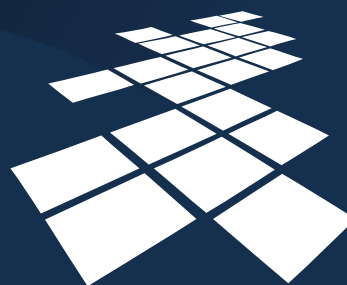


Ćwiczenie 5 – podzapytania

Podzapytania zwykłe i
skorelowane,
podzapytania w
klauzulach FROM i
SELECT



UCZELNIA
ONLINE

Ćwiczenie 5 – podzapytania

Celem ćwiczenia jest zaprezentowanie zagadnień dotyczących stosowania podzapytań w zapytaniach języka SQL. Podzapytania są konstrukcjami, pozwalającymi na wykonywanie zapytań w stylu „podaj nazwisko pracownika, który zarabia najwięcej”, „podaj nazwę zespołu, do którego należy najwięcej pracowników”, itd.

Wymagania:

Konstrukcja prostych zapytań w języku SQL, umiejętność wykorzystania funkcji wierszowych i agregujących.



Plan ćwiczenia

- Charakterystyka ogólnej postaci zapytania z podzapytaniem.
- Zastosowanie podzapytań wierszowych.
- Zastosowanie podzapytań tablicowych.
- Przegląd najczęściej popełnianych błędów przy konstrukcji zapytań z podzapytaniem.
- Charakterystyka operatorów ANY i ALL.
- Podzapytania skorelowane.
- Podzapytania w klauzuli SELECT.
- Podzapytania w klauzuli FROM.

Ćwiczenie 5 – podzapytania (2)

Ćwiczenie rozpoczniemy od charakterystyki zapytań z podzapytaniem i zaprezentowania ogólnej postaci zapytania z zagnieżdżonym podzapytaniem. Dalej przejdziemy do omówienia podzapytań wierszowych i tablicowych. Kolejne slajdy zawierają przegląd najczęściej popełnianych błędów przy konstrukcji zapytań z podzapytaniem. Następne zagadnienie to zastosowanie operatorów ANY i ALL. Dalej omówimy podzapytania skorelowane, podzapytania w klauzuli SELECT i podzapytania w klauzuli FROM.



Podzapytanie (1)

- Ujęte w nawiasy zapytanie, umieszczone wewnątrz innego zapytania (tzw. zapytania zewnętrznego), najczęściej w warunkach klauzul WHERE i HAVING, również w SELECT i FROM (rozwiązania specyficzne).
- Ogólny schemat stosowania podzapytań:

```
SELECT wyrażenie_A1, ...  
FROM nazwa_relacji_A1  
WHERE wyrażenie_A3 operator  
      (SELECT wyrażenie_B1  
        FROM nazwa_relacji_B1  
        WHERE ...)  
ORDER BY wyrażenie_A4;
```

Ćwiczenie 5 – podzapytania (3)

Podzapytanie jest zapytaniem umieszczonym wewnątrz innego zapytania, tzw. zapytania zewnętrznego. Podzapytania najczęściej umieszcza się w warunkach w klauzulach WHERE i HAVING zapytania zewnętrznego, niektóre SZBD dopuszczają również stosowanie podzapytań w klauzulach SELECT i FROM.

Na slajdzie przedstawiono ogólny schemat stosowania podzapytań.

Podzapytanie, zaznaczone czerwonym kolorem, umieszczono w nawiasach po prawej stronie warunku w klauzuli WHERE zapytania zewnętrznego. W dalszej części tego rozdziału przekonamy się, że ten ogólny schemat może podlegać znacznym modyfikacjom.



Podzapytanie (2)

- Przebieg wykonania:
 1. jednokrotne wykonanie podzapytania,
 2. wykonanie zapytania zewnętrznego z wykorzystaniem wartości dostarczonej przez podzapytanie.
- Zastosowany operator zależy od rodzaju podzapytania:
 - podzapytania wierszowe,
 - podzapytania tablicowe.

