

Grupowanie danych

- Ograniczanie rozmiaru zbioru wynikowego (TOP n)
- Użycie funkcji agregujących
- Grupowanie danych - klauzula GROUP BY
- Generowanie wartości zagregowanych

Wyświetlanie początkowych wierszy

- Wyświetlanie tylko pierwszych n wierszy zbioru wynikowego
- Określenie zakresu wartości w klauzuli ORDER BY
- fraza WITH TIES powoduje, że zwracane są dodatkowo wszystkie elementy o wartościach takich jak element ostatni

```
SELECT TOP 5 orderid, productid, quantity  
FROM [order details]  
ORDER BY quantity DESC
```

```
SELECT TOP 5 WITH TIES orderid, productid, quantity  
FROM [order details]  
ORDER BY quantity DESC
```

Użycie funkcji agregujących

Funkcja agregująca	Opis
AVG	Średnia wartości w wyrażeniu numerycznym
COUNT	Liczba wartości w wyrażeniu
COUNT (*)	Liczba wybranych wierszy
MAX	Największa wartość w wyrażeniu
MIN	Najmniejsza wartość w wyrażeniu
SUM	Totalna wartość w wyrażeniu numerycznym
STDEV	Statystyczne odchylenie wszystkich wartości
STDEVP	Statystyczne odchylenie dla populacji
VAR	Statystyczna wariancja dla wszystkich wartości
VARP	Statystyczna wariancja dla wszystkich wartości w populacji

Count

- Większość funkcji agregujących ignoruje wartości Null
- Funkcja COUNT(*) zlicza wiersze

```
USE northwind
SELECT COUNT (*)
  FROM employees
GO
```

```
USE northwind
SELECT COUNT(reportsto)
  FROM employees
GO
```

Przykłady

- **Policz średnią cenę jednostkową dla wszystkich produktów w tabeli products**

```
SELECT AVG(unitprice)
FROM products
```

- **Zsumuj wszystkie wartości w kolumnie quantity w tabeli order details (dla wierszy w których wartość productid = 1)**

```
SELECT SUM(quantity)
FROM [order details]
WHERE productid = 1
```

Ćwiczenie

1. **Podaj liczbę produktów o cenach mniejszych niż 10\$ lub większych niż 20\$**
2. **Podaj maksymalną cenę produktu dla produktów o cenach poniżej 20\$**
3. **Podaj maksymalną i minimalną i średnią cenę produktu dla produktów o produktach sprzedawanych w butelkach ('bottle')**
4. **Wypisz informację o wszystkich produktach o cenie powyżej średniej**
5. **Podaj sumę/wartość zamówienia o numerze 10250**

Grupowanie danych - GROUP BY

- Użycie GROUP BY
- Użycie GROUP BY z klauzulą HAVING

Użycie klauzuli GROUP BY

```
SELECT productid, orderid, quantity
FROM orderhist
```

```
SELECT productid
, SUM(quantity) AS total_quantity
FROM orderhist
GROUP BY productid
```

productid	orderid	quantity
1	1	5
1	2	10
2	1	10
2	2	25
3	1	15
3	2	30

Tylko wiersze
spełniające klauzulę
WHERE są
grupowane

productid	total_quantity
1	15
2	35
3	45

productid	total_quantity
2	35

```
SELECT productid
, SUM(quantity) AS total_quantity
FROM orderhist
WHERE productid = 2
GROUP BY productid
```


Przykład

- Napisz polecenie, które zwraca informacje o zamówieniach z tablicy order details. Zapytanie ma grupować i wyświetlać identyfikator każdego produktu a następnie obliczać ogólną zamówioną ilość. Ogólna ilość jest sumowana funkcją agregującą SUM i wyświetlana jako jedna wartość dla każdego produktu.

```
SELECT productid, SUM(quantity) AS total_quantity  
FROM [order details]  
GROUP BY productid
```

Ćwiczenia


1. **Podaj maksymalną cenę zamawianego produktu dla każdego zamówienia**
2. **Posortuj zamówienia wg maksymalnej ceny produktu**
3. **Podaj maksymalną i minimalną cenę zamawianego produktu dla każdego zamówienia**
4. **Podaj liczbę zamówień dostarczanych przez poszczególnych spedytorów (przewoźników)**
5. **Który z spedytorów był najaktywniejszy w 1997 roku**

