# Zajęcia 2.

*Zdanie SELECT c.d. Złączenia tabel. Zapytania z funkcjami agregującymi: COUNT(\*), COUNT(Region), COUNT(DISTINCT Country), AVG, MIN, MAX, SUM.*

Proszę wypisać numer zamówienia i datę zamówienia (OrderID i OrderDate) z tabeli z zamówieniami (Orders) oraz identyfikator klienta, jego nazwę i kraj (pola CustomerID, CompanyName, Country z tabeli Customers). Proszę połączyć tabele wykorzystując klauzulę WHERE.

select o.OrderID, o.OrderDate, c.CustomerID, c.CompanyName, c.Country

from Orders as o, Customers as c

where o.CustomerID = c.CustomerID;

Proszę zmodyfikować poprzednie zadanie tak, by złączenie tabel było zrealizowane przy pomocy operatora JOIN w klauzuli FROM.

select o.OrderID, o.OrderDate, c.CustomerID, c.CompanyName, c.Country

from Orders as o

inner join Customers as c

on o.CustomerID = c.CustomerID;

Proszę wypisać wszystkie dane z tabeli [Order Details] oraz Identyfikator, nazwę i bieżącą cenę produktu (kolumny ProductID, ProductName i UnitPrice z tabeli Products).

select od.\*, p.ProductName, p.UnitPrice

from [Order Details] as od

inner join Products as p

on od.ProductID = p.ProductID;

Proszę zmodyfikować rozwiązanie poprzedniego zadania tak, by wyświetlana była też data zamówienia oraz by między numerem zamówienia a datą wyświetlone były dane klienta – jego identyfikator i nazwa (pola CustomerID i CompanyName z tabeli Customers). Wymaga to złączenia kilku tabel.

select od.OrderID, c.CustomerID, c.CompanyName,

cast(o.OrderDate as date) as OrderDate,

od.ProductID, od.UnitPrice, od.Quantity,

od.Discount, p.ProductName, p.UnitPrice

from [Order Details] as od

inner join Products as p

on od.ProductID = p.ProductID

inner join Orders as o

on od.OrderID = o.OrderID

inner join Customers as c

on o.CustomerID = c.CustomerID;

Proszę wypisać wszystkich klientów, którzy nigdy jeszcze nic nie zamawiali. Chcemy wyświetlić tylko ich identyfikatory i nazwy (CustomerID oraz CompanyName w tabeli Customers). Klient, który nic nie zamawiał to taki klient, którego identyfikatora nie ma na żadnym zamówieniu w tabeli Orders.

Proszę wykorzystać LEFT JOIN.

select c.CustomerID, c.CompanyName

from Customers as c

left join Orders as o

on c.CustomerID = o.CustomerID

where o.OrderID is null;

Proszę wypisać ilu jest klientów (tabela Customers). Należy wykorzystać funkcję agregującą COUNT.

select count(\*) as TotalQty

from Customers;

Proszę zmodyfikować wyniki poprzedniego zapytania tak, by policzyć tylko klientów z Niemiec.

select count(\*) as TotalQty

from Customers

where Country = N'Germany';

Proszę zmodyfikować wyniki poprzedniego zapytania tak, by policzyć klientów z Niemiec i Austrii.

select count(\*) as TotalQty

from Customers

where Country in (N'Germany', N'Austria');

Proszę dla każdego kraju (Country) wypisać ilu mamy klientów z tego kraju. Dane są w tabeli Customers. Dane powinny być posortowane rosnąco według nazwy kraju.

select Country, count(\*) as TotalQty

from Customers

group by Country

order by Country asc;

Proszę zmodyfikować rozwiązanie poprzedniego zadania tak, by dane były posortowane malejąco według liczby klientów, a jeśli z pewnych krajów jest tyle samo klientów, to rekordy te powinny być posortowane rosnąco według nazwy kraju.

select Country, count(\*) as TotalQty

from Customers

group by Country

order by TotalQty desc, Country asc;

Proszę zmodyfikować rozwiązanie poprzedniego zadania tak, by wyświetlane były tylko kraje, z których mamy więcej niż pięciu klientów.

select Country, count(\*) as TotalQty

from Customers

group by Country

having count(\*) > 5

order by TotalQty desc, Country asc;

Proszę wypisać, ilu mamy klientów z każdego kraju i miasta.

select Country, City, count(\*) as TotalQty

from Customers

group by Country, City

order by Country asc, TotalQty desc;

Proszę wypisać, ilu mamy klientów z każdego kraju i miasta, przy czym chcemy zliczać tylko tych klientów, którzy mają nazwę rozpoczynającą się od C lub A (nazwa klienta jest w kolumnie CompanyName)

select Country, City, count(\*) as TotalQty

from Customers

where CompanyName like N'[AC]%'

group by Country, City

order by Country asc, TotalQty desc;

Weźmy pod uwagę tabelę Products. Proszę wypisać dla każdej kategorii (wystarczy CategoryID) ile jest w niej produktów (należy wykorzystać funkcję agregującą COUNT), jaka jest najmniejsza cena jednostkowa produktu (UnitPrice) w kategorii, największa cena jednostkowa produktu, średnia cena jednostkowa oraz ile w sumie jest jednostek towarów w magazynie (należy zsumować zawartość kolumny UnitsInStock).

