

# Metody Bazodanowe i Bazy Danych - zadania

dr Piotr Syga (**A**wesome **R**adiator **M**an)



## Lista 1

**Zadanie 1.** Podaj wszystkie miejsca, w których stoczono jakiekolwiek bitwy.

*Rozwiązanie.*

```
1 SELECT DISTINCT Miejsce
2 FROM Bitwy;
```

**Zadanie 2.** Wybierz wszystkich zwycięzców bitew przed 5 rokiem naszej ery.

*Rozwiązanie.*

```
1 SELECT DISTINCT Zwyciezca
2 FROM Bitwy
3 WHERE Rok < 5;
```

**Zadanie 3.** Wybierz bitwy, w których wygrała Macedonia między rokiem 400 a 320.

*Rozwiązanie.*

```
1 SELECT Bitwa
2 FROM Bitwy
3 WHERE Zwyciezca = 'Macedonia' AND ROK BETWEEN -400 AND -320;
```

**Zadanie 4.** Podaj bitwy wraz ze stronami i dowódcami, w których uczestniczyło więcej żołnierzy niż w bitwie nad rzeką Fei.

*Rozwiązanie.*

```
1 SELECT bs1.Bitwa, bs1.Strona, bs2.Strona, bs1.Dowodca, bs2.Dowodca
2 FROM Bitwy_Szczegoly bs1 INNER JOIN Bitwy_Szczegoly bs2
3   ON bs1.Bitwa = bs2.Bitwa AND bs1.Strona < bs2.Strona
4 WHERE (bs1.Liczebnosc + bs2.Liczebnosc) >
5   (SELECT (bs1.Liczebnosc + bs2.Liczebnosc
6 FROM Bitwy_Szczegoly bs1 INNER JOIN Bitwy_Szczegoly bs2
7   ON bs1.Bitwa = bs2.Bitwa
8   AND bs1.Strona < bs2.Strona WHERE bs1.Bitwa = 'Nad_rzeka_Fei'))
```

**Zadanie 5.** Znajdź wszystkie miejsca, w których odbyła się więcej niż jedna bitwa.

*Rozwiązanie.*

```
1 SELECT Miejsce, COUNT(Miejsce) n
2 FROM Bitwy
3 GROUP BY Miejsce
4 HAVING n>1;
```

**Zadanie 6.** Podaj wszystkie narodowości, które uczestniczyły w jakiegokolwiek bitwie.

*Rozwiązanie.*

```
1 SELECT DISTINCT Kraj
2 FROM (SELECT Zwyciezca AS Kraj FROM Bitwy) UNION (SELECT Przegrany AS Kraj FROM Bitwy)
```

**Zadanie 7.** Wybierz osoby, które dowodziły inną stroną konfliktu niż posiadana przez niego narodowość.

*Rozwiązanie.*

```
1 SELECT Ludzie.Osoba
2 FROM Ludzie
3 JOIN Bitwy_Szczegoly ON bs.Dowodca = Ludzie.Osoba
4 WHERE Bitwy_Szczegoly.Strona <> Ludzie.Narodowosc;
```

**Zadanie 8.** Wskaż bitwy (wraz z miejscem i datą) oraz strony konfliktu, w których sojusz zwycięski dysponował mniejszymi siłami niż przegrany.

*Rozwiązanie.*

```
1 SELECT b.Bitwa, b.Miejsce, b.Rok, b.Zwyciezca, b.Przegrany
2 FROM Bitwy b
3 INNER JOIN Bitwy_Szczegoly bs1 ON b.Bitwa = bs1.Bitwa AND b.Zwyciezca = bs1.Strona
4 INNER JOIN Bitwy_Szczegoly bs2 ON b.Bitwa = bs2.Bitwa AND b.Przegrany = bs2.Strona
5 WHERE bs1.Liczebosc < bs2.Liczebosc;
```

**Zadanie 9.** Wskaż wszystkich dowódców przed rokiem 1454, którzy nosili przydomek Wielki.

*Rozwiązanie.*

```
1 SELECT DISTINCT Bitwy_Szczegoly.Dowodca
2 FROM Bitwy_Szczegoly
3 INNER JOIN Bitwy ON Bitwy_Szczegoly.Bitwa = Bitwy.Bitwa
4 WHERE ROK < 1454 AND Bitwy_Szczegoly.Dowodca LIKE '%Wielki%'
```

