

# Bazy danych

Andrzej Grzybowski

Instytut Fizyki, Uniwersytet Śląski

# Wykład 1

## Algebra relacyjnych baz danych

jako podstawa języka SQL i jego  
implementacji w systemach baz danych  
Oracle

# Algebra relacyjnych baz danych

**W teorii mnogości (teorii zbiorów)**  
**relacją  $R$  określoną:**

- **w zbiorze  $A$**  nazywamy dowolny podzbiór iloczynu kartezyjskiego  $A \times A$ :

$$R \subset A \times A$$

- **w zbiorach  $A, B$**  nazywamy dowolny podzbiór iloczynu kartezyjskiego  $A \times B$ :

$$R \subset A \times B$$

- **w zbiorach  $A_1, A_2, \dots, A_n$**  nazywamy dowolny podzbiór iloczynu  $A_1 \times A_2 \times \dots \times A_n$ :

$$R \subset A_1 \times A_2 \times \dots \times A_n$$

# Algebra relacyjnych baz danych

- **Relację R relacyjnej bazy danych (RBD)** można z punktu widzenia teorii zbiorów wstępnie traktować jako nazwany dowolny podzbiór iloczynu kartezyjskiego zbiorów  $D_1, D_2, \dots, D_n$ , będących dziedzinami (zbiorami wartości) odpowiednio atrybutów  $A_1, A_2, \dots, A_n$ .
- Jednak należy pamiętać, że wykorzystanie w tej definicji iloczynu kartezyjskiego  $D_1 \times D_2 \times \dots \times D_n$  **nie jest precyzyjne**. Jest tak głównie dlatego, że kolejność atrybutów  $A_1, A_2, \dots, A_n$  relacji RBD nie ma znaczenia.

# Algebra relacyjnych baz danych

## Tabelaryczna reprezentacja relacji RBD

<b>R</b> ← nazwa relacji		
<b>A</b>	<b>B</b>	<b>C</b>
a1	b2	c3
a2	b1	c3
a3	b1	c2
a2	b2	c2

← nagłówek relacji złożony z nazw atrybutów

← krotka

← krotka

← krotka

← krotka

Dziedziny atrybutów:

$$D_A = D(A) = \{a1, a2, a3\}$$

$$D_B = D(B) = \{b1, b2\}$$

$$D_C = D(C) = \{c2, c3\}$$

# Algebra relacyjnych baz danych

## Wymagane cechy, aby relacja $R$ była relacją RBD:

- 1) atomowość: elementy krotek relacji  $R$  reprezentują pojedyncze wartości,
- 2) każdy atrybut  $A$  ma rozróżnialną nazwę,
- 3) wszystkie wartości atrybutu  $A$  pochodzą z tej samej dziedziny (domeny)  $D$ ,
- 4) porządek atrybutów w relacji jest nieistotny,
- 5) każda krotka jest różna,
- 6) porządek krotek w relacji jest nieistotny.

# Kompatybilność relacji

- Schemat relacji inaczej nagłówek relacji R to zbiór nazw atrybutów  $A_1, A_2, \dots, A_n$  relacji R:

$$N(R) = \{A_1, A_2, \dots, A_n\}$$

- Relacje R i S nazywamy kompatybilnymi, gdy mają takie same nagłówki:  $N(R) = N(S)$  tzn.  $D(R.A_1) = D(S.A_1), D(R.A_2) = D(S.A_2), \dots, D(R.A_n) = D(S.A_n)$

# Działania algebry RBD

## **Oparte na teorii zbiorów:**

- 1) Unia
- 2) Przecięcie (iloczyn)
- 3) Różnica
- 4) Produkt (iloczyn kartezjański)

## **Dotyczące informacji o strukturze krotek:**

- 5) Przypisanie (alias)
- 6) Projekcja (rzutowanie)
- 7) Selekcja
- 8) Złączenie
- 9) Dzielenie



# Unia

Niech  $R$  i  $S$  są relacjami kompatybilnymi.  
Unią relacji  $R$  i  $S$  jest relacja  $R \cup S$  o takim samym nagłówku, zawierająca wszystkie krotki znajdujące się w  $R$ ,  $S$  lub w obydwu relacjach.