Do zrozumienia działania podzapytań musimy wyjaśnić przebieg wykonania zapytania z podzapytaniem. Otóż jako pierwsze wykonuje się podzapytanie. Wartości, odczytane przez wyrażenia umieszczone w klauzuli `SELECT` podzapytania, dostarczane są do zapytania zewnętrznego (np. do warunku w klauzuli `WHERE`), tam wykorzystywane są do utworzenia zbioru wynikowego ze zbioru rekordów, przetwarzanych przez zapytanie zewnętrzne. Należy pamiętać, że podzapytanie wykonuje się jednokrotnie.

W przypadku podzapytań w klauzulach `WHERE` i `HAVING`, a więc podzapytań stanowiących część warunku logicznego zapytania zewnętrznego, operator, jaki może zostać zastosowany w warunku, zależy od rodzaju podzapytania. Wyróżniamy tutaj podzapytania wierszowe i tablicowe.



Podzapytanie wierszowe (1)

- Zwraca zawsze co najwyżej jeden rekord, zawierający jedną lub wiele wartości.
- Dopuszczalne zastosowanie operatorów logicznych: =, !=, <>, >, >=, <, <=.
- Przykład: znajdź nazwisko pracownika otrzymującego najniższą płacę podstawową (podzapytanie zwraca rekord z jedną wartością).

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

900

Ćwiczenie 5 – podzapytania (5)

Podzapytanie wierszowe zwraca zawsze co najwyżej jeden rekord, zawierający jedną lub kilka wartości (w zależności od liczby wyrażeń w klauzuli SELECT podzapytania). W przypadku konstruowania zapytań z podzapytaniami wierszowymi dopuszczalne jest stosowanie wszystkich operatorów logicznych, a więc: =, !=, <>, >, >=, < i <=. Przykład pokazuje zapytanie z podzapytaniem wierszowym. Jak już wspomniano, jako pierwsze wykonuje się podzapytanie. Tutaj znajduje ono minimalną płacę podstawową wśród pracowników. Wynikiem podzapytania będzie zawsze jeden rekord z jedną wartością, stąd dopuszczalne jest zastosowanie operatora =. Odczytana przez podzapytanie wartość (w naszym przypadku 900), zostaje przekazana do zapytania zewnętrznego, gdzie jest użyta w warunku w klauzuli WHERE (placa_pod = 900). W wyniku otrzymujemy nazwiska pracowników, zarabiających minimalną płacę podstawową.



Podzapytanie wierszowe (2)

- Przykład: podaj nazwisko profesora, otrzymującego najniższą płacę podstawową wśród profesorów (podzapytanie zwraca rekord z wieloma wartościami).

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

3070, PROFESOR

- Wartość atrybutu `placa_pod` porównywana jest z wartością wyrażenia `min(placa_pod)`, `etat` z ciągiem znaków `PROFESOR`.

Ćwiczenie 5 – podzapytania (6)

Kolejny przykład pokazuje zapytanie z podzapytaniem wierszowym, którego rekord składa się z dwóch wartości. Podzapytanie wylicza minimalną płacę podstawową pracowników na etacie `PROFESOR`. W klauzuli `SELECT` podzapytania dodano statyczny ciąg znaków `PROFESOR`, tak więc wynikiem podzapytania będzie jeden rekord z dwiema wartościami: minimalną pensją profesora (3070) i ciągiem znaków „PROFESOR”. Jeśli podzapytanie zwraca rekord z więcej niż jedną wartością, konieczne jest zastosowanie specjalnej konstrukcji w warunku zapytania zewnętrznego, w którym umieszczono podzapytanie. Otóż w nawiasach okrągłych umieszcza się listę atrybutów, które zostaną użyte do porównania z wartościami podzapytania. W przykładzie na liście znajdują się dwa atrybuty: `PLACA_POD` i `ETAT`, atrybut `PLACA_POD` będzie porównany z wynikiem wyrażenia `min(placa_pod)` z podzapytania, a atrybut `ETAT` ze statycznym ciągiem znaków „PROFESOR”, z podzapytania. Jeśli oba porównania zakończą się sukcesem, wówczas cały warunek jest prawdziwy i rekord zapytania zewnętrznego trafi do zbioru wynikowego.



