

## Podzapytania

Podzapytanie jest zapytaniem zagnieżdżonym w innym zapytaniu. Podzapytanie może zostać za-

- WHERE
- HAVING
- SELECT
- FROM

Podzapytania dzielimy na zwykłe i skorelowane.

### Podzapytanie zagnieżdżone w klauzuli WHERE

```
SELECT wyr_A1, ...  
FROM rel_A1  
WHERE wyr_A3 operator  
      (SELECT wyr_B1  
       FROM rel_B1  
       WHERE ...)  
ORDER BY wyr_A4;
```

- Podzapytanie wierszowe:

- podzapytanie zwraca zawsze co najwyżej jedną krotkę
- dopuszczalne operatory logiczne: =, !=, <>, >, >=, <, <=
- Przykład 1: *Podaj nazwisko pracownika otrzymującego najniższą płacę podstawową.*

```
SELECT nazwisko FROM pracownicy  
WHERE placa_pod = (SELECT MIN(placa_pod) FROM pracownicy);
```

- Przykład 2: *Podaj nazwisko profesora otrzymującego najniższą płacę podstawową wśród profesorów.*

```
SELECT nazwisko FROM pracownicy  
WHERE (placa_pod, etat) =  
      (SELECT MIN(placa_pod), 'PROFESOR'  
       FROM pracownicy  
       WHERE etat = 'PROFESOR');
```

- Podzapytanie tablicowe:

- dopuszczamy wiele krotek w podzapytaniu
- dopuszczane operatory to: IN, ANY, ALL
- Przykład 1: *Podaj nazwiska pracowników otrzymujących najwyższe płace podstawowe w swoich grupach etatowych.*

```
SELECT nazwisko FROM pracownicy
WHERE (etat,placa_pod) IN
      (SELECT etat, MAX(placa_pod) FROM pracownicy GROUP BY etat);
```

- Przykład 2 (ANY): *Podaj nazwiska pracowników, których płaca podstawowa jest większa od płacy podstawowej jakiegoś pracownika zespoły 10.*
- Przykład 3 (ALL): *Podaj nazwiska pracowników, których płaca podstawowa jest większa od płacy podstawowej wszystkich pracowników zespoły 10.*

### Podzapytania w klauzuli HAVING.

- Przykład: *Podaj nazwy i średnie płace w zespołach, w których średnia płaca przekracza średnią płacę wśród wszystkich pracowników.*

```
SELECT nazwa, AVG(placa_pod) AS srednia
FROM pracownicy NATURAL JOIN zespoly
GROUP BY nazwa
HAVING AVG(placa_pod) > (SELECT AVG(placa_pod) FROM pracownicy);
```

### Podzapytania skorelowane.

- W podzapytaniu znajduje się odwołanie do wyrażenia z zapytania zewnętrznego.

```
SELECT wyr_A1, ...
FROM rel_A1
WHERE wyr_A2 operator
      (SELECT wyr_B1
       FROM rel_B1
       WHERE wyr_B2 = wyr_A_3)
ORDER BY wyr_A4;
```

- Przykład: *Podaj nazwiska pracowników zarabiających więcej niż średnia płaca w ich grupie etatowej.*

```
SELECT nazwisko FROM pracownicy p
WHERE placa_pod > (SELECT AVG(placa_pod) FROM pracownicy WHERE etat = p.etat);
```

- Operator EXISTS

- przyjmuje wartość prawdy, gdy podzapytanie zwróci przynajmniej jedną krotkę
- Przykład: *Podaj nazwiska pracowników, którzy posiadają podwładnych.*

```
SELECT nazwisko FROM pracownicy p
WHERE EXISTS (SELECT * FROM pracownicy WHERE id_szefa = p.id_prac);
```

## Podzapytania w klauzuli SELECT.

- Podzapytanie musi zwrócić dokładnie jedną wartość dla każdego zapytania zewnętrznego.

– Przykład: *Dla każdego zespołu podaj jego nazwę i średnią płacę podstawową pracowników w zespole.*

```
SELECT nazwa,  
       (SELECT AVG(placa_pod) FROM pracownicy WHERE id_zesp = z.id_zesp) as srednia_placa  
FROM zespoly z;
```

## Podzapytanie w klauzuli FROM.

- Podzapytanie tworzy zbiór danych (relację) dla zapytania zewnętrznego.

