

Podzapytania skorelowane, EXISTS i tabele pochodne

Czas: 2 × 45 min

praca indywidualna lub w parach

Cel: Nauczenie się pisania podzapytań skorelowanych, warunków EXISTS / NOT EXISTS i tabel pochodnych w klauzuli FROM oraz oceny, które rozwiązanie jest czytelniejsze i wydajniejsze.

Po zajęciach potrafisz:

- stosuje EXISTS i NOT EXISTS do sprawdzania istnienia powiązanych wierszy
- pisze podzapytanie skorelowane odwołujące się do aliasu tabeli zewnętrznej
- buduje tabelę pochodną w FROM i nadaje jej alias
- wybiera najdroższy element w każdej grupie dwiema metodami
- odczytuje kolumnę select_type w wyniku EXPLAIN i porównuje warianty zapytania

Sprzęt i materiały

- komputer z XAMPP (MariaDB) lub MySQL 8.4
- baza `sklep` zaimportowana z pliku `sklep.sql` (folder Pliki_do_zadan)

Oprogramowanie

- **XAMPP (Apache + MariaDB + phpMyAdmin)** — phpMyAdmin: <http://localhost/phpmyadmin>, zakładka SQL apachefriends.org
- **Konsola mysql** — `mysql -u root sklep` (Shell w panelu XAMPP)
- **Edytor tekstu (VS Code lub Notatnik)** — plik `zad25.sql` code.visualstudio.com

Przygotowanie przed zajęciami

- Miej pod ręką swoje rozwiązania z zadania 24 (Z7, Z8 i Z10) — porównasz je z wersjami z NOT EXISTS.
- Zaimportuj ponownie `sklep.sql`, jeśli dane były zmieniane.

Bezpieczeństwo

- Wykonujesz tylko zapytania SELECT i EXPLAIN — nie zmieniasz danych w bazie `sklep`.
- Nie usuwaj i nie nadpisuj baz innych osób na wspólnym komputerze.

Przebieg zadania

1 EXISTS i NOT EXISTS

EXISTS jest prawdziwe, gdy podzapytanie zwróci choć jeden wiersz; to, co jest na liście SELECT podzapytania, nie ma znaczenia (zwykle pisze się `SELECT 1`). Podzapytanie łączy się z zapytaniem zewnętrznym przez alias.

```
SELECT k.imie, k.nazwisko
FROM klienci AS k
WHERE EXISTS (SELECT 1 FROM zamowienia AS z
              WHERE z.id_klienta = k.id_klienta);
```

1. Z1. Uruchom zapytanie z ramki i policz wiersze. Opisz własnymi słowami, co sprawdza podzapytanie dla każdego klienta.
2. Z2. Wyświetl imię i nazwisko klientów bez żadnego zamówienia, używając NOT EXISTS. Porównaj wynik z Z7 z zadania 24.
3. Z3. Wyświetl nazwy produktów, które nigdy nie zostały zamówione (NOT EXISTS na tabeli `pozycje_zamowienia`).
4. Z4. Wyświetl pracowników, którzy nie są niczymi przełożonymi, używając NOT EXISTS. Porównaj liczbę wierszy z pierwszą (błędą) wersją Z10 z zadania 24 i wyjaśnij, dlaczego NOT EXISTS nie ma problemu z NULL.
5. Z5. Wyświetl klientów, którzy mają co najmniej jedno zamówienie o statusie „nowe” (EXISTS z dodatkowym warunkiem w podzapytaniu).

2 Podzapytania skorelowane

Podzapytanie skorelowane wykonuje się logicznie osobno dla każdego wiersza zapytania zewnętrznego, bo korzysta z jego kolumn.

1. Z6. Wyświetl nazwę, cenę i numer kategorii produktów droższych od średniej ceny w ich własnej kategorii.
2. Z7. Wyświetl najdroższy produkt w każdej kategorii (nazwa kategorii, nazwa produktu, cena) — podzapytanie skorelowane z MAX. Posortuj po nazwie kategorii.
3. Z8. Dla każdego klienta wyświetl imię, nazwisko i liczbę jego zamówień (kolumna `liczba_zamowien`) obliczoną podzapytaniem skorelowanym w liście SELECT. Posortuj malejąco po liczbie zamówień.
4. Z9. Dla każdego produktu wyświetl nazwę i łączną sprzedaną ilość (kolumna `sprzedano`). Produkty, których nie sprzedano, mają mieć 0, a nie NULL (użyj IFNULL lub COALESCE).
5. Z10. Wyświetl imię, nazwisko i pensję pracowników, którzy zarabiają więcej niż średnia pensja osób mających tego samego przełożonego co oni.