Podzapytanie tablicowe

- Zwraca zbiór rekordów zawierających jedną lub wiele wartości.
- Dopuszczalne zastosowanie operatorów: IN, ANY, ALL.
- Przykład: 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);
```

ADIUNKT	2845,5
ASYSTENT	1971
DOKTORANT	900
DYREKTOR	4730
PROFESOR	3960
SEKRETARKA	1590

Ćwiczenie 5 – podzapytania (7)

Drugi rodzaj podzapytań, tzw. podzapytania tablicowe (nazywane również podzapytaniami wielowierszowymi), zwraca zbiór rekordów, zawierających jedną lub kilka wartości. W przypadku stosowania podzapytań tablicowych w warunkach w klauzulach WHERE i HAVING zapytań zewnętrznych można użyć jedynie operatora IN zawierania w zbiorze oraz dwóch nowych operatorów, ANY i ALL (operatory te zostaną przedstawione na następnych slajdach).

W zaprezentowanym na slajdzie przykładzie podzapytanie znajduje maksymalną płacę podstawową pracownika dla każdej grupy etatowej. Podzapytanie odczytuje zbiór rekordów, z których każdy posiada dwie wartości: nazwę etatu i maksymalną pensję dla etatu. Podzapytanie umieszczono w klauzuli WHERE zapytania zewnętrznego, stosując operator IN. Ponieważ podzapytanie zwraca w każdym rekordzie dwie wartości, po lewej stronie warunku klauzuli WHERE umieszczono listę atrybutów: atrybut ETAT z rekordu przeglądane przez zapytanie zewnętrzne będzie porównywana z wartością atrybutu ETAT z podzapytania, a atrybut PLACA_POD z wartością wyrażenia MAX(PLACA_POD) podzapytania. Jeśli dopasowanie znajdzie dla obu atrybutów (np. dany pracownik jest na etacie „ADIUNKT” i zarabia 2845,5), wówczas rekord zapytania zewnętrznego trafi do zbioru wynikowego. Przykładowe zapytanie znajduje nazwiska pracowników, zarabiających maksymalne płace w ramach swoich grup etatowych.



Operatory ANY i ALL (1)

- Stosowane razem z operatorami logicznymi w zapytaniach z podzapytaniem tablicowym.
- Operator ANY – warunek prawdziwy jeśli jest spełniony dla przynajmniej jednej wartości, odczytanej przez podzapytanie.
- Operator ALL – warunek prawdziwy jeśli jest spełniony dla wszystkich wartości, odczytanych przez podzapytanie.

Omówimy teraz dwa nowe operatory, stosowane w zapytaniach z podzapytaniem tablicowym: ANY i ALL. Operatory te stosuje się w połączeniu z operatorami logicznymi. Warunek, skonstruowany z operatorem ANY jest prawdziwy, jeśli jest spełniony dla chociaż jednej wartości, zwracanej przez podzapytanie do zapytania zewnętrznego. Z kolei warunek z operatorem ALL jest prawdziwy wtedy, gdy spełniony jest dla wszystkich wartości, odczytywanych przez podzapytanie. Tak więc operator ALL tworzy bardziej restrykcyjne warunki niż operator ANY.



Operatory ANY i ALL (2)

- Podaj nazwiska pracowników, których płaca podstawowa jest większa od płacy podstawowej dowolnego pracownika zespołu 30.

```
SELECT nazwisko FROM pracownicy WHERE placa_pod > ANY  
(SELECT placa_pod FROM pracownicy WHERE id_zesp = 30);
```

- Podaj nazwiska pracowników, których płaca podstawowa jest większa od płac podstawowych wszystkich pracowników zespołu 30.