R:

X	Y	Z
x1	y1	z1
x1	y2	z3
x2	y1	z2

S:

X	Y	Z
x1	y1	z1
x1	y1	z2
x1	y2	z3
x3	y2	z3

$R \cup S$ :

X	Y	Z
x1	y1	z1
x1	y2	z3
x2	y1	z2
x1	y1	z2
x3	y2	z3

Standard SQL i Oracle SQL:

```
select X,Y,Z from R
```

**union**

```
select X,Y,Z from S;
```

# Przecięcie (iloczyn)

Niech  $R$  i  $S$  są relacjami kompatybilnymi.

Przecięciem relacji  $R$  i  $S$  jest relacja  $R \cap S$

o takim samym nagłówku, zawierająca te krotki, które znajdują się zarówno w  $R$ , jak i w  $S$ .

R:

X	Y	Z
x1	y1	z1
x1	y2	z3
x2	y1	z2

S:

X	Y	Z
x1	y1	z1
x1	y1	z2
x1	y2	z3
x3	y2	z3

$R \cap S$ :

X	Y	Z
x1	y1	z1
x1	y2	z3

Standard SQL i Oracle SQL:

```
select X,Y,Z from R
```

**intersect**

```
select X,Y,Z from S;
```

# Różnica

Niech  $R$  i  $S$  są relacjami kompatybilnymi.  
Różnicą relacji  $R$  i  $S$  jest relacja  $R - S$  o takim samym nagłówku, zawierająca wszystkie krotki z relacji  $R$ , których nie ma w relacji  $S$ .

R:

X	Y	Z
x1	y1	z1
x1	y2	z3
x2	y1	z2

S:

X	Y	Z
x1	y1	z1
x1	y1	z2
x1	y2	z3
x3	y2	z3

$R-S$ :

X	Y	Z
x2	y1	z2

Standard SQL i Oracle SQL:

```
select X,Y,Z from R
```

**minus**

```
select X,Y,Z from S;
```

# Produkt (iloczyn kartezjański)

Niech relacje  $R$  i  $S$  mają nagłówki odpowiednio  $N(R)=\{X1, X2, \dots, Xn\}$  i  $N(S)=\{Y1, Y2, \dots, Ym\}$ .

Iloczynem kartezjańskim relacji  $R$  i  $S$  jest relacja  $R \times S$  o nagłówku

$$N(R \times S) = \{R.X1, R.X2, \dots, R.Xn, S.Y1, S.Y2, \dots, S.Ym\},$$

zawierająca krotki będące wszystkimi możliwymi połączeniami krotek z relacji  $R$  i  $S$ .

Jeśli nazwy atrybutów relacji  $R$  i  $S$  są różne, to wówczas nagłówek iloczynu kartezjańskiego można zapisać w postaci:

$$N(R \times S) = \{X1, X2, \dots, Xn, Y1, Y2, \dots, Ym\}$$

# Produkt (iloczyn kartezjański)

R:

X	Y	Z
x1	y1	z1
x1	y2	z3
x2	y1	z2

S:

X	Y	Z
x1	y1	z1
x1	y1	z2
x1	y2	z3
x3	y2	z3

$R \times S$ :

R.X	R.Y	R.Z	S.X	S.Y	S.Z
x1	y1	z1	x1	y1	z1
x1	y1	z1	x1	y1	z2
x1	y1	z1	x1	y2	z3
x1	y1	z1	x3	y2	z3
x1	y2	z3	x1	y1	z1
x1	y2	z3	x1	y1	z2
x1	y2	z3	x1	y2	z3
x1	y2	z3	x3	y2	z3
x2	y1	z2	x1	y1	z1
x2	y1	z2	x1	y1	z2
x2	y1	z2	x1	y2	z3
x2	y1	z2	x3	y2	z3

