

Zapytania zagnieżdżone

Zapytania zagnieżdżone (podzapytania, subqueries)

```
SELECT (SELECT MAX(przebieg) FROM auto) as max;
```

Podzapytania zwracające pojedyncze wyniki niezależnych zapytań.

```
SELECT (SELECT MAX(przebieg) FROM auto) AS max,  
(SELECT MAX(przebieg) FROM auto WHERE marka = 'bmw') AS max_bmw,  
(SELECT MAX(przebieg) FROM auto WHERE kolor = 'czarny') AS max_czarny;
```

Podzapytania pozwalają pisać równoważniki zapytań.

```
SELECT * FROM auto WHERE marka = 'mazda' and kolor = 'czarny';
```

Można również tak:

```
SELECT * FROM auto WHERE id IN (SELECT id FROM auto WHERE marka = 'mazda' and  
kolor = 'czarny');
```

Zapytania zagnieżdżone można stosować do wszystkich poleceń DML. Zapytania takie składają się z zapytania zewnętrznego i wewnętrznego. **Przykłady z IN:**

```
SELECT * FROM auto WHERE auto.fk_id_komis IN (SELECT id FROM komis WHERE  
komis.id BETWEEN 0 AND 5);
```

```
SELECT * FROM auto WHERE auto.fk_id_komis IN (SELECT id FROM komis WHERE  
komis.wojew IN('opolskie', 'podlaskie'));
```

```
SELECT * FROM auto WHERE id IN (SELECT id FROM auto ORDER BY cena_pln) LIMIT 10;
```

Przykłady z NOT IN:

```
SELECT * FROM auto WHERE auto.fk_id_wlas NOT IN (SELECT id FROM wlasciciel WHERE  
nazwa = 'BARAN');
```

```
SELECT * FROM auto WHERE auto.fk_id_wlas NOT IN (SELECT id FROM wlasciciel WHERE  
nazwa IN ('Baran', 'Inny', 'Duda'));
```

Przykłady z WHERE:

```
SELECT * FROM auto WHERE cena_eu < (SELECT AVG(cena_eu) FROM auto) ORDER BY  
cena_eu limit 10;
```

```
SELECT * FROM wlasciciel WHERE wlasciciel.fk_id_komis = (SELECT id FROM komis  
WHERE komis.nazwa = 'asdhajd');
```

Podzapytania skolerowane.

```
SELECT * FROM auto WHERE cena_pln < (SELECT AVG(cena_pln) FROM auto);
```

```
SELECT * FROM auto AS tmp WHERE cena_pln < (SELECT AVG(cena_pln) FROM auto  
WHERE marka = tmp.marka);
```

```
SELECT * FROM auto AS tmp WHERE cena_eu > (SELECT AVG(cena_eu) FROM auto WHERE  
kolor LIKE tmp.kolor);
```

```
SELECT * FROM auto AS tmp WHERE przebieg < (SELECT AVG(przebieg) FROM auto  
WHERE kolor LIKE tmp.kolor AND marka LIKE tmp.marka);
```

Poza konkursem – przykład z FROM.

```
SELECT MAX(item), AVG(item2), MIN(item3) FROM (SELECT cena_eu AS item1, cena_pln AS  
item2, przebieg_km AS item3 FROM auto) AS items;
```

Klauzula GROUP BY

Jak nazwa wskazuje grupuje wiersze.

SELECT kolumny_lub_funkcje FROM tabela GROUP BY nazwaKolumny;

Główne zastosowanie to połączenie z funkcjami SUM, AVG, MAX, MIN, COUNT.

Normalnie średnia cena_pln to:

średnia = suma ceny_pln / liczba aut

Grupowanie po marce spowoduje obliczenie osobnych średnich dla każdej marki.

Średnia marki = suma ceny_pln danej marki / liczba aut marki

Konkretne przykłady(ważne):

SELECT MAX(cena_pln), marka FROM auto;

SELECT MAX(cena_pln), marka FROM auto GROUP BY marka;

SELECT COUNT(*), kolor FROM auto GROUP BY kolor;

SELECT AVG(cena_pln), marka, kolor FROM auto GROUP BY marka, kolor;

SELECT AVG(cena_pln), marka FROM auto WHERE przebieg BETWEEN 50000 AND 100000
GROUP BY marka;

SELECT SUM(cena), COUNT(*), dzien FROM zamowienie GROUP BY dzien;

SELECT SUM(cena_pln), COUNT(*), dzien FROM zamowienie GROUP BY YEAR(dzien);

SELECT SUM(kwota) as tmp, waluta FROM pozyczka GROUP BY waluta ORDER BY tmp;

SELECT COUNT(*) as tmp, oprocentowanie FROM pozyczka WHERE oprocentowanie > 4
GROUP BY oprocentowanie;

SELECT SUM(kwota), termin_splaty FROM pozyczka WHERE YEAR(termin_splaty) = 2015
GROUP BY(MONTH(termin_splaty));

Klauzula HAVING

HAVING to odpowiednik WHERE dla klauzuli GROUP BY.

HAVING wykorzystuje się gdy chcemy ograniczyć wyniki na podstawie tego co zwracają funkcje agregujące lub wyświetlić tylko określone grupy wyników.

Np. w BD są samochody 50 marek a my potrzebujemytylko średnią cenę marek o nazwach X, Y, Z.

Przykłady:

SELECT SUM(cena_pln) as suma_cen, marka FROM auto GROUP BY marka HAVING suma_cen
< 40000;

SELECT SUM(cena_pln) as suma_cen, marka FROM auto WHERE kolor LIKE 'czerwony'
GROUP BY marka HAVING suma_cen < 40000;

SELECT SUM(cena_pln) as tmp, marka FROM auto WHERE kolor LIKE 'czerwony' GROUP BY
marka HAVING marka LIKE 'bmw' and tmp > 40000;

SELECT SUM(splacono) as s, waluta FROM pozyczka GROUP BY waluta HAVING s < 3000
ORDER BY s;

```
SELECT MIN(cena_pln) as min, marka FROM auto as a GROUP BY marka HAVING min < 1000;
```

```
HAVING marka LIKE „A”;
```

```
SELECT SUM(cena_eu), marka FROM auto GROUP BY marka HAVING marka LIKE „_____”;
```