**Zadanie 10.** Podaj wszystkie osoby, które nie dowodziły w żadnej bitwie.

*Rozwiązanie.*

```
1 SELECT Ludzie.Osoba
2 FROM Ludzie
3 LEFT JOIN Bitwy_Szczegoly ON Ludzie.Osoba = Bitwy_Szczegoly.Dowodca
4 WHERE Bitwy_Szczegoly.Dowodca IS NULL;
```

lub:

```

1 SELECT l.Osoba
2 FROM ((SELECT Osoba From Ludzie) EXCEPT
3 (SELECT Dowodca AS Osoba FROM Bitwy_Szczegoly) AS l)

```

**Zadanie 11.** Wskaż bitwy wraz z ich datą i miejscem, w których uczestniczył sojusz składający się z więcej niż 4 członków.

*Rozwiązanie.*

```

1 SELECT b.Bitwa, b.Miejsce, b.Rok
2 FROM
3 (
4     SELECT Bitwa, Count(Panstwo2)+1 AS Counter
5     FROM Sojusze AS S
6     GROUP BY Panstwo1, Bitwa
7     INNER JOIN Bitwa AS B
8     ON B.Bitwa = S.Bitwa
9 )
10 HAVING Counter >= 4;

```

**Zadanie 12.** Podaj bitwy i państwa, które były w nich sprzymierzone zarówno ze zwycięzcą jak i z przegranym

*Rozwiązanie.*

```

1 SELECT b.Bitwa AS Bitwa, s.Panstwo2 AS Sojusznik
2 FROM Bitwy AS B
3 INNER JOIN Bitwy_Szczegoly AS BS ON B.Bitwa = BS.Bitwa AND B.Zwyciezca = BS.Strona
4 INNER JOIN Sojusze AS S ON S.Panstwo1 = BS.Strona AND S.Bitwa = B.Bitwa
5 INTERSECT
6 SELECT B.Bitwa AS Bitwa, S.Panstwo2 AS Sojusznik
7 FROM Bitwy AS B
8 INNER JOIN Bitwy_Szczegoly AS BS ON B.Bitwa = BS.Bitwa AND B.Przegraný = BS.Strona
9 INNER JOIN Sojusze AS S ON S.Panstwo1 = BS.Strona AND S.Bitwa = B.Bitwa

```

lub:

```

1 SELECT S1.Bitwa, S1.Panstwo2
2 FROM Sojusze S1 JOIN Sojusze S2 ON S1.Bitwa = S2.Bitwa AND S1.Panstwo2 = S2.Panstwo2
3 AND S1.Panstwo1 < S2.Panstwo2

```

**Zadanie 13.** Wybierz dowódców oraz dowodzone przez nich strony konfliktu, którzy w swojej karierze zarówno wygrali jak i przegrali z danym przeciwnikiem.

*Rozwiązanie.*

```

1 SELECT bs.Dowodca, bs.Strona
2 FROM ((SELECT bs.Dowodca FROM Bitwy_Szczegoly bs INNER JOIN Bitwy AS B
3 ON B.Bitwa = bs.Bitwa AND bs.Strona = B.Zwyciezca)
4 INTERSECT
5 SELECT bs.Dowodca FROM Bitwy_Szczegoly bs INNER JOIN Bitwy AS B
6 ON B.Bitwa = bs.Bitwa AND bs.Strona = B.Przegraný) AS bs

```

**Zadanie 14.** Podaj miejsca, w których walczył zarówno dowolny Cezar jak i dowolny Napoleon. Zadbaj o brak powtórzeń.

*Rozwiązanie.*

```
1 SELECT DISTINCT b.Miejsce
2 FROM Bitwy b
3 INNER JOIN Bitwy_Szczegoly bs ON b.Bitwa = bs.Bitwa AND bs.Zwyciezca = b.Strona
4 INNER JOIN Bitwy_Szczegoly bs1 ON b.Bitwa = bs1.Bitwa AND bs1.Przegraný = b.Strona
5 WHERE bs.Dowodca LIKE '%Cezar%' OR bs1.Dowodca LIKE '%Cezar'
6 INTERSECT
7 SELECT DISTINCT b.Miejsce
8 FROM Bitwy b
9 INNER JOIN Bitwy_Szczegoly bs ON b.Bitwa = bs.Bitwa AND bs.Zwyciezca = b.Strona
10 INNER JOIN Bitwy_Szczegoly bs1 ON b.Bitwa = bs1.Bitwa AND bs1.Przegraný = b.Strona
11 WHERE bs.Dowodca LIKE '%Napoleon%' OR bs1.Dowodca LIKE '%Napoleon'
```

