Hurtownie danych  
Laboratorium Czw 11:15  
  
Lista 3  
  
Kajetan Pynka 254495

**Zad 1.1a**

SELECT ISNULL(P.FirstName + ' ' + P.LastName, '') "Klient", ISNULL(STR(YEAR(SOH.DueDate)), '') "Rok",

    SUM(SOH.TotalDue) "Kwota"

    FROM Sales.SalesOrderHeader SOH

    JOIN Sales.Customer C ON C.CustomerID=SOH.CustomerID

    JOIN Person.Person P ON P.BusinessEntityID=C.PersonID

    GROUP BY GROUPING SETS (

        (YEAR(SOH.DueDate)),

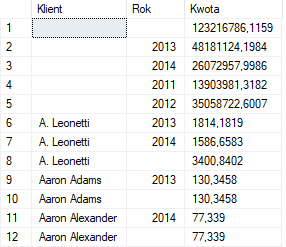
        (P.FirstName + ' ' + P.LastName, YEAR(SOH.DueDate)),

        (P.FirstName + ' ' + P.LastName),

        ()

    )

    ORDER BY 1;



**Zad 1.1b**

SELECT ISNULL(P.FirstName + ' ' + P.LastName, '') "Klient", ISNULL(STR(YEAR(SOH.DueDate)), '') "Rok",

    SUM(SOH.TotalDue) "Kwota"

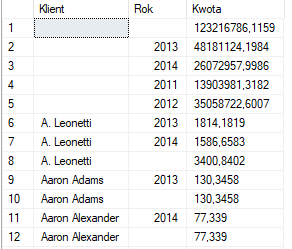
    FROM Sales.SalesOrderHeader SOH

    JOIN Sales.Customer C ON C.CustomerID=SOH.CustomerID

    JOIN Person.Person P ON P.BusinessEntityID=C.PersonID

    GROUP BY CUBE (YEAR(SOH.DueDate), P.FirstName + ' ' + P.LastName)

    ORDER BY 1;

****

**Zad 1.1c**

SELECT ISNULL(P.FirstName + ' ' + P.LastName, '') "Klient", ISNULL(STR(YEAR(SOH.DueDate)), '') "Rok",

    SUM(SOH.TotalDue) "Kwota"

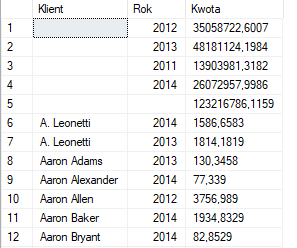
    FROM Sales.SalesOrderHeader SOH

    JOIN Sales.Customer C ON C.CustomerID=SOH.CustomerID

    JOIN Person.Person P ON P.BusinessEntityID=C.PersonID

    GROUP BY ROLLUP (YEAR(SOH.DueDate), P.FirstName + ' ' + P.LastName)

    ORDER BY 1;

****

**Zad 1.2**

SELECT PC.Name "Kategoria", P.Name "Produkt", ISNULL(STR(YEAR(SOH.DueDate)), '') "Rok",

    SUM(SOD.UnitPrice \* SOD.UnitPriceDiscount \* SOD.OrderQty) "Kwota"

    FROM Sales.SalesOrderHeader SOH

    JOIN Sales.SalesOrderDetail SOD ON SOH.SalesOrderID=SOD.SalesOrderID

    JOIN Production.Product P ON P.ProductID=SOD.ProductID

    JOIN Production.ProductSubcategory PSC ON PSC.ProductSubcategoryID=P.ProductSubcategoryID

    JOIN Production.ProductCategory PC ON PC.ProductCategoryID=PSC.ProductCategoryID

    GROUP BY GROUPING SETS (

        (PC.Name, P.Name, YEAR(SOH.DueDate)),

        (PC.Name, P.Name)

    ) ORDER BY 1, 2;