GROUP BY z klauzulą HAVING

```
SELECT productid, orderid  
       ,quantity  
FROM orderhist
```

productid	orderid	quantity
1	1	5
1	2	10
2	1	10
2	2	25
3	1	15
3	2	30

```
SELECT productid, SUM(quantity)  
       AS total_quantity  
FROM orderhist  
GROUP BY productid  
HAVING SUM(quantity)>=30
```



productid	total_quantity
2	35
3	45

Przykład

- Wyświetl listę identyfikatorów produktów i ilość dla tych produktów, których zamówiono ponad 1200 jednostek

```
SELECT productid, SUM(quantity) AS total_quantity  
FROM [order details]  
GROUP BY productid  
HAVING SUM(quantity)>1200
```

Ćwiczenia

1. Wyświetl zamówienia dla których liczba pozycji zamówienia jest większa niż 5
2. Wyświetl klientów dla których w 1998 roku zrealizowano więcej niż 8 zamówień (wyniki posortuj malejąco wg łącznej kwoty za dostarczenie zamówień dla każdego z klientów)

Generowanie wartości zagregowanych

- Użycie GROUP BY z operatorem ROLLUP
- Użycie GROUP BY z operatorem CUBE

Użycie klauzuli GROUP BY z operatorem ROLLUP

```
SELECT productid, orderid, SUM(quantity) AS total_quantity
FROM orderhist
GROUP BY productid, orderid
WITH ROLLUP
ORDER BY productid, orderid
```

productid	orderid	total_quantity
NULL	NULL	95
1	NULL	15
1	1	5
1	2	10
2	NULL	35
2	1	10
2	2	25
3	NULL	45
3	1	15
3	2	30

Suma całkowita
Sumuje wiersze tylko dla productid 1
Szczegółowa wartość dla productid 1, orderid 1
Szczegółowa wartość productid 1, orderid 2
Sumuje wiersze tylko dla productid 2
Szczegółowa wartość dla productid 2, orderid 1
Sumuje wiersze tylko dla productid 3
Szczegółowa wartość dla productid 3, orderid 1
Szczegółowa wartość dla productid 3, orderid 2

Przykład

- Przykład zwraca informacje o zamówieniach z tablicy order details.
- Zapytanie zawiera instrukcję select z klauzulą GROUP BY bez operatora ROLLUP.
- Przykład ten zwraca listę ogólnej ilości zamawianej dla każdego produktu w każdym zamówieniu, dla zamówień z orderid mniejszym niż 10250

```
SELECT orderid, productid, SUM(quantity) AS total_quantity  
FROM [order details]  
WHERE orderid < 10250  
GROUP BY orderid, productid  
ORDER BY orderid, productid
```


Przykład

- **Przykład dodaje operator ROLLUP do poprzedniego wyrażenia. Zbiór wynikowy zawiera ogólną ilość dla:**
 - każdego produktu w każdym zamówieniu
 - wszystkich produktów dla każdego zamówienia
 - wszystkich produktów dla wszystkich zamówień

```
SELECT orderid, productid, SUM(quantity) AS total_quantity  
FROM [order details]  
WHERE orderid < 10250  
GROUP BY orderid, productid  
WITH ROLLUP  
ORDER BY orderid, productid
```

GROUP BY z operatorem CUBE

```
USE northwind
SELECT productid,orderid, SUM(quantity) AS total_quantity
FROM orderhist
GROUP BY productid,orderid
WITH CUBE
ORDER BY productid,orderid
```

Operator CUBE
produkuje więcej
podsumowujących
wartości niż
operator ROLLUP

productid	orderid	total_quantity
NULL	NULL	95
NULL	1	30
NULL	2	65
1	NULL	15
1	1	5
1	2	10
2	NULL	35
2	1	10
2	2	25
3	NULL	45
3	1	15
3	2	30

Suma całkowita
Sumuje wszystkie wiersze dla orderid 1
Sumuje wszystkie wiersze dla orderid 2
Sumuje tylko wiersze dla productid 1
Szczegółowa wartość dla productid 1, orderid 1
Szczegółowa wartość dla productid 1, orderid 2
Sumuje tylko wiersze dla productid 2
Szczegółowa wartość dla productid 2, orderid 1
Szczegółowa wartość dla productid 2, orderid 2
Sumuje tylko wiersze dla productid 3
Szczegółowa wartość dla productid 3, orderid 1
Szczegółowa wartość dla productid 3, orderid 2