

Operacje mnogościowe na wynikach zapytań

Filmy

id_filmu	tytul	rok_produkcji	cena
1	Wesele	2004	10
2	Ghostbusters	1984	5.50
3	Terminator	1984	8.50

Klienci

id_klienta	imie	nazwisko
1	Jan	Kowalski
2	Anna	Nowak

```
mydb=# SELECT TYTUL FROM FILMY
mydb=# UNION
mydb=# SELECT NAZWISKO FROM KLIENCI;
```

tytul

Ghostbusters

Kowalski

Nowak

Terminator

Wesele

(5 rows)

UNION – suma mnogościowa z eliminacją duplikatów

UNION ALL – suma mnogościowa bez eliminacji duplikatów

INTERSECT – część wspólna z eliminacją duplikatów

INTERSECT ALL – część wspólna bez eliminacji duplikatów

EXCEPT – różnica mnogościowa z eliminacją duplikatów

EXCEPT ALL – różnica mnogościowa bez eliminacji duplikatów

Operacje mnogościowe na wynikach zapytań

Filmy

id_filmu	tytul	rok_produkcji	cena
1	Wesele	2004	10
2	Ghostbusters	1984	5.50
3	Terminator	1984	8.50

```
mydb=# SELECT TYTUL FROM FILMY WHERE ROK_PRODUKCJI=1984
```

```
mydb=# INTERSECT
```

```
mydb=# SELECT TYTUL FROM FILMY WHERE CENA>7;
```

```
tytul
```

```
-----
```

```
Terminator
```

```
(1 row)
```

```
mydb=# SELECT TYTUL FROM FILMY WHERE ROK_PRODUKCJI=1984
```

```
mydb=# EXCEPT
```

```
mydb=# SELECT TYTUL FROM FILMY WHERE CENA>7;
```

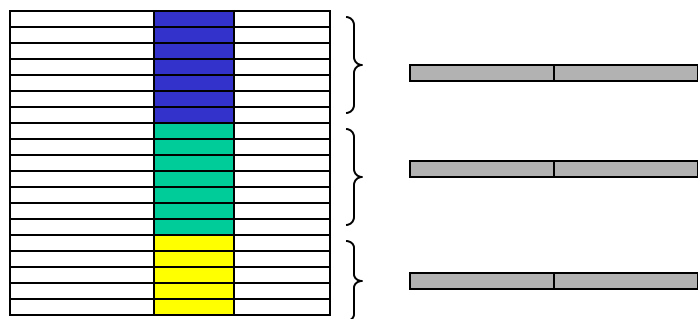
```
tytul
```

```
-----
```

```
Ghostbusters
```

```
(1 row)
```

Grupowanie i funkcje grupowe



Każda wartość atrybutu grupującego wyznacza jedną grupę krotek. Dla każdej grupy krotek wyznaczana jest pojedyncza wartość funkcji grupowej.

```
SELECT FUNKCJA(ATRYBUT)  
FROM RELACJA  
GROUP BY ATRYBUT ;
```

Grupowanie – przykład

Pracownicy

id_pracownika	imie	nazwisko	miasto	pensja
1	John	Smith	New York	150
2	Ben	Johnson	New York	250
3	Louis	Armstrong	New Orleans	75
4	John	Lennon	London	300
5	Peter	Gabriel	London	100

```
mydb=# SELECT MIASTO, AVG(PENSJA)
mydb=# FROM PRACOWNICY
mydb=# GROUP BY MIASTO;
```

```
      miasto      | avg
-----+-----
New York          | 200
New Orleans       | 75
London            | 200
(3 rows)
```

Funkcje grupowe – przykłady

Funkcja	Opis
avg()	średnia arytmetyczna
count()	liczba niepustych wartości
count(*)	liczba krotek
max()	największa z wartości
min()	najmniejsza z wartości
sum()	suma wszystkich wartości