**Zadanie 15.** Podaj miejsce, rok oraz dowódcę zwycięskiej strony, najstarszej zarejestrowanej bitwy.

*Rozwiązanie.*

```
1 SELECT b.Miejsce, b.Rok, bs.Dowodca
2 FROM Bitwy b
3 INNER JOIN Bitwy_Szczegoly bs ON b.Zwyciezca = b.Strona AND b.Bitwa = bs.Bitwa
4 WHERE b.Rok = (SELECT MIN(ROK) FROM Bitwy);
```

**Zadanie 16.** Wybierz miejsca, w których odstęp między kolejnymi bitwami był mniejszy niż 10 lat. Wyniki posortuj alfabetycznie. (Co zmieni się jeśli chcemy sortować według długości odstępów?)

*Rozwiązanie.*

```
1 SELECT DISTINCT b.Miejsce
2 FROM Bitwy b
3 INNER JOIN Bitwy c ON b.Miejsce = c.Miejsce
4 WHERE (b.Rok - c.Rok) < 10 AND b.Bitwa < c.Bitwa
5 ORDER BY b.Miejsce
```

**Zadanie 17.** Podaj dowódcę, który wygrał największą liczbę bitew.

*Rozwiązanie.*

```
1 SELECT bs.Dowodca, COUNT(b.Zwyciezca)
2 FROM Bitwy_Szczegoly bs
3 INNER JOIN Bitwy b ON b.Bitwa = bs.Bitwa AND b.Zwyciezca = B.Strona
4 GROUP BY bs.Dowodca
5 ORDER BY COUNT(b.Zwyciezca)
```

**Zadanie 18.** Podaj wszystkie sojusze w historii strony, która nie przegrała żadnej z bitew.

*Rozwiązanie.*

```

1 SELECT Panstwo1, Panstwo2
2 FROM Sojusze
3 WHERE Panstwo1 NOT IN
4     (SELECT Panstwo1
5      FROM Bitwy b
6      LEFT JOIN Sojusze s ON b.Bitwa = s.Bitwa
7      WHERE b.Przeegrany = s.Panstwo1)

```

## Lista 2

**Zadanie 1.** Wybierz wszystkich muzyków, którzy grają w zagranicznych zespołach.

*Rozwiązanie.*

SQL:

```

1 SELECT m.imie, m.nazwisko FROM Muzycy m JOIN Zespoly z ON
2 m.zespol = z.ID WHERE m.narodowosc <> z.kraj;

```

AR:

$$\pi_{m.imie, m.nazwisko}(\sigma_{m.narodowosc \neq z.kraj}(\rho_m(Muzycy) \bowtie_{m.zespol = z.ID} \rho_z(Zespoly))))$$

**Zadanie 2.** Wybierz wszystkie pary zespołów oraz kraje ich pochodzenia, które nagrały album o tej samej nazwie.

*Rozwiązanie.*

SQL:

```

1 SELECT DISTINCT za.nazwa, za.kraj, zb.nazwa, zb.kraj
2 FROM
3     (SELECT z.nazwa, z.kraj, a.nazwa as album
4      FROM Zespoly z JOIN Albumy a ON z.ID = a.zespol) za
5 JOIN
6     (SELECT z.nazwa, z.kraj, a.nazwa as album
7      FROM Zespoly z JOIN Albumy a ON z.ID = a.zespol) zb
8 ON za.nazwa > zb.nazwa and za.album = zb.album;