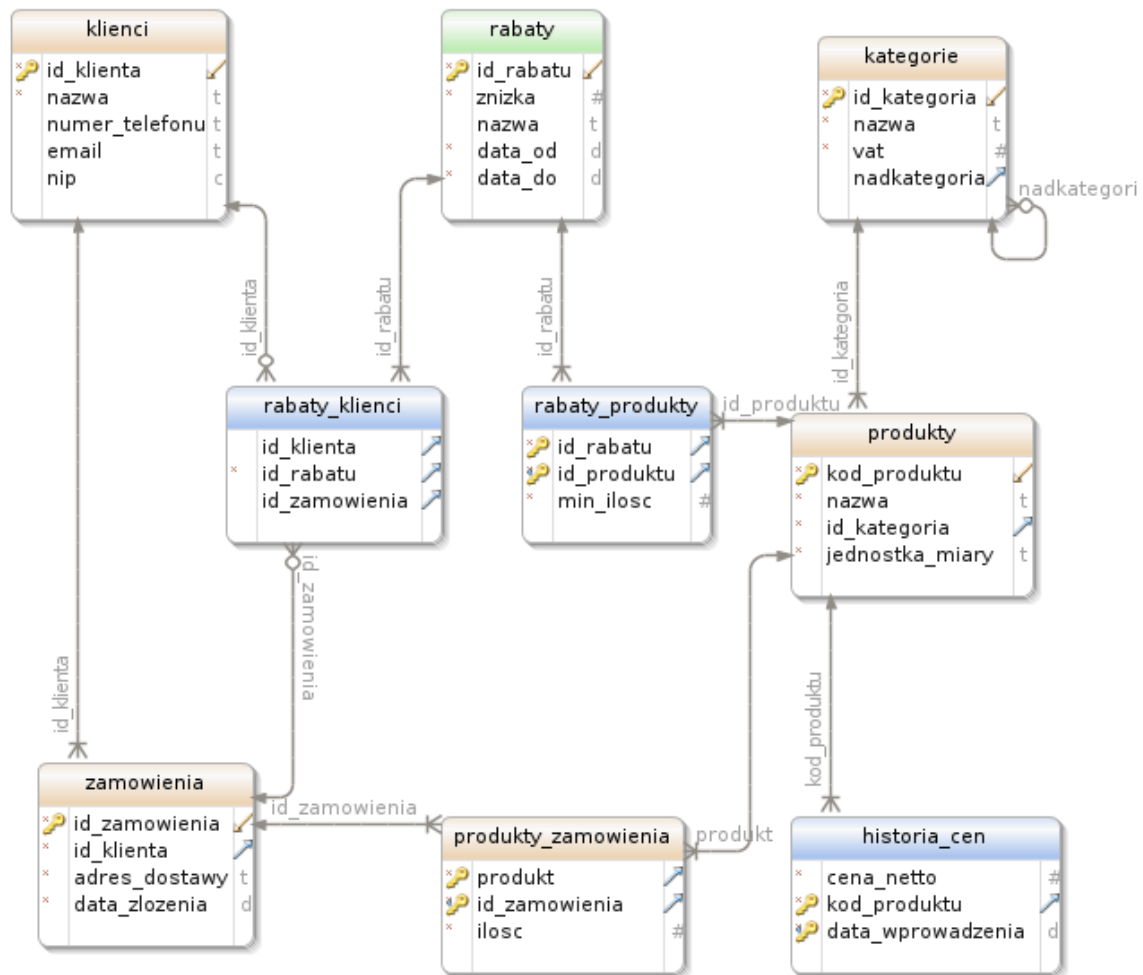
– Przykład: *Dla każdego zespołu podaj jego nazwę i średnią płacę podstawową pracowników w zespole.*

```
SELECT nazwa, srednia_placa  
FROM zespoly NATURAL JOIN  
     (SELECT id_zesp, AVG(placa_pod) as srednia_placa  
      FROM pracownicy  
      GROUP BY id_zesp) srednie;
```

## Zadania

Poniższe zadania dotyczą schematu bazy sklep-schemat.sql (należy uwzględnić również plik sklep-fix.sql) przedstawionego na poniższym rysunku. Przykładowe dane znajdują się w pliku sklep-dane.sql. Pliki należy wczytać w kolejności:

- sklep-schemat.sql
- sklep-dane.sql
- sklep-fix.sql



Generated using DbSchema

**Uwaga 1:** W poniższych zadaniach nie jest wymagane sortowanie danych jeśli zadanie tego nie precyzuje. Wszystkie liczby zmiennoprzecinkowe należy wypisywać z dokładnością do dwóch miejsc po przecinku. W niektórych zadaniach zabronione jest używanie pewnych słów.