Grupowanie bez GROUP BY

Pracownicy

id_pracownika	imie	nazwisko	miasto	pensja
1	John	Smith	New York	150
2	Ben	Johnson	New York	250
3	Louis	Armstrong	New Orleans	75
4	John	Lennon	London	300
5	Peter	Gabriel	London	100

```
mydb=# SELECT AVG(PENSJA)
```

```
mydb=# FROM PRACOWNICY;
```

```
avg
```

```
----
```

```
175
```

```
(1 rows)
```

Selekcja krotek przed grupowaniem

Pracownicy

id_pracownika	imie	nazwisko	miasto	pensja
1	John	Smith	New York	150
2	Ben	Johnson	New York	250
3	Louis	Armstrong	New Orleans	75
4	John	Lennon	London	300
5	Peter	Gabriel	London	100

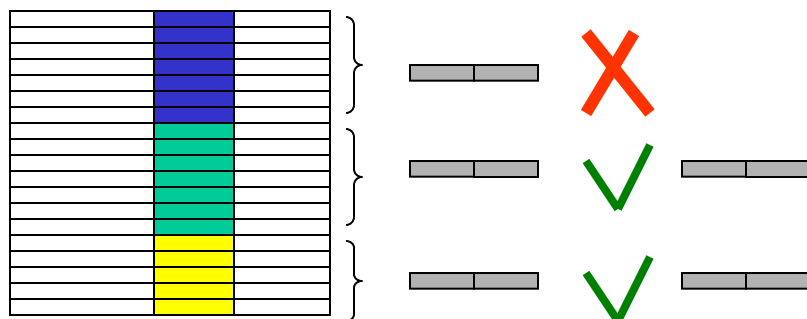


W klauzuli WHERE nie wolno
odwoływać się do funkcji grupowych

```
mydb=# SELECT SUM(PENSJA)
mydb=# FROM PRACOWNICY
mydb=# WHERE IMIE='John' ;

sum
-----
450
(1 row)
```

Selekcja krotek po grupowaniu



Każda wartość atrybutu grupującego wyznacza jedną grupę krotek. Dla każdej grupy krotek wyznaczana jest pojedyncza wartość funkcji grupowej. Z wyniku eliminowane są te grupy, które nie spełniają warunku HAVING

```
SELECT FUNKCJA(ATRYBUT)  
FROM RELACJA  
GROUP BY ATRYBUT  
HAVING WARUNEK ;
```


Selekcja krotek po grupowaniu - przykład

Pracownicy

id_pracownika	imie	nazwisko	miasto	pensja
1	John	Smith	New York	150
2	Ben	Johnson	New York	250
3	Louis	Armstrong	New Orleans	75
4	John	Lennon	London	300
5	Peter	Gabriel	London	100

```
mydb=# SELECT MIASTO, AVG(PENSJA)
mydb=# FROM PRACOWNICY
mydb=# GROUP BY MIASTO
mydb=# HAVING AVG(PENSJA)>100;
```

```
miasto      | avg
-----+-----
New York    | 200
London      | 200
(2 rows)
```

W klauzuli HAVING odwołujemy się do funkcji grupowych

Selekcja krotek przed grupowaniem i po grupowaniu

Pracownicy

id_pracownika	imie	nazwisko	miasto	pensja
1	John	Smith	New York	150
2	Ben	Johnson	New York	250
3	Louis	Armstrong	New Orleans	75
4	John	Lennon	London	300
5	Peter	Gabriel	London	100

Spośród pracowników, których imię jest różne od John, utwórz grupy według miast, wyznacz średnie pensje dla każdej z grup, a następnie przedstaw wyniki tylko dla tych grup, dla których średnia pensja przekroczyła 100.

```
mydb=# SELECT MIASTO, AVG(PENSJA)
mydb=# FROM PRACOWNICY
mydb=# WHERE IMIE<>'John'
mydb=# GROUP BY MIASTO
mydb=# HAVING AVG(PENSJA)>100;
```

```
miasto      | avg
-----+-----
New York    | 250
(1 row)
```