```

AR:

$$\begin{aligned} & \delta(\pi_{za.nazwa, za.kraj, zb.nazwa, zb.kraj} \\ & (\rho_{za}(\pi_{z.nazwa, z.kraj, album}(\rho_{album} / a.nazwa(\rho_z(Zespoly) \bowtie_{z.ID=a.zespol} \rho_a(Albumy)))) \\ & \bowtie_{za.nazwa > zb.nazwa \text{ AND } za.album = zb.album} \\ & \rho_{zb}(\pi_{z.nazwa, z.kraj, album}(\rho_{album} / a.nazwa(\rho_z(Zespoly) \bowtie_{z.ID=a.zespol} \rho_a(Albumy))))) \end{aligned}$$

**Zadanie 3.** Wybierz wszystkie nazwy krajów pojawiające się w bazie danych.

*Rozwiązanie.*

SQL:

```

1 (SELECT kraj FROM Zespoly)
2 UNION
3 (SELECT narodowosc AS kraj FROM Muzycy)
4 UNION
5 (SELECT kraj FROM Koncerty);

```

AR:

$$\pi_{kraj}(Zespoły) \cup \rho_{kraj / narodowosc}(\pi_{narodowosc}(Muzycy)) \cup \pi_{kraj}(Koncerty)$$

**Zadanie 4.** Wybierz wszystkie utwory, w których nazwie pojawia się podciąg „part” lub „część”. W schemacie wynikowym uwzględnij nazwy zespołów oraz albumy powiązane z tymi utworami.

*Rozwiązanie.*

SQL:

```
1 SELECT u.tytul, a.nazwa, z.nazwa
2 FROM Utwory u JOIN Albumy a ON u.album = a.nazwa JOIN Zespoły z ON u.zespol = z.ID
3 WHERE u.tytul LIKE "%part%" or u.tytul LIKE "%czesc%";
```

AR:

$$\pi_{u.tytul, a.nazwa, z.nazwa}(\sigma_{u.tytul \text{ LIKE } \%part\% \text{ OR } u.tytul \text{ LIKE } \%czesc\%}(\rho_u(Utwory) \bowtie_{u.zespol = z.ID} \rho_a(Albumy) \bowtie_{u.album = a.nazwa} \rho_z(Zespoły)))$$

**Zadanie 5.** Wybierz zespoły, w których żaden członek zespołu nie przekroczył 50 roku życia.

*Rozwiązanie.*

SQL:

```
1 (SELECT nazwa FROM Zespoły)
2 EXCEPT
3 (SELECT DISTINCT nazwa
4 FROM (SELECT z.nazwa, m.imie FROM Zespoły z JOIN Muzycy m ON z.ID = m.zespol
5 WHERE m.wiek > 50) a);
```

AR:

$$\pi_{nazwa}(Zespoły) \setminus \delta(\pi_{nazwa}(\pi_{z.nazwa, m.imie}(\sigma_{m.wiek > 50}(\rho_z(Zespoły) \bowtie_{z.ID = m.zespol} \rho_m(Muzycy)))))$$

**Zadanie 6.** Wybierz wszystkich muzyków wraz z nazwą ich zespołu, którzy nagrali utwór dłuższy niż „Fire and Ice”.

*Rozwiązanie.*

SQL:

```
1 SELECT DISTINCT m.imie, m.nazwisko, z.nazwa
2 FROM Zespoły z JOIN Utwory u ON z.ID = u.zespol JOIN Muzycy m ON z.ID = m.zespol
3 WHERE u.czas > (SELECT czas FROM Utwory WHERE tytul = "Fire_and_Ice");
```

AR:

$$A = \pi_{czas}(\sigma_{tytul = \text{Fire and Ice}}(Utwory))$$

$$\delta(\pi_{m.imie, m.nazwisko, z.nazwa}(\sigma_{u.czas > A}(\rho_z(Zespoły) \bowtie_{z.ID = u.zespol} \rho_u(Utwory) \bowtie_{z.ID = m.zespol} \rho_m(Muzycy)))))$$

**Zadanie 7.** Wybierz zespoły, które w ciągu ostatnich 3 lat nie wydały żadnego albumu ani nie zagrały koncertu.

*Rozwiązanie.*

SQL:

```
1 (SELECT z.nazwa
2 FROM Zespoły z JOIN Albumy a ON z.ID = a.zespol
3 WHERE NOW() < DATEADD(year, 3, CONVERT(datetime, a.wydanie)))
4 INTERSECT
```

```

5 (SELECT z.nazwa
6 FROM Zespoly z JOIN Koncerty k ON z.ID = k.zespol
7 WHERE NOW() < DATEADD(year, 3, k.data));