```
SELECT nazwisko FROM pracownicy WHERE placa_pod > ALL  
(SELECT placa_pod FROM pracownicy WHERE id_zesp = 30);
```

Ćwiczenie 5 – podzapytania (9)

Pierwszy przykład pokazuje zastosowanie operatora ANY z operatorem logicznym >. Podzapytanie zwraca płace podstawowe pracowników z zespołu o numerze 30. Zapytanie zewnętrzne przegląda rekordy z relacji PRACOWNICY, sprawdzając dla każdego z rekordów warunek: płaca pracownika ma być większa od przynajmniej jednej płacy, odczytanej przez podzapytanie (czyli płacy pracowników z zespołu o numerze 30). Jeśli warunek jest spełniony, nazwisko pracownika trafia do zbioru wynikowego.

W drugim przykładzie wykonujemy to samo zapytanie, zastępując operator ANY operatorem ALL. Teraz rekord, przeglądany przez zapytanie zewnętrzne, trafi do zbioru wynikowego, jeśli płaca podstawowa pracownika będzie większa od płac podstawowych wszystkich pracowników z zespołu o numerze 30.



Podzapytania w klauzuli HAVING

- Zasady konstrukcji – te same co dla podzapytań w klauzuli WHERE.
- Podaj nazwy i średnie płace podstawowe w zespołach, w których średnia płaca przekracza średnią płacę wśród wszystkich pracowników.

```
SELECT nazwa, AVG(placa_pod) AS srednia
FROM pracownicy natural join zespoły
GROUP BY nazwa
HAVING AVG(placa_pod) >
      (SELECT AVG(placa_pod)
       FROM pracownicy);
```

Ćwiczenie 5 – podzapytania (10)

Przykłady, prezentowane na poprzednich slajdach, pokazywały podzapytania umieszczane jedynie w klauzuli WHERE. Umieszczenie podzapytania w klauzuli HAVING rządzi się tymi samymi zasadami, jakie zostały omówione dla podzapytań w klauzuli WHERE zapytania.

W zaprezentowanym przykładzie podzapytanie znajduje średnią płacę podstawową wśród pracowników. Zapytanie zewnętrzne dokonuje grupowania ze względu na wartość atrybutu NAZWA zbioru rekordów, powstałego z połączenia pracowników z zespołami, następnie w każdej z grup wylicza średnią płacę podstawową. Do zbioru wynikowego trafiają tylko te grupy, w których wartość średniej płacy podstawowej jest większa od wartości wyliczonej przez podzapytanie.



Reguły zagnieżdżania podzapytań (1)

- Podzapytanie może być umieszczone w dowolnym miejscu klauzuli WHERE i HAVING, może stanowić również część wyrażenia.

```
SELECT nazwisko FROM pracownicy  
WHERE 1.5*(SELECT AVG(placa_pod) FROM pracownicy  
          WHERE etat = 'ASYSTENT') < placa_pod;
```

- Brak klauzuli ORDER BY w podzapytaniu (dopuszczalne w SZBD Oracle).
- Kolejność wykonywania: od najbardziej zagnieżdżonego do najbardziej zewnętrznego.

Omówimy teraz reguły zagnieżdżania podzapytań. Podzapytanie możemy umieścić w dowolnym miejscu warunku w klauzuli WHERE i HAVING, zarówno po lewej jak i prawej stronie warunku. Jeśli w warunku mamy wyrażenie, podzapytanie może stanowić część wyrażenia. W zaprezentowanym na bieżącym slajdzie przykładzie podzapytanie umieszczono w wyrażeniu po lewej stronie warunku (wartość wyznaczona przez podzapytanie zostaje pomnożona przez 1,5 i porównana z wartością płacy podstawowej).

W podzapytaniu nie należy umieszczać klauzuli ORDER BY, powinna się ona pojawić jako ostatnia klauzula zapytania zewnętrznego. Niektóre SZBD (np. Oracle) dopuszczają jednak stosowanie ORDER BY wewnątrz podzapytania.

Wewnątrz podzapytania, umieszczonego w zapytaniu zewnętrznym, można umieścić kolejne podzapytanie, w tym podzapytaniu kolejne, itd., tworząc wielopoziomą strukturę zagnieżdżeń. W takim przypadku wykonanie zapytania rozpoczyna się od podzapytania najgłębiej zagnieżdżonego w kierunku malejącego zagnieżdżenia.



Reguły zagnieżdżania podzapytań (2)

- Podaj nazwy i średnie płace podstawowe w zespołach, w których średnia płaca przekracza średnią płacę w zespole o nazwie ALGORYTMY.

3.