****

**Zad 2.1a**

SELECT PC.Name, STR(YEAR(SOH.OrderDate)) "Rok",

        SUM(SOD.UnitPrice \* SOD.OrderQty \* (1 - SOD.UnitPriceDiscount)) OVER (PARTITION BY YEAR(SOH.OrderDate))

            / SUM(SOD.UnitPrice \* SOD.OrderQty \* (1 - SOD.UnitPriceDiscount)) OVER()\*100 "Procent"

    FROM Sales.SalesOrderHeader SOH

    JOIN Sales.SalesOrderDetail SOD ON SOH.SalesOrderID=SOD.SalesOrderID

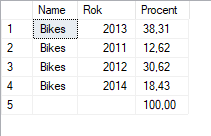
    JOIN Production.Product P ON P.ProductID=SOD.ProductID

    JOIN Production.ProductSubcategory PSC ON P.ProductSubcategoryID=PSC.ProductSubcategoryID

    JOIN Production.ProductCategory PC ON PC.ProductCategoryID=PSC.ProductCategoryID

    WHERE PC.Name='Bikes'

    UNION SELECT '', '', 100;

****

**Zad 2.1b**

SELECT PC.Name, STR(YEAR(SOH.OrderDate)) "Rok",

        SUM(SOD.UnitPrice \* SOD.OrderQty \* (1 - SOD.UnitPriceDiscount)) OVER (PARTITION BY YEAR(SOH.OrderDate))

            / SUM(SOD.UnitPrice \* SOD.OrderQty \* (1 - SOD.UnitPriceDiscount)) OVER()\*100 "Procent"

    FROM Sales.SalesOrderHeader SOH

    JOIN Sales.SalesOrderDetail SOD ON SOH.SalesOrderID=SOD.SalesOrderID

    JOIN Production.Product P ON P.ProductID=SOD.ProductID

    JOIN Production.ProductSubcategory PSC ON P.ProductSubcategoryID=PSC.ProductSubcategoryID

    JOIN Production.ProductCategory PC ON PC.ProductCategoryID=PSC.ProductCategoryID

    WHERE PC.Name='Accessories'

    UNION SELECT '', '', 100;

****

**Zad 2.1c**

SELECT PC.Name, STR(YEAR(SOH.OrderDate)) "Rok",

        SUM(SOD.UnitPrice \* SOD.OrderQty \* (1 - SOD.UnitPriceDiscount)) OVER (PARTITION BY YEAR(SOH.OrderDate))

            / SUM(SOD.UnitPrice \* SOD.OrderQty \* (1 - SOD.UnitPriceDiscount)) OVER()\*100 "Procent"

    FROM Sales.SalesOrderHeader SOH

    JOIN Sales.SalesOrderDetail SOD ON SOH.SalesOrderID=SOD.SalesOrderID

    JOIN Production.Product P ON P.ProductID=SOD.ProductID

    JOIN Production.ProductSubcategory PSC ON P.ProductSubcategoryID=PSC.ProductSubcategoryID

    JOIN Production.ProductCategory PC ON PC.ProductCategoryID=PSC.ProductCategoryID

    WHERE PC.Name='Clothing'

    UNION SELECT '', '', 100;

****

**Zad 2.1d**

SELECT PC.Name, STR(YEAR(SOH.OrderDate)) "Rok",

        SUM(SOD.UnitPrice \* SOD.OrderQty \* (1 - SOD.UnitPriceDiscount)) OVER (PARTITION BY YEAR(SOH.OrderDate))

            / SUM(SOD.UnitPrice \* SOD.OrderQty \* (1 - SOD.UnitPriceDiscount)) OVER()\*100 "Procent"

    FROM Sales.SalesOrderHeader SOH

    JOIN Sales.SalesOrderDetail SOD ON SOH.SalesOrderID=SOD.SalesOrderID

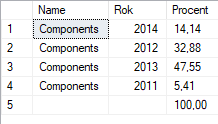
    JOIN Production.Product P ON P.ProductID=SOD.ProductID