**Uwaga 2:** We wszystkich zadaniach (poza 3c) należy używać podzapytań. We wszystkich zadaniach, poza 3c, 8, 10 i 12, należy użyć dokładnie jednego podzapytania.

1. Wypisz nazwy wszystkich produktów, które należą do tej samej kategorii co 'Piórnik duży'. Możesz założyć, że w bazie znajduje się tylko jeden 'Piórnik duży'.
2. Podaj wszystkie nazwy produktów, których kategoria jest podkategorią większej kategorii.
3. Podaj nazwy kategorii, które reprezentują co najmniej 3 produkty. Zadanie należy rozwiązać w trzech wersjach. Porównaj ich plany wykonań.
  - ZAD3a: Użyj podzapytania w klauzuli WHERE.
  - ZAD3b: Użyj podzapytania w klauzuli SELECT.

- ZAD3c: Nie używaj podzapytań.
4. Wypisz wszystkie informacje (\*) o tych rabatach, które najdłużej obowiązywały.
  5. Wyświetl wszystkie informacje (\*) o kategoriach, których VAT jest niższy niż VAT któregoś z jego bezpośrednich podkategorii. Użyj słowa kluczowego ANY.
  6. Podaj wszystkie zamówienia (wypisz `id_zamowienia`), które zawierają najwięcej zakupionych produktów. W tym zadaniu zabronione jest używanie słowa kluczowego ORDER BY.
  7. Podaj nazwy produktów, których cena zmieniła się (w tym została wprowadzona) w kwietniu co najmniej 2 razy. Liczbę owych zmian cen wypisz w drugiej kolumnie.
  8. Dla każdej kategorii wypisz jej najwyższą nadkategorię. Jeżeli kategoria nie posiada nadkategorii przyjmij, że sama jest swoją najwyższą nadkategorią. W wyniku mają pojawić się dwie kolumny: nazwa kategorii i nazwa najwyższej nadkategorii.
  9. Dla każdego produktu, podaj jego nazwę oraz jego ostatnią cenę brutto. W tym zadaniu zabronione jest używanie słowa kluczowego ORDER BY.
  10. Dla każdego produktu, podaj jego nazwę oraz jego ostatnią cenę brutto, uwzględniając wszystkie możliwe rabaty aktywne w momencie ostatniej zmiany ceny. Zniżka rabatu dotyczy ceny brutto i określa wartość bezwzględną (nie procentową). W tym zadaniu nie uwzględniaj wymagania ilości produktów w rabacie.
  11. Dla każdego zamówienia, wypisz jego id oraz jego ówczesną cenę transakcji brutto, nie uwzględniając rabatów.
  12. Dla każdego zamówienia, wypisz jego id oraz jego ówczesną cenę transakcji brutto, uwzględniając wszystkie możliwe rabaty. Uwzględnij następujące zależności:
    - rabaty klientów i produktów potraktuj niezależnie, tzn. dany rabat może być jednocześnie zaliczony do rabatu produktu jak i rabatu klienta,
    - rabat klienta jest uwzględniony jeśli pole `id_zamowienia` w tabeli `rabaty_klienci` wskazuje na rozpatrywane zamówienie, wtedy (zapewnienie dotyczące danych) również pole `id_klienta` z tabeli `rabaty_klienci` jest równe (lub wynosi null) polu `id_klienta` z tabeli `zamowienia`,
    - wszystkie rabaty dotyczą ceny brutto i określają wartość bezwzględną (nie procentową),
    - rabat  $r$  jest uznany jeśli data  $d$  realizacji zamówienia spełnia warunek  $r.data\_od \leq d \leq r.data\_do$  oraz (tylko w przypadku produktu) warunek ilości danego produktu,
    - przyjmij, że w danym zamówienie każdy produkt pojawia się co najwyżej jeden raz.