```
SELECT nazwa, AVG(placa_pod) AS srednia
FROM pracownicy natural join zespoly
GROUP BY nazwa
HAVING AVG(placa_pod) >
```

3350

2.

```
(SELECT AVG(placa_pod)
FROM pracownicy
WHERE id_zesp =
```

40

1.

```
(SELECT id_zesp
FROM zespoly
WHERE nazwa = 'ALGORYTMY'));
```

Ćwiczenie 5 – podzapytania (12)

Bieżący slajd przedstawia przykład zapytania z dwoma zagnieżdżonymi podzapytaniem. Jako pierwsze wykonuje się zapytanie, odczytujące numer zespołu o nazwie „ALGORYTMY”. Zapytanie na wyższym poziomie wykorzystuje ten numer do wyliczenia średniej płacy pracowników, należących do zespołu ALGORYTMY. Wreszcie zapytanie zewnętrzne znajduje nazwy i średnie płace w zespołach, w których średnie płace są większe od średniej płacy w zespole ALGORYTMY.



Najczęściej popełniane błędy

- Zastosowanie operatora logicznego dla podzapytania tablicowego.

```
SELECT nazwa, adres FROM zespoły
WHERE id_zesp =
      (SELECT id_zesp FROM pracownicy WHERE nazwisko IN
       ('Nowak','Kowalski'));
```

- Brak dopasowania liczby atrybutów w warunku zapytania zewnętrznego i klauzuli SELECT podzapytania.

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

Ćwiczenie 5 – podzapytania (13)

Bieżący slajd zwraca uwagę na najczęściej popełniane błędy przy konstruowaniu zapytań z podzapytaniami.

Pierwszy błąd wynika z zastosowania złego operatora. W zaprezentowanym przykładzie podzapytanie jest podzapytaniem tablicowym, tymczasem w warunku zapytania zewnętrznego użyto operatora logicznego, który, jak pamiętamy, może być stosowany jedynie dla podzapytań wierszowych. Wykonanie tego zapytania zakończy się komunikatem o błędzie.

Kolejny błąd to niedopasowanie liczby atrybutów warunku zapytania zewnętrznego do liczby wartości w rekordzie, zwracanym przez podzapytanie. W przykładzie podzapytanie tablicowe zwraca po dwie wartości w każdym rekordzie (etat i maksymalną płacę dla etatu), tymczasem warunek w zapytaniu zewnętrznym skonstruowany został z tylko jednym atrybutem (PLACA_POD). Wykonanie zapytania spowoduje błąd.



Zadania

1. Wyświetl nazwiska i etaty pracowników pracujących w tym samym zespole co pracownik o nazwisku Nowak (załóż, że w zbiorze pracowników istnieje tylko jeden Nowak).
2. Wyświetl wszystkie dane o najdłużej zatrudnionym profesorze.
3. Wyświetl najkrócej pracujących pracowników każdego zespołu. Uszereguj wyniki zgodnie z kolejnością zatrudnienia.
4. Wyświetl dane zespołów, które nie zatrudniają żadnych pracowników.

Ćwiczenie 5 – podzapytania (14)

Bieżący slajd rozpoczyna zbiór zadań, których celem jest utrwalenie wiadomości o konstrukcji zapytań wykorzystujących podzapytania wierszowe i tablicowe.



Zadania

5. Wyświetl nazwiska tych profesorów, którzy wśród swoich podwładnych nie mają żadnych stażystów.
6. Wyświetl numer zespołu wypłacającego miesięcznie swoim pracownikom najwięcej pieniędzy.
7. Podaj nazwę zespołu zatrudniającego najwięcej pracowników.