    JOIN Production.ProductSubcategory PSC ON P.ProductSubcategoryID=PSC.ProductSubcategoryID

    JOIN Production.ProductCategory PC ON PC.ProductCategoryID=PSC.ProductCategoryID

    WHERE PC.Name='Components'

    UNION SELECT '', '', 100;

****

**Zad 2.2**

SELECT P.FirstName + ' ' + P.LastName "Klient", YEAR(SOH.DueDate) "Rok",

        MAX(COUNT(SOH.SalesOrderID)) OVER(PARTITION BY SOH.CustomerID, YEAR(SOH.DueDate)

            ORDER BY SOH.CustomerID, YEAR(SOH.DueDate) ROWS UNBOUNDED PRECEDING) "Suma transakcji"

    FROM Sales.SalesOrderHeader SOH

    JOIN Sales.Customer C ON C.CustomerID=SOH.CustomerID

    JOIN Person.Person P ON P.BusinessEntityID=C.PersonID

    GROUP BY P.FirstName + ' ' + P.LastName, YEAR(SOH.DueDate), SOH.CustomerID

    ORDER BY 1,2;

****

**Zad 2.3**

SELECT "Imię i nazwisko", "Rok", "Miesiąc", "W miesiącu", "W roku",

    MAX("W roku narastająco") "W roku narastająco" ,

    ISNULL(LAG("W miesiącu") OVER (ORDER BY "Imię i nazwisko", "Rok", "Miesiąc"), 0) + "W miesiącu" "Obecny i poprzedni miesiąc"

    FROM (SELECT P.FirstName + ' ' + P.LastName "Imię i nazwisko",

        YEAR(SOH.OrderDate) "Rok", MONTH(SOH.OrderDate) "Miesiąc",

        COUNT(SOH.SalesOrderID) OVER (PARTITION BY SOH.SalesPersonID, YEAR(SOH.OrderDate), MONTH(SOH.OrderDate)) "W miesiącu",

        COUNT(SOH.SalesOrderID) OVER (PARTITION BY SOH.SalesPersonID, YEAR(SOH.OrderDate)) "W roku",

        COUNT(SOH.SalesOrderID)

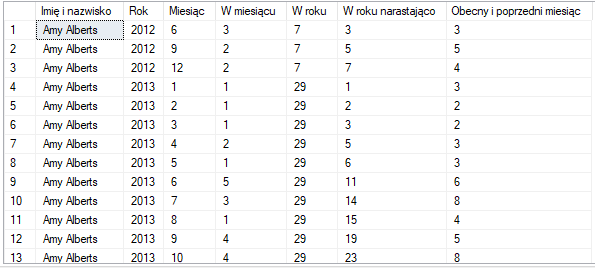
    OVER (PARTITION BY SOH.SalesPersonID, YEAR(SOH.OrderDate) ORDER BY SOH.SalesPersonID, YEAR(SOH.OrderDate) ROWS BETWEEN UNBOUNDED PRECEDING AND CURRENT ROW) "W roku narastająco"

    FROM Sales.SalesOrderHeader SOH JOIN Sales.SalesPerson SP ON SP.BusinessEntityID = SOH.SalesPersonID

    JOIN Person.Person P ON P.BusinessEntityID = SP.BusinessEntityID) INSIDE

GROUP BY "Imię i nazwisko", "Rok", "Miesiąc", "W miesiącu", "W roku"

ORDER BY 1,2,3;

****

**Zad 2.4**

SELECT "Kategoria", SUM("Kwota") "Suma maksymalnych" FROM (

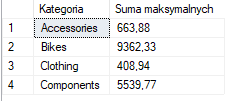
SELECT DISTINCT PC.Name "Kategoria", MAX(P.ListPrice) OVER(PARTITION BY PSC.ProductSubcategoryID) "Kwota"

    FROM Production.ProductCategory PC

    JOIN Production.ProductSubcategory PSC ON PSC.ProductCategoryID=PC.ProductCategoryID