Standard SQL i Oracle SQL:

```
select R.X,R.Y,R.Z,S.X,S.Y,S.Z from R, S;
```

# Alias (przypisanie) relacji

Aliasem relacji  $R$  nazywamy nadanie tej relacji innej nazwy, np.  $R=A$

A:

X	Y	Z
x1	y1	z1
x1	y2	z3
x2	y1	z2

Oracle SQL:

```
select A.X,A.Y,A.Z from R A;
```

Przykład (koniecznego) zastosowania aliasu do obliczenia kwadratu kartezyjskiego  $R \times R$ :

$R=A$ ,  $R=B$ , wówczas:

$R \times R = A \times B$ :

A.X	A.Y	A.Z	B.X	B.Y	B.Z
x1	y1	z1	x1	y1	z1
x1	y1	z1	x1	y2	z3
x1	y1	z1	x2	y1	z2
x1	y2	z3	x1	y1	z1
x1	y2	z3	x1	y2	z3
x1	y2	z3	x2	y1	z2
x2	y1	z2	x1	y1	z1
x2	y1	z2	x1	y2	z3
x2	y1	z2	x2	y1	z2

# Alias (przypisanie) atrybutu

Aliasem atrybutu X relacji R nazywamy nadanie temu atrybutowi innej nazwy, np. **X=Xnowy**

R:

Xnowy	Y	Z
x1	y1	z1
x1	y2	z3
x2	y1	z2

Oracle SQL:

```
select X as Xnowy, Y, Z from R;
```

lub

```
select X Xnowy, Y, Z from R;
```

# Alias (przypisanie) atrybutu

Aliasem atrybutu X relacji R nazywamy nadanie temu atrybutowi innej nazwy, np. **X=Xnowy**

R:

Xnowy	Y	Z
x1	y1	z1
x1	y2	z3
x2	y1	z2

Oracle SQL:

```
select X as Xnowy, Y, Z from R;
```

lub

```
select X Xnowy, Y, Z from R;
```



# Alias (przypisanie) atrybutu

Aliasem atrybutu X relacji R nazywamy nadanie temu atrybutowi innej nazwy, np. **X=Xnowy**

R:

Xnowy	Y	Z
x1	y1	z1
x1	y2	z3
x2	y1	z2

Oracle SQL:

```
select X as Xnowy, Y, Z from R;
```

lub

```
select X Xnowy, Y, Z from R;
```

# Projekcja (rzutowanie)

Niech relacja  $R$  ma nagłówek  $N(R)=\{X_1, X_2, \dots, X_n\}$ .  
Projekcja relacji  $R$  na atrybuty  $X_{i1}, X_{i2}, \dots, X_{ik}$ ,  
gdzie  $\{X_{i1}, X_{i2}, \dots, X_{ik}\} \subset \{X_1, X_2, \dots, X_n\}$ , jest relacją  
 $T=R[X_{i1}, X_{i2}, \dots, X_{ik}]$  o nagłówku  $N(T)=\{X_{i1}, X_{i2}, \dots, X_{ik}\}$ ,  
zawierającą krotki  $t$  spełniające warunek:

$\bigwedge$  krotki  $r \in R \quad \bigvee$  pojedyncza krotka  $t \in T$ :

$t(X_{ij})=r(X_{ij}) \quad \bigwedge j: 1 \leq j \leq k.$

Inna często używana notacja projekcji relacji  $R$   
na atrybuty  $X_{i1}, X_{i2}, \dots, X_{ik}$ :  $\pi_{X_{i1}, X_{i2}, \dots, X_{ik}}(R)$

Wersja tekstowa:  $\pi_{\{X_{i1}, X_{i2}, \dots, X_{ik}\}}(R)$

# Projekcja (rzutowanie)

S:

X	Y	Z
x1	y1	z1
x1	y1	z2
x1	y2	z3
x3	y2	z3

S[X,Y]:

