

# Połączenia relacji



# Połączenia relacji

- Iloczyn kartezjański – umożliwia połączenia każdej wartości jednej relacji z każdą wartością innej, dając w wyniku wszystkie możliwe kombinacje. Operatorem jest wyrażenie CROSS JOIN

Przykład: Wyświetl wszystkie kombinacje nazw etatów zaczynających się na literę A i nazwisk pracowników na literę N:

NAZWA	NAZWISKO
ADIUNKT	Nowicki
ADIUNKT	Nowak
ASYSTENT	Nowicki
ASYSTENT	Nowak

```
SELECT nazwa, nazwisko  
FROM etaty CROSS JOIN pracownicy  
WHERE nazwa LIKE 'A%' AND nazwisko LIKE 'N%';
```

# Połączenia relacji

- Połączenia równościowe – łączą ze sobą relacje względem atrybutu tej samej wartości. Operatorem jest wyrażenie JOIN.

Przykład: Wyświetl nazwiska pracowników wraz z identyfikatorem zespołu i jego nazwą.

NAZWISKO	ID_ZESP	ID_ZESP	NAZWA
Marecki	10	10	ADMINISTRACJA
Janicki	40	20	SYSTEMY ROZPROSZONE
Nowicki	30	30	SYSTEMY EKSPERCKIE
Nowak	20	40	ALGORYTMY
.....	.....	50	BADANIA OPERACYJNE

# Połączenia relacji

Notacja pierwsza:

```
SELECT pracownicy.nazwisko, zespoly.nazwa,  
        zespoly.id_zesp  
FROM pracownicy JOIN zespoly ON  
        pracownicy.id_zesp = zespoly.id_zesp;
```

Notacja druga:

```
SELECT p.nazwisko, z.nazwa, z.id_zesp  
FROM pracownicy p JOIN zespoly z ON  
        p.id_zesp = z.id_zesp;
```

# Połączenia relacji

Zadanie: Dla każdego pracownika zatrudnionego na etacie DYREKTOR, albo SEKRETARKA wyświetl jego płacę podstawową i widełki płacowe.

NAZWISKO	PLACA_POD	PLACA_OD	PLACA_DO
Marecki	4730	4280	5100
Krakowska	1590	1470	1650

```
SELECT p.nazwisko, p.placa_pod, e.placa_od,  
       e.placa_do  
FROM etaty e JOIN pracownicy p ON p.etat = e.nazwa  
WHERE p.etat IN ('DYREKTOR', 'SEKRETARKA');
```

# Połączenia relacji

- Połączenia naturalne – są szczególnym rodzajem połączeń równościowych, gdzie połączenie pomiędzy relacjami następuje na podstawie atrybutu o tej samej nazwie (atrybuty muszą mieć te same wartości), np.:

```
SELECT p.nazwisko, z.nazwa, id_zesp  
FROM pracownicy p NATURAL JOIN zespoły z;
```

Drugi sposób zapisu:

```
SELECT p.nazwisko, z.nazwa, id_zesp  
FROM pracownicy p JOIN zespoły z USING (id_zesp);
```

# Połączenia relacji

Zadanie: Dla każdego zespołu wyświetl liczbę zatrudnionych w nim pracowników. Użyj połączenia naturalnego.

<u>NAZWA</u>	<u>COUNT(*)</u>
ALGORYTMY	1
SYSTEMY EKSPERCKIE	3
SYSTEMY ROZPROSZONE	7
ADMINISTRACJA	2

```
SELECT z.nazwa, COUNT(p.id_prac)  
FROM pracownicy p NATURAL JOIN zespoly z  
GROUP BY z.nazwa;
```



# Połączenia relacji

- Połączenia nierównościowe – są podobne do połączeń równościowych, w tych jednak zmienia się warunek połączeń relacji występujący po wyrażeniu ON

Przykład: Wyświetl nazwiska i etaty pracowników, których rzeczywiste zarobki odpowiadają widełkom płacowym przewidzianym dla sekretarek

NAZWISKO	ETAT
Krakowska	SEKRETARKA

```
SELECT p.nazwisko, p.etat  
FROM pracownicy p JOIN etaty e ON  
p.placa_pod BETWEEN e.placa_od AND e.placa_do  
WHERE e.nazwa = 'SEKRETARKA';
```

# Połączenia relacji

- Połączenia zewnętrzne – połączenia relacji, które w wyniku zwracają również te krotki które nie znalazły wspólnej pary z drugą relacją. Łączone są one z dodatkową wirtualną pustą krotką i wyświetlane są w wyniku.

Przykład: Wyświetl nazwę zespołu, nazwisko pracownika oraz jego etat uwzględniając wszystkie wystąpienia w relacji pracownicy.

```
SELECT z.nazwa, p.nazwisko, p.etat  
FROM zespoly z RIGHT OUTER JOIN pracownicy p  
ON z.id_zesp = p.id_zesp;
```

Zamiast wyrażenia RIGHT może wystąpić LEFT (wskazuje na relację po lewej stronie) lub FULL (co jest połączeniem zewnętrznym pełnym)

# Połączenia relacji

Zadanie: Dla każdego zespołu wyświetl liczbę zatrudnionych w nim pracowników. W wyniku ma zostać uwzględniony zespół badania operacyjne, na którym nie zatrudniono żadnego pracownika.

NAZWA	COUNT
SYSTEMY EKSPERCKIE	3
ALGORYTMY	1
SYSTEMY ROZPROSZONE	7
ADMINISTRACJA	2
BADANIA OPERACYJNE	0

```
SELECT z.nazwa, COUNT(p.id_prac)  
FROM pracownicy p NATURAL RIGHT JOIN zespoły z  
GROUP BY z.nazwa;
```

# Połączenia relacji

- Połączenia zwrotne – specjalny typ połączeń polegający na łączeniu tabeli z samą sobą. Połączenie może być wewnętrzne, zewnętrzne, równościowe lub nierównościowe.

Klasycznym przykładem takiego połączenia jest wyszukanie szefa pracownika. W tabeli pracownicy jest informacja o wszystkich osobach w zakładzie, a każdy pracownik posiada przypisany „numer id” swojego szefa:

```
SELECT p.nazwisko, s.nazwisko  
FROM pracownicy p JOIN pracownicy s  
ON p.id_szefa = s.id_prac;
```

# Połączenia relacji

Zadanie: Wyświetl nazwiska wszystkich pracowników, którzy zarabiają więcej od Nowickiego

NAZWISKO

Marecki

Janicki

Nowak

Kowalski

```
SELECT p.nazwisko  
FROM pracownicy p JOIN pracownicy r  
ON p.placa_pod > r.placa_pod  
WHERE r.nazwisko = 'Nowicki';
```

# Połączenia relacji

- Łączenie wielu tabel - szczególny rodzaj operacji, gdzie wynik (nowa relacja) połączenia dwóch relacji jest w odpowiedni sposób łączony z inną relacją.

```
SELECT p.nazwisko, e.nazwa, e.placa_od, e.placa_do,  
       z.nazwa  
FROM zespoly z NATURAL RIGHT OUTER JOIN  
     pracownicy p JOIN etaty e  
     ON p.etat = e.nazwa;
```

```
SELECT p.nazwisko, e.placa_od, e.placa_do, z.nazwa  
FROM zespoly z FULL OUTER JOIN  
     (pracownicy p JOIN etaty e ON p.etat = e.nazwa)  
     ON p.id_zesp = z.id_zesp;
```