    JOIN Production.Product P ON P.ProductSubcategoryID=PSC.ProductSubcategoryID) INSIDE

    GROUP BY "Kategoria";

****

**Zad 2.5a**

SELECT RANK() OVER(ORDER BY COUNT(SOD.OrderQty) DESC) "Ranga",

    P.FirstName + ' ' + P.LastName "Imię i nazwisko", COUNT(SOD.OrderQty) "Liczba transakcji"

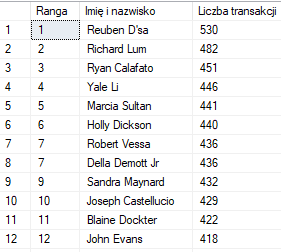
    FROM Sales.SalesOrderHeader SOH

    JOIN Sales.SalesOrderDetail SOD ON SOH.SalesOrderID=SOD.SalesOrderID

    JOIN Sales.Customer C ON C.CustomerID=SOH.CustomerID

    JOIN Person.Person P ON P.BusinessEntityID=C.PersonID

    GROUP BY P.FirstName + ' ' + P.LastName;

****

**Zad 2.5b**

SELECT DENSE\_RANK() OVER(ORDER BY COUNT(SOD.OrderQty) DESC) "Ranga",

    P.FirstName + ' ' + P.LastName "Imię i nazwisko", COUNT(SOD.OrderQty) "Liczba transakcji"

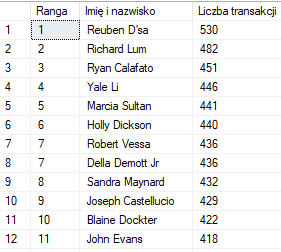
    FROM Sales.SalesOrderHeader SOH

    JOIN Sales.SalesOrderDetail SOD ON SOH.SalesOrderID=SOD.SalesOrderID

    JOIN Sales.Customer C ON C.CustomerID=SOH.CustomerID

    JOIN Person.Person P ON P.BusinessEntityID=C.PersonID

    GROUP BY P.FirstName + ' ' + P.LastName;

****

**Zad 2.6**

SELECT P.Name "Nazwa produktu", AVG(SOD.OrderQty) "Średnia liczba sztuk",

    CASE NTILE(3) OVER(ORDER BY AVG(SOD.OrderQty) DESC)

        WHEN 1 THEN 'Najlepiej'

        WHEN 2 THEN 'Średnio'

        WHEN 3 THEN 'Najsłabiej'

    END "Ranga sprzedaży"

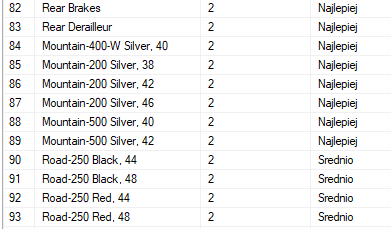
    FROM Sales.SalesOrderHeader SOH

    JOIN Sales.SalesOrderDetail SOD ON SOD.SalesOrderID=SOH.SalesOrderID

    JOIN Production.Product P ON P.ProductID=SOD.ProductID

    GROUP BY P.Name

    ORDER BY 2 DESC;

****

**Wnioski:**

* Funkcje grupujące pozwalają nam w bardzo precyzyjny sposób dobrać dane, po których chcemy pogrupować wynik. Korzystając z GROUPING SETS możemy np. określić by w wyniku znalazły się rekordy pogrupowane po wszystkich kolumnach, tylko po jednej czy dwóch wybranych kolumnach.
* Funkcje okienkowe pozwalają nam zaimplementować w bardzo prosty i czytelny sposób pewne narastające wartości wśród danych grup jak i pewne rankingi, które możemy przypisać rekordom. Należy jednak rozróżniać funkcje okienkowe od agregujących, ponieważ ich wpływ na strukturę ostatecznego wyniku jest zgoła odmienny.