X	Y
x1	y1
x1	y2
x3	y2

Standard SQL i Oracle SQL:

```
select X,Y from S;
```

# Selekcja

Niech relacja  $R$  ma nagłówek  $N(R)=\{X_1, X_2, \dots, X_n\}$ . Selekcja relacji  $R$  zgodnie z kryterium  $C$  jest relacją  $T = R$  gdzie  $C$  o tym samym nagłówku co relacja  $R$ , zawierającą krotki  $s$  spełniające warunek selekcji  $C$ , który może przyjmować następujące postaci:

1)  $C$  może być dowolnym porównaniem:

$X_i \propto X_j$  lub  $X_i \propto x$ , gdzie  $x \in D(X_i)$  - dziedziny atrybutu  $X_i$ , a operator porównania  $\propto$  może być następujący:

$<$ ,  $>$ ,  $=$ ,  $<=$ ,  $>=$ ,  $<>$ ; w Oracle'u można stosować także operator różności:  $!=$

2) Jeżeli  $C'$  i  $C''$  są warunkami selekcji, wówczas można utworzyć nowy warunek  $C$  w postaci:

$C'$  i  $C''$  ( $C'$  and  $C''$ ),  $C'$  lub  $C''$  ( $C'$  or  $C''$ ), nie  $C$  ( $\text{not } C$ ).

Inna często używana notacja selekcji relacji  $R$  zgodnie z kryterium  $C$ :  $\sigma_C(R)$

Wersja tekstowa:  $\sigma_{\{C\}}(R)$

# Selekcja

R:

X	Y	Z
x1	y1	z1
x1	y2	z3
x2	y1	z2

R gdzie (X=x1 i Y=y2)

X	Y	Z
x1	y2	z3

Standard SQL i Oracle SQL:

```
select X,Y,Z from R
where X=x1 and Y=y2;
```

# Złączenia

- **Złączenie naturalne** (odnosi się zwykle do identycznie nazwanych atrybutów w relacjach wchodzących w to złączenie)
  - Złączenie wewnętrzne (inaczej: proste złączenie), tu wynikiem są krotki spełniające dokładnie warunek złączenia
  - **Złączenia zewnętrzne** (nadzbiór złączenia wewnętrznego):
    - złączenie zewnętrzne lewostronne,
    - złączenie zewnętrzne prawostronne,
    - złączenie zewnętrzne obustronne
  - Równozłączenie (do porównania atrybutów współdzielonych używamy operatora: =)
- $\theta$ -złączenia, w tym typie złączenia do porównania atrybutów współdzielonych używamy operatora  $\theta$ :
- jeśli operatorem  $\theta$  jest:  $<$ ,  $>$ ,  $<=$ ,  $>=$ , to mówimy o semizłączeniu (półzłączeniu)
  - jeśli  $\neq$  operatorem różności  $<>$  ( $\neq$ ), to mówimy o antyzłączeniu
- Samozłączenie (złączenie relacji samej z sobą)

# Złączenie naturalne wewnętrzne

Niech relacje  $R$  i  $S$  mają nagłówki odpowiednio

$N(R) = \{X_1, X_2, \dots, X_n, Z_1, Z_2, \dots, Z_k\}$  i  $N(S) = \{Z_1, Z_2, \dots, Z_k, Y_1, Y_2, \dots, Y_m\}$ ,

gdzie  $n, k, m \geq 0$  oraz  $\{Z_1, Z_2, \dots, Z_k\}$  jest **zupełnym podzbiorem atrybutów współdzielonych** przez obie relacje.

Złączenie (naturalne)  $R$  i  $S$  jest relacją  $R \bowtie S$  o nagłówku

$N(R \bowtie S) = \{X_1, X_2, \dots, X_n, Z_1, Z_2, \dots, Z_k, Y_1, Y_2, \dots, Y_m\}$ , zawierającą jedynie takie krotki  $t$ , dla których **spełniony jest warunek równozłączenia**, tzn.

