

Rozdział 4

Połączenia i operatory zbiorowe

Iloczyn kartezjański, połączenie równościowe, połączenie nierównościowe, połączenie zwrotne, połączenie zewnętrzne, operatory zbiorowe, składnia ANSI połączeń



Iloczyn kartezjański

- Iloczyn kartezjański dwóch relacji to zbiór wszystkich możliwych kombinacji krotek z obu relacji. W większości wypadków wystąpienie iloczynu kartezjańskiego sygnalizuje błąd w zapytaniu. Iloczyn kartezjański rzadko bywa przydatny.
- Iloczyn kartezjański nazywa się także *cross-join*.

```
SELECT nazwisko, etat, nazwa
FROM pracownicy, etaty;
```

```
SELECT nazwisko, etat, nazwa
FROM pracownicy CROSS JOIN etaty;
```



Połączenie równościowe (1)

PRACOWNICY				ZESPOLY	
ID_PRAC	NAZWISKO	ID_ZESP		ID_ZESP	NAZWA
180	MAREK	10		10	ADMINISTRACJA
200	ZAKRZEWICZ	30		30	SYSTEMY EKSPERCKIE
210	BIAŁY	30			

PRACOWNICY				ETATY	
ID_PRAC	NAZWISKO	ETAT		NAZWA	PLACA_MIN
180	BLAZEWICZ	PROFESOR		PROFESOR	800
200	SLOWINSKI	PROFESOR		ADIUNKT	510
210	KOSZLAJDA	ADIUNKT			

• Połączenie relacji w wyniku którego uzyskujemy tylko krotki spełniające warunki połączenia nazywa się połączeniem wewnętrznym (*inner join*).

• Połączenie równościowe nazywa się *equi join*.

• Połączenie równościowe, w którym warunek połączenia dotyczy atrybutów o tej samej nazwie nazywa się *natural join*.



Połączenie równościowe (2)

- Warunek porównujący ze sobą wartości atrybutów z dwóch różnych relacji nazywamy warunkiem połączenia.
- Jeżeli w łączonych relacjach występują atrybuty o takich samych nazwach, to w zapytaniu muszą być poprzedzone nazwą relacji w celu uniknięcia dwuznaczności.
- W klauzuli FROM można wprowadzić aliasy (alternatywne nazwy relacji) i używać aliasów zamiast nazw relacji. Jeśli alias został użyty, to nie wolno używać oryginalnej nazwy relacji.
- Łącząc N relacji należy podać minimum N-1 warunków połączenia.

```
SELECT p.nazwisko, z.id_zesp, z.nazwa
FROM pracownicy p, zespoly z
WHERE p.id_zesp = z.id_zesp
```

aliasy



Połączenie naturalne ANSI

Połączenie naturalne dwóch relacji to połączenie równościowe tych relacji, w którym warunki równości dotyczą wszystkich par atrybutów o takich samych nazwach należących do łączonych relacji.

```
SELECT nazwisko, id_zesp, nazwa
FROM pracownicy NATURAL JOIN zespoly;
```

```
SELECT nazwisko, id_zesp, nazwa
FROM pracownicy JOIN zespoly USING (id_zesp);
```

Połączenie równościowe ANSI

W wyniku połączenia równościowego otrzymujemy wszystkie krotki z obu tabel spełniające warunki połączenia. Połączenie jest lewostronnie łączne.

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

```
SELECT nazwisko, etat, nazwa
FROM pracownicy INNER JOIN etaty
ON etat = nazwa;
```

```
SELECT p.nazwisko, z.nazwa, e.nazwa, e.placa_min, e.placa_max
FROM pracownicy p JOIN zespoly z ON p.id_zesp = z.id_zesp
JOIN etaty e ON p.etat = e.nazwa;
```

Różnica między połączeniem naturalnym i równościowym

W wyniku połączenia naturalnego (zarówno NATURAL JOIN jak i JOIN USING ()) atrybut połączeniowy występuje tylko raz. W wyniku połączenia równościowego występują oba atrybuty połączeniowe z obu łączonych relacji.

```
SELECT * FROM pracownicy NATURAL JOIN zespoly;
```

```
SELECT * FROM pracownicy JOIN zespoly USING (id_zesp);
```

ID_ZESP	ID_PRAC	NAZWISKO	ETAT	ID_SZEFA	ZATRUDNIO	PLACA_POD	PLACA_DOD	NAZWA	ADRES
10	100	WEGLARZ	DYREKTOR		01-JAN-68	1730	420.5	ADMINISTRACJA	PIOTROWO 3A
40	110	BLAZEWICZ	PROFESOR	100	01-MAY-73	1350	210	ALGORYTMY	WLODKOWICA 16
30	120	SLOWINSKI	PROFESOR	100	01-SEP-77	1070	305	SYSTEMY EKSPERCKIE	STRZELECKA 14

```
SELECT * FROM pracownicy JOIN zespoly
ON pracownicy.id_zesp = zespoly.id_zesp;
```