select CategoryID,

count(\*) as TotalQty,

min(UnitPrice) as MinPrice,

max(UnitPrice) as MaxPrice,

avg(UnitPrice) as AvgPrice,

sum(UnitsInStock) as SumUnits

from Products

group by CategoryID;

Proszę wypisać, ile jest produktów, których cena jednostkowa zawiera się w przedziale od 20 do 30 (łącznie z 20 i 30).

select count(\*) as TotalQty

from Products

where UnitPrice between 20 and 30;

Proszę zmodyfikować rozwiązanie poprzedniego zadania tak, by wynik był podany dla każdej kategorii (CategoryID).

select CategoryID, count(\*) as TotalQty

from Products

where UnitPrice between 20 and 30

group by CategoryID;

Proszę zmodyfikować rozwiązanie poprzedniego zadania tak, by grupę rekordów stanowiły wszystkie rekordy, których pierwsza litera nazwy jest taka sama.

select left(ProductName, 1), count(\*) as TotalQty

from Products

where UnitPrice between 20 and 30

group by left(ProductName, 1);

Proszę zmodyfikować rozwiązanie poprzedniego zadania tak, wyświetlone zostały dane dotyczące tylko tych kategorii, w których średnia cena jednostkowa jest większa niż 30 a mniejsza niż 50.

select CategoryID, count(\*) as TotalQty

from Products

group by CategoryID

having avg(UnitPrice) > 30 and avg(UnitPrice) < 50;

Proszę na podstawie tabeli [Order Details] wypisać jaka jest kwota na każdym zamówieniu (to ma być podsumowanie wszystkich pozycji szczegółowych każdego zamówienia).

select OrderID, round(sum(UnitPrice \* Quantity \* (1 - Discount)), 2) as OrderSum

from [Order Details]

group by OrderID;

# 

Proszę na podstawie tabeli [Order Details] oraz Orders wypisać jaka jest suma kwot na wszystkich zamówieniach w każdym roku.

select year(o.OrderDate) as OrderYear,

round(sum(od.UnitPrice \* od.Quantity \* (1 - od.Discount)), 2) as OrderSum

from [Order Details] as od

inner join Orders as o

on od.OrderID = o.OrderID

group by year(o.OrderDate)

order by OrderYear asc;

# 

Proszę na podstawie tabeli [Order Details] oraz Orders wypisać jaka jest suma kwot na wszystkich zamówieniach każdego klienta. Zestaw powinien zawierać dwie kolumny – CustomerID oraz suma kwot.

select o.CustomerID,

round(sum(od.UnitPrice \* od.Quantity \* (1 - od.Discount)), 2) as OrderSum

from [Order Details] as od

inner join Orders as o

on od.OrderID = o.OrderID

group by o.CustomerID

order by o.CustomerID asc;

# 

Proszę zmodyfikować rozwiązanie poprzedniego zadania tak, by oprócz identyfikatora klienta wyświetlana była też jego nazwa (tj. kolumna CompanyName z tabeli Customers). Zestaw powinien zawierać trzy kolumny – CustomerID, CompanyName oraz suma kwot.

select o.CustomerID, c.CompanyName,

round(sum(od.UnitPrice \* od.Quantity \* (1 - od.Discount)), 2) as OrderSum

from [Order Details] as od

inner join Orders as o

on od.OrderID = o.OrderID

inner join Customers as c

on o.CustomerID = c.CustomerID

group by o.CustomerID, c.CompanyName

order by o.CustomerID asc;

# 

Proszę zmodyfikować rozwiązanie poprzedniego zadania tak, by dane były posortowane malejąco według sum kwot.

select o.CustomerID, c.CompanyName,

round(sum(od.UnitPrice \* od.Quantity \* (1 - od.Discount)), 2) as OrderSum

from [Order Details] as od

inner join Orders as o

on od.OrderID = o.OrderID

inner join Customers as c

on o.CustomerID = c.CustomerID

group by o.CustomerID, c.CompanyName

order by OrderSum desc;

Proszę zmodyfikować rozwiązanie poprzedniego zadania tak, by dane były posortowane malejąco według sum kwot i w zestawie było tylko 3 rekordy (TOP 3).

select top 3 o.CustomerID, c.CompanyName,

round(sum(od.UnitPrice \* od.Quantity \* (1 - od.Discount)), 2) as OrderSum

from [Order Details] as od

inner join Orders as o

on od.OrderID = o.OrderID

inner join Customers as c

on o.CustomerID = c.CustomerID

group by o.CustomerID, c.CompanyName

order by OrderSum desc;

# 

Proszę zmodyfikować rozwiązanie zadania 23 tak, by w zestawie wynikowym były dane tylko tych klientów, dla których suma kwot była większa niż 100 000.

select o.CustomerID, c.CompanyName,

round(sum(od.UnitPrice \* od.Quantity \* (1 - od.Discount)), 2) as OrderSum

from [Order Details] as od

inner join Orders as o

on od.OrderID = o.OrderID

inner join Customers as c

on o.CustomerID = c.CustomerID

group by o.CustomerID, c.CompanyName

having sum(od.UnitPrice \* od.Quantity \* (1 - od.Discount)) > 100000

order by OrderSum desc;