$\forall$  krotki  $r \in R$  oraz  $s \in S$ :  $r(Z_i) = s(Z_i) \wedge i: 1 \leq i \leq k$

wówczas  $\forall t \in R \bowtie S$ :  $t(X_1, X_2, \dots, X_n, Z_1, Z_2, \dots, Z_k, Y_1, Y_2, \dots, Y_m)$

Zdefiniowane powyżej ZŁĄCZENIE NATURALNE należy do ZŁĄCZEŃ WEWNĘTRZNYCH oraz RÓWNOZŁĄCZEŃ, ale należy pamiętać, że **ZŁĄCZENIE NATURALNE może dotyczyć szerszej kategorii złączeń.**

ZŁĄCZENIE NATURALNE bywa utożsamiane po prostu ze ZŁĄCZENIEM, jednak częściej, tak jak np. w Oracle SQL,

**ZŁĄCZENIE utożsamia się ze ZŁĄCZENIEM WEWNĘTRZNYM!**

# Złączenie naturalne wewnętrzne

R:

X	Z1	Z2
x1	z1	z1'
x1	z2	z1'
x2	z1	z2'

S:

Z1	Z2	Y
z1	z1'	y1
z1	z1'	y2
z1	z2'	y3
z2	z2'	y4

$R \bowtie S$ :

X	Z1	Z2	Y
x1	z1	z1'	y1
x1	z1	z1'	y2
x2	z1	z2'	y3



# Złączenie naturalne wewnętrzne

Jedyna możliwość w starszych wersjach Oracle SQL:

```
select X, R.Z1, R.Z2, Y from R, S  
where R.Z1=S.Z1 and R.Z2=S.Z2;
```

Nowe możliwości Oracle SQL inspirowane przez Standard SQL:

```
select X, Z1, Z2, Y from R natural join S;  
select X, R.Z1, R.Z2, Y from R inner join S  
on R.Z1=S.Z1 and R.Z2=S.Z2;  
select X, R.Z1, R.Z2, Y from R join S  
on R.Z1=S.Z1 and R.Z2=S.Z2;
```

# Złączenie (naturalne) zewnętrzne lewostronne

R:

X	Z1	Z2
x1	z1	z1'
x1	z2	z1'
x2	z1	z2'

S:

Z1	Z2	Y
z1	z1'	y1
z1	z1'	y2
z1	z2'	y3
z2	z2'	y4

$R \bowtie_{LO} S$ :

X	Z1	Z2	Y
x1	z1	z1'	y1
x1	z1	z1'	y2
x2	z1	z2'	y3
x1	z2	z1'	NULL

## Złączenie (naturalne) zewnętrzne lewostronne

Jedyna możliwość w starszych wersjach Oracle SQL:

```
select X, R.Z1, R.Z2, Y from R, S  
where R.Z1=S.Z1(+) and R.Z2=S.Z2(+);
```

Nowe możliwości Oracle SQL inspirowane przez Standard SQL:

```
select X, R.Z1, R.Z2, Y from R left join S  
on R.Z1=S.Z1 and R.Z2=S.Z2;
```

```
select X, R.Z1, R.Z2, Y from R left outer join S  
on R.Z1=S.Z1 and R.Z2=S.Z2;
```

A nawet:

```
select X, Z1, Z2, Y from R natural left join S;
```

```
select X, Z1, Z2, Y from R natural left outer join S;
```

# Złączenie (naturalne) zewnętrzne prawostronne

R:

X	Z1	Z2
x1	z1	z1'
x1	z2	z1'
x2	z1	z2'

S:

Z1	Z2	Y
z1	z1'	y1
z1	z1'	y2
z1	z2'	y3
z2	z2'	y4

$R \bowtie_{RO} S$ :

X	Z1	Z2	Y
x1	z1	z1'	y1
x1	z1	z1'	y2
x2	z1	z2'	y3
NULL	z2	z2'	y4

# Złączenie (naturalne) zewnętrzne prawostronne

Jedyna możliwość w starszych wersjach Oracle SQL:

```
select X, R.Z1, R.Z2, Y from R, S  
where R.Z1(+)=S.Z1 and R.Z2(+)=S.Z2;
```