ID_PRAC	NAZWISKO	ETAT	ID_SZEFA	ZATRUDNIO	PLACA_POD	PLACA_DOD	ID_ZESP	ID_ZESP	NAZWA	ADRES
100	WEGLARZ	DYREKTOR		01-JAN-68	1730	420.5	10	100	ADMINISTRACJA	PIOTROWO 3A
110	BLAZEWICZ	PROFESOR	100	01-MAY-73	1350	210	40	40	ALGORYTMY	WLODKOWICA 16
120	SLOWINSKI	PROFESOR	100	01-SEP-77	1070	305	30	305	SYSTEMY EKSPERCKIE	STRZELECKA 14

Połączenie nierównościowe

PRACOWNICY

ID_PRAC	NAZWISKO	PLACA_POD
180	MAREK	410.2
200	ZAKRZEWICZ	208

ETATY

NAZWA	PLACA_MIN	PLACA_MAX
SEKRETARKA	270	450
STAŻYSTA	150	250

- Jako operator w warunku połączenia można zastosować dowolny operator języka SQL. Użycie operatora innego niż = jest rzadko spotykane.

```
SELECT nazwisko, nazwa, placa_pod, placa_min, placa_max
FROM pracownicy, etaty
WHERE placa_pod BETWEEN placa_min AND placa_max;
```

```
SELECT nazwisko, nazwa, placa_pod, placa_min, placa_max
FROM pracownicy JOIN etaty
ON placa_pod BETWEEN placa_min AND placa_max;
```

Połączenie zewnętrzne (1)

PRACOWNICY

ID_PRAC	NAZWISKO	ID_ZESP
180	MAREK	10
?	?	?

null	null	null
------	------	------

ZESPOLY

ID_ZESP	NAZWA
40	ALGORYTMY
50	BADANIA OPERACYJNE

ZESPOLY

ID_ZESP	NAZWA
50	BADANIA OPERACYJNE
?	?

null	null
------	------

PRACOWNICY

ID_PRAC	NAZWISKO	ID_ZESP
180	MAREK	10
300	NOWICKI	99

Połączenie zewnętrzne (2)

- Wiersze z obu relacji nie posiadające odpowiedników spełniających warunek połączenia nie są wyświetlane. W efekcie zespół 50 „Badania operacyjne” nie zostaje wyświetlony.
- Połączenie zewnętrzne zachowuje wszystkie krotki z wybranej relacji, łącząc je z "wirtualnymi" krotkami z drugiej krotki. "Wirtualne" krotki są wypełnione wartościami pustymi.
- Połączenie zewnętrzne nazywa się też *outer join*.

```
SELECT z.id_zesp, z.nazwa, p.nazwisko
FROM ZESPOLY z, pracownicy p
WHERE p.id_zesp (+) = z.id_zesp;
```

```
SELECT z.id_zesp, z.nazwa, p.nazwisko
FROM zespoly z, PRACOWNICY p
WHERE p.id_zesp = z.id_zesp (+);
```

Połączenie zewnętrzne ANSI

Składnia połączeń ANSI pozwala jawnie wskazać relację, której wszystkie krotki mają być zachowane. Poza tym składnia ANSI pozwala na wykonanie obustronnego połączenia zewnętrznego.

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

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

```
SELECT nazwa, nazwisko, etat
FROM ZESPOLY FULL OUTER JOIN PRACOWNICY
USING (id_zesp);
```

Połączenie zwrotne (1)



- Użycie aliasów w połączeniu zwrotnym jest obowiązkowe.
- Połączenie zwrotne nazywa się też *self join*.

Połączenie zwrotne (2)

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

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

Operatory zbiorowe

UNION

- suma zbiorów
- eliminuje duplikaty

UNION ALL

- suma zbiorów
- nie eliminuje duplikatów

INTERSECT

- część wspólna zbiorów
- eliminuje duplikaty

MINUS (standard SQL: EXCEPT)

- różnica zbiorów
- eliminuje duplikaty

- Operatory zbiorowe nazywamy również operatorami złączeń pionowych (*vertical join*) ponieważ łączą kolumny zbiorów wynikowych.

```
SELECT etat  
FROM pracownicy  
WHERE id_zesp = 20  
MINUS  
SELECT etat  
FROM pracownicy  
WHERE id_zesp = 30  
ORDER BY etat;
```

```
SELECT nazwisko  
FROM pracownicy  
UNION  
SELECT nazwa  
FROM zespoly;
```

Reguły stosowania operatorów zbiorowych

- W łączonych operatorami zbiorowymi klauzulach SELECT musi wystąpić ta sama liczba atrybutów.
- Typy odpowiednich atrybutów różnych klauzul SELECT muszą być zgodne.
- W wyniku zapytania pojawiają się nazwy atrybutów wyłącznie z pierwszej klauzuli SELECT.
- Klauzula ORDER BY może być użyta tylko jako ostatnia klauzula zapytania.
- Polecenia SELECT są wykonywane w kolejności ich wystąpienia (od góry do dołu), nawiasy umożliwiają zmianę domyślnej kolejności wykonywania łączonych operatorami zbiorowymi poleceń SELECT.
- UWAGA: w celu zapewnienia zgodności ze standardem SQL w przyszłych wydaniach Oracle operator INTERSECT będzie miał największy priorytet.