```

AR:

$$A = \pi_{z.nazwa}(\sigma_{NOW() < DATEADD(year, 3, CONVERT(datetime, a.wydanie))}(\rho_z(Zespoly) \bowtie_{z.ID = a.zespol} \rho_a(Albumy)))$$

$$B = \pi_{z.nazwa}(\sigma_{NOW() < DATEADD(year, 3, CONVERT(datetime, k.data))}(\rho_z(Zespoly) \bowtie_{z.ID = k.zespol} \rho_k(Koncerty)))$$

$$A \cap B$$

**Zadanie 8.** Wybierz wszystkie utwory, w których nagrywaniu brał udział przynajmniej jeden muzyk starszy niż Johan Edlund.

*Rozwiązanie.*

SQL:

```

1 SELECT DISTINCT u.tytul
2 FROM Utwory u JOIN Muzycy m ON u.zespol = m.zespol
3 WHERE m.wiek > (SELECT wiek FROM Muzycy WHERE imie = "Johan" and nazwisko = "Edlund");

```

AR:

$$A = \pi_{wiek}(\sigma_{imie = Johan \text{ AND } nazwisko = Edlund}(Muzycy))$$

$$\delta(\pi_{u.tytul}(\sigma_{m.wiek > A}(\rho_u(Utwory) \bowtie_{u.zespol = m.zespol} \rho_m(Muzycy))))$$

**Zadanie 9.** Wybierz imię, które nosi największa liczba muzyków.

*Rozwiązanie.*

SQL:

```

1 SELECT imie
2 FROM
3     (SELECT imie, count(imie) wystapienia
4     FROM Muzycy group by imie) A
5 WHERE A.wystapienia = (SELECT max(wystapienia) FROM A);

```

AR:

*a*

**Zadanie 10.** Wybierz wszystkie zespoły, które zagrały więcej koncertów we Włoszech niż zespół, w którym występuje Cristina Scabbia.

*Rozwiązanie.*

SQL:

```

1 SELECT DISTINCT nazwa
2 FROM
3     (SELECT z.nazwa, m.imie, m.nazwisko, count(k.zespol, k.data)
4     FROM Zespoly z JOIN Koncerty k ON z.ID = k.zespol JOIN Muzycy m ON z.ID = m.zespol
5     WHERE k.kraj = "Wlochy" group by k.zespol, k.data) A
6 WHERE A.razy > (SELECT razy FROM A WHERE imie = "Cristina" and nazwisko = "Scabbia");

```

AR:

*a*

**Zadanie 11.** Wypisz wszystkie zespoły z USA, w których nie występują muzycy z Meksyku ani Niemiec.

*Rozwiązanie.*

SQL:

```
1 (SELECT ID
2 FROM Zespoly
3 WHERE kraj = "USA")
4 EXCEPT
5 (SELECT DISTINCT ID
6 FROM Zespoly z JOIN Muzycy m ON z.ID = m.zespol
7 WHERE m.kraj = "Meksyk" or m.kraj = "Niemcy");
```

AR:

*a*

**Zadanie 12.** Wybierz wszystkie albumy, na których jest przynajmniej 7 utworów. Wyniki posortuj malejąco względem łącznego czasu trwania wszystkich utworów na albumie.

*Rozwiązanie.*

SQL:

```
1 SELECT c.nazwa FROM
2     (SELECT a.nazwa, b.trwanie, b.liczba
3     FROM
4         (SELECT nazwa, count(Utwory.tytul) liczba, sum(czas) trwanie
5         FROM Albumy JOIN Utwory ON Albumy.nazwa = Utwory.tytul
6         group by Albumy.nazwa) b WHERE liczba > 7
7     ) c
8 order by c.trwanie desc;
```

AR:

$$A = \pi_{a.nazwa, b.trwanie, b.liczba}(\sigma_{liczba > 7}(??))$$
$$\tau_{c.trwanie}(\pi_{c.nazwa}(\rho_c(A)))$$