Nowe możliwości Oracle SQL inspirowane przez Standard SQL:

```
select X, R.Z1, R.Z2, Y from R right join S  
on R.Z1=S.Z1 and R.Z2=S.Z2;
```

```
select X, R.Z1, R.Z2, Y from R right outer join S  
on R.Z1=S.Z1 and R.Z2=S.Z2;
```

A nawet:

```
select X, Z1, Z2, Y from R natural right join S;
```

```
select X, Z1, Z2, Y from R natural right outer join S;
```

# Złączenie (naturalne) zewnętrzne obustronne

R:

X	Z1	Z2
x1	z1	z1'
x1	z2	z1'
x2	z1	z2'

S:

Z1	Z2	Y
z1	z1'	y1
z1	z1'	y2
z1	z2'	y3
z2	z2'	y4

$R \bowtie S$ :

X	Z1	Z2	Y
x1	z1	z1'	y1
x1	z1	z1'	y2
x2	z1	z2'	y3
x1	z2	z1'	NULL
NULL	z2	z2'	y4

# Złączenie (naturalne) zewnętrzne obustronne

Jedyna możliwość w starszych wersjach Oracle SQL:

```
select X, R.Z1, R.Z2, Y from R, S  
where R.Z1=S.Z1(+) and R.Z2=S.Z2(+)
```

union

```
select X, R.Z1, R.Z2, Y from R, S  
where R.Z1(+)=S.Z1 and R.Z2(+)=S.Z2;
```

Nowe możliwości Oracle SQL inspirowane przez Standard SQL:

```
select X, R.Z1, R.Z2, Y from R full join S  
on R.Z1=S.Z1 and R.Z2=S.Z2;
```

```
select X, R.Z1, R.Z2, Y from R full outer join S  
on R.Z1=S.Z1 and R.Z2=S.Z2;
```

A nawet:

```
select X, Z1, Z2, Y from R natural full join S;
```

```
select X, Z1, Z2, Y from R natural full outer join S;
```

# Uwaga o implementacji złączenia naturalnego w Oracle SQL

Złączenie naturalne relacji w Oracle SQL odbywa się względem atrybutów współdzielonych o takich samych nazwach, gdy takie nie istnieją, to wykonany zostaje iloczyn kartezyjański relacji.

Dozwolone jest również wykorzystanie złączeń naturalnych do wykonania złączeń zewnętrznych relacji. Jednak aby uzyskać taki wynik, muszą istnieć w tych relacjach współdzielone atrybuty o takich samych nazwach, w przeciwnym razie zostanie wykonany iloczyn kartezyjański.



# Dzielenie

Niech relacje  $R$  i  $S$  mają nagłówki odpowiednio  $N(R)=\{X_1, X_2, \dots, X_n, Z_1, Z_2, \dots, Z_k\}$  i  $N(S)=\{Z_1, Z_2, \dots, Z_k\}$ , tzn.  $N(S) \subset N(R)$ .

Relacja  $T = R \div S$  o nagłówku  $N(T) = \{X_1, X_2, \dots, X_n\}$  jest rezultatem dzielenia  $R$  przez  $S$ , jeżeli  $T$  zawiera dokładnie te krotki  $t$ , dla których spełniony jest warunek:

$$\bigvee R_s \subset R: |R_s| = |S| \wedge |R_s[X_1, X_2, \dots, X_n]| = 1 \wedge$$

$$\wedge \left( \bigwedge \text{krotki } s \in S \bigvee r \in R_s: r(Z_i) = s(Z_i) \wedge i: 1 \leq i \leq k \right)$$

wówczas  $\bigvee$  pojedyncza krotka  $t \in R \div S: t(X_1, X_2, \dots, X_n)$ .

W Oracle SQL nie istnieje operator dzielenia.