3 Tabele pochodne w FROM

Wynik podzapytania w FROM traktuje się jak tymczasową tabelę. Musi mieć alias — sprawdź, co się stanie bez niego.

```
SELECT AVG(w.wartosc)
FROM (SELECT id_zamowienia, SUM(ilosc * cena_jedn) AS wartosc
      FROM pozycje_zamowienia
      GROUP BY id_zamowienia) AS w;
```

1. Z11. Uruchom zapytanie z ramki, a potem to samo bez `AS w`. Zapisz numer i treść błędu. Uzupełnij zapytanie tak, aby zwracało średnią wartość zamówienia zaokrągloną do 2 miejsc (kolumna `srednia_wartosc`).
2. Z12. Wyświetl numer i wartość zamówień, których wartość jest większa od średniej wartości zamówienia. Użyj tabeli pochodnej z wartościami zamówień.
3. Z13. Zbuduj zestawienie: ilu klientów złożyło 0, 1, 2, 3... zamówień (kolumny `liczba_zamowien`, `liczba_klientow`). Wewnętrzne zapytanie: klienci LEFT JOIN zamówienia z GROUP BY; zewnętrzne: grupowanie po liczbie zamówień.
4. Z14. Wyświetl klienta (imię, nazwisko) o największej łącznej wartości zamówień, pomijając zamówienia anulowane. Użyj tabeli pochodnej i porównania z MAX, tak aby przy remisie pokazali się wszyscy.

4 Porównanie wariantów

To samo pytanie można zadać na kilka sposobów. Porównaj czytelność i plan wykonania.

1. Z15. Przepisz Z7 na złączenie produktów z tabelą pochodną (`SELECT id_kategorii, MAX(cena) AS maks FROM produkty GROUP BY id_kategorii`). Sprawdź, czy wynik jest taki sam.
2. Poprzedź Z7 i Z15 słowem EXPLAIN i zapisz w tabeli sprawozdania wartości kolumny `select_type` dla każdego wiersza planu (np. PRIMARY, DEPENDENT SUBQUERY, DERIVED). Wersje serwera mogą przepisać zapytanie inaczej — zapisz to, co pokazuje Twój serwer, i jego wersję (`SELECT VERSION();`).
3. Dla chętnych (Z16): wyświetl drugi najdroższy produkt w każdej kategorii — podzapytanie skorelowane liczące, ile produktów tej samej kategorii jest droższych.
4. Zapisz plik `zad25.sql` i wykonaj zrzuty ekranu wyników Z4, Z7, Z11 (błąd i poprawka), Z13 oraz obu planów EXPLAIN.

W sprawozdaniu umieść

- ☐ tabela zapytań Z1–Z15 (i Z16, jeśli wykonane) z treścią SQL i liczbą wierszy
- ☐ zrzuty ekranu wyników Z4, Z7, Z11 (błąd i poprawka), Z13 i planów EXPLAIN
- ☐ porównanie NOT IN i NOT EXISTS na przykładzie Z4
- ☐ tabela `select_type` dla Z7 i Z15 z wersją serwera oraz wnioski o czytelności i wydajności
- ☐ plik `zad25.sql`
- ☐ odpowiedzi na pytania kontrolne

Pytania kontrolne

1. Czym różni się podzapytanie skorelowane od nieskorelowanego? Po czym je rozpoznać w treści zapytania?
2. Dlaczego NOT EXISTS daje poprawny wynik tam, gdzie NOT IN z kolumną zawierającą NULL zwraca pusty zbiór?
3. Po co tabeli pochodnej alias i jaki błąd pojawia się bez niego?

4. Podaj dwa sposoby znalezienia najdroższego produktu w każdej kategorii. Który wybrałbyś przy bardzo dużej tabeli i dlaczego?
5. Dlaczego w podzapytaniu po EXISTS zwykle pisze się `SELECT 1` ?

Kryteria oceny

Kryterium	Punkty
Z1–Z5: poprawne EXISTS / NOT EXISTS i porównanie z NOT IN	5
Z6–Z10: poprawne podzapytania skorelowane	5
Z11–Z14: tabele pochodne	4
Z15 i EXPLAIN: porównanie wariantów z wnioskami	2
Dokumentacja: plik .sql, zrzuty, tabela zapytań	2
Odpowiedzi na pytania kontrolne	2
Razem	20

Sprawozdanie przygotuj na wzorze dołączonym do zadania (plik .docx o tym samym numerze).