hendergart	49
50	Aidan Delaney	50
51	Stefan Delmarco	50
52	James McCoy	52
53	Kathleen Garza	53
54	Valerie Hendricks	54
55	Christopher Beck	55
56	James Glynn	56
57	Jeffrey Kurtz	57
58	Elizabeth Sullivan	58

	Klient	Pozycja
40	Lindsey Camacho	39
41	Stacey Cereghino	39
42	Thomas Armstrong	40
43	Andrew Dixon	41
44	Michael Graff	41
45	Lowell Graham	42
46	Jauna Elson	43
47	Mike Choi	44
48	Min Su	45
49	James Hendergart	46
50	Aidan Delaney	47
51	Stefan Delmarco	47
52	James McCoy	48
53	Kathleen Garza	49
54	Valerie Hendricks	50
55	Christopher Beck	51
56	James Glynn	52
57	Jeffrey Kurtz	53
58	Elizabeth Sullivan	54

Na powyższych zrzutach ekranu możemy zobaczyć różnicę w funkcjach RANK oraz DENSE_RANK – brak ciągłości dla funkcji RANK w przeciwieństwie do DENSE_RANK.

6)

	Produkt	Grupa	Pozycja	Średnia liczba sztuk
1	Full-Finger Gloves, L	1	1	9
2	Full-Finger Gloves, M	1	2	6
3	Classic Vest, S	1	2	6
4	Women's Mountain Shorts, S	1	4	5
5	ML Headset	1	4	5
6	Mountain Bike Socks, M	1	4	5
7	Men's Sports Shorts, M	1	7	4
8	Women's Tights, L	1	7	4
9	Racing Socks, L	1	7	4
10	Short-Sleeve Classic Jersey, XL	1	7	4
11	Women's Mountain Shorts, L	1	7	4
12	Minipump	1	7	4
13	Women's Tights, S	1	7	4
14	Long-Sleeve Logo Jersey, L	1	7	4
15	Cable Lock	1	7	4
16	Men's Bib-Shorts, M	1	7	4
17	Classic Vest, M	1	7	4
18	Men's Bib-Shorts, S	1	18	3
19	Men's Sports Shorts, S	1	18	3

Rękawiczki w rozmiarze L są kupowane średnio w największych ilościach w poszczególnym zamówieniu przez klientów.

Zadanie na zajęciach:

```
SELECT DISTINCT e.Gender, DATEPART(q, soh.OrderDate) Kwartał, MONTH(soh.OrderDate)
Miesiąc,
SUM(sod.OrderQty) OVER (PARTITION BY e.Gender, DATEPART(q, soh.OrderDate) ORDER BY
MONTH(soh.OrderDate)) AS Suma,
SUM(sod.OrderQty) OVER (PARTITION BY MONTH(soh.OrderDate), e.Gender)
FROM HumanResources.Employee e
JOIN Sales.SalesPerson sp ON sp.BusinessEntityID = e.BusinessEntityID
JOIN Sales.SalesOrderHeader soh ON soh.SalesPersonID = sp.BusinessEntityID
JOIN Sales.SalesOrderDetail sod ON sod.SalesOrderID = soh.SalesOrderID
ORDER BY 2, 3, 4;
```

Wnioski:

	Gender	Kwartal	Miesiąc	Suma	W miesiącu
1	F	1	1	6884	6884
2	M	1	1	7348	7348
3	F	1	2	9875	2991
4	M	1	2	11031	3683
5	F	1	3	22560	12685
6	M	1	3	26528	15497
7	M	2	4	4041	4041
8	F	2	4	4304	4304
9	F	2	5	18241	13937
10	M	2	5	18563	14522
11	F	2	6	30177	11936
12	M	2	6	32838	14275
13	F	3	7	12396	12396
14	M	3	7	13321	13321
15	F	3	8	19477	7081
16	M	3	8	21810	8489
17	F	3	9	27365	7888
18	M	3	9	32594	10784
19	M	4	10	10498	10498

Widzimy, że mężczyźni kupują więcej produktów niż kobiety. Można także zauważyć, że najwyższa sprzedaż występuje w ciepłych miesiącach.