# Dzielenie

*R:*

X	Y	Z
x1	y1	z1
x2	y1	z1
x1	y2	z1
x1	y2	z2
x2	y1	z2

*S1:*

Z
z1

*R ÷ S1:*

X	Y
x1	y1
x2	y1
x1	y2

*S4:*

Z
z1
z2

*R ÷ S4:*

X	Y
x2	y1
x1	y2

*S2:*

Y	Z
y1	z1

*R ÷ S2:*

X
x1
x2

*S3:*

Y	Z
y1	z1
y2	z2

*R ÷ S3:*

X
x1

# Przydatne własności

## Prawo łączności:

iloczynu kartezjańskiego:  $(R \times S) \times T = R \times (S \times T)$

złączenia:  $(R \triangleright \triangleleft S) \triangleright \triangleleft T = R \triangleright \triangleleft (S \triangleright \triangleleft T)$

## Prawo przemienności:

iloczynu kartezjańskiego:  $R \times S = S \times R$

złączenia:  $R \triangleright \triangleleft S = S \triangleright \triangleleft R$

# PRIORYTETY DZIAŁANIA OPERATORÓW

NAJWYŻSZY



NAJNIŻSZY

PROJEKCJA

SELEKCJA

ILOCZYN KARTEZJAŃSKI

ZŁĄCZENIE, DZIELENIE

RÓŻNICA

PRZECIĘCIE, UNIA

Przykładowo:  $R[X, Y]$  gdzie  $Z=5$

jest **niepoprawne** (wynik projekcji  $R[X, Y]$  nie ma już atrybutu  $Z$ ).

Poprawny zapis to:  $(R \text{ gdzie } Z=5)[X, Y]$

Oczywiście niekiedy konieczne jest STOSOWANIE  
NAWIASÓW.

# Zupełny zbiór działań

ZUPEŁNY ZBIÓR DZIAŁAŃ ALGEBRY RELACYJNYCH  
BAZ DANYCH tworzą:

- UNIA
- RÓŻNICA
- PRODUKT (ILOCZYN KARTEZJAŃSKI)
- PROJEKCJA (RZUTOWANIE)
- SELEKCJA
- ALIAS (PRZYPISANIE)

Przy pomocy tych 6 działań można wyrazić pozostałe 3:  
PRZECIĘCIE (ILOCZYN), ZŁĄCZENIE, DZIELENIE.

# Zupełny zbiór działań

ZUPEŁNY ZBIÓR DZIAŁAŃ ALGEBRY RELACYJNYCH  
BAZ DANYCH tworzą:

- UNIA
- RÓŻNICA
- PRODUKT (ILOCZYN KARTEZJAŃSKI)
- PROJEKCJA (RZUTOWANIE)
- SELEKCJA
- ALIAS (PRZYPISANIE)

Przy pomocy tych 6 działań można wyrazić pozostałe 3:  
PRZECIĘCIE (ILOCZYN), ZŁĄCZENIE, DZIELENIE.

## Wybrane podręczniki

1. C. J. Date, Wprowadzenie do systemów baz danych, WNT, Warszawa 2001.
2. J. D. Ullman, Jennifer Widom, Podstawowy wykład z systemów baz danych, WNT, Warszawa 2000.
3. H. Garcia-Molina, J.D. Ullman, J. Widom, Systemy baz danych. Pełny wykład, WNT, Warszawa 2006.
4. J. Widom , H. Garcia-Molina , J. D. Ullman Implementacja systemów baz danych, WNT Warszawa 2003.
5. T. M. Connolly, C. E. Begg, Systemy baz danych, Warszawa, RM 2004.
6. S. Allen, Modelowanie danych, Helion, Gliwice 2006
7. T.Pankowski; Podstawy baz danych, PWN, Warszawa 1992.
8. J. Kukuczka, Relacyjne bazy danych, Wydawnictwo Pracowni Komputerowej Jacka Skalmierskiego, Gliwice 2000.
9. K. Czapla, Bazy danych. Podstawy projektowania i języka SQL, Helion, Gliwice 2015.
10. H. Landayi, SQL. Księga eksperta, Wydawnictwo Helion, Gliwice 2000.