```
① SELECT nazwisko, etat FROM pracownicy  
WHERE id_zesp = (SELECT id_zesp FROM pracownicy  
WHERE nazwisko = 'Nowak')
```

```
② SELECT * FROM pracownicy  
WHERE etat = 'PROFESOR' and zatrudniony =  
(SELECT MIN(zatrudniony) FROM pracownicy  
WHERE etat = 'PROFESOR');
```

```
③ SELECT nazwisko, zatrudniony, id_zesp  
FROM pracownicy  
WHERE (id_zesp, zatrudniony) IN  
(SELECT id_zesp, MAX(zatrudniony) FROM pracownicy  
GROUP BY id_zesp) ORDER BY zatrudniony;
```

Bieżący slajd przedstawia rozwiązania zadań (1), (2) i (3), których treść zacytowano poniżej.

- (1) Wyświetl nazwiska i etaty pracowników pracujących w tym samym zespole co pracownik o nazwisku Nowak (załóż, że w zbiorze pracowników istnieje tylko jeden Nowak).
- (2) Wyświetl wszystkie dane o najdłużej zatrudnionym profesorze.
- (3) Wyświetl najkrócej pracujących pracowników każdego zespołu. Uszereguj wyniki zgodnie z kolejnością zatrudnienia.



- 4 **SELECT * FROM zespoly WHERE id_zesp not IN
(SELECT id_zesp FROM pracownicy WHERE id_zesp is not null);**
- 5 **SELECT nazwisko FROM pracownicy
WHERE etat = 'PROFESOR' AND id_prac not IN
(SELECT id_szefa FROM pracownicy WHERE etat = 'STAŻYSTA');**
- 6 **SELECT id_zesp, SUM(placa_pod) as suma_plac FROM pracownicy
GROUP BY id_zesp HAVING SUM(placa_pod) =
(SELECT MAX(SUM(placa_pod)) FROM pracownicy
GROUP BY id_zesp);**
- 7 **SELECT nazwa FROM zespoly NATURAL JOIN pracownicy
GROUP BY id_zesp, nazwa HAVING COUNT (*) =
(SELECT MAX(COUNT(*)) FROM pracownicy GROUP BY id_zesp);**

Bieżący slajd przedstawia rozwiązania zadań (4), (5), (6) i (7), których treść zacytowano poniżej.

- (4) Wyświetl dane zespołów, które nie zatrudniają żadnych pracowników.
- (5) Wyświetl nazwiska tych profesorów, którzy wśród swoich podwładnych nie mają żadnych stażystów.
- (6) Wyświetl numer zespołu wypłacającego miesięcznie swoim pracownikom najwięcej pieniędzy.
- (7) Podaj nazwę zespołu zatrudniającego najwięcej pracowników.



Podzapytania skorelowane (1)

- Podzapytanie wykonywane wielokrotnie – raz dla każdego rekordu przeładowanego przez zapytanie zewnętrzne.
- W podzapytaniu odwołanie do wyrażenia z zapytania zewnętrznego.
- Ogólny schemat:

```

SELECT wyrażenie_A1, ...
FROM nazwa_relacji_A1
WHERE wyrażenie_A2 operator
      (SELECT wyrażenie_B1
       FROM nazwa_relacji_B1
       WHERE wyrażenie_B2 = wyrażenie_A3)
ORDER BY wyrażenie_A4;
  
```

korelacja

Ćwiczenie 5 – podzapytania (18)

Rozpoczniemy teraz omawianie odmiennie wykonywanej grupy podzapytań, tzw. podzapytań skorelowanych. Przypomnijmy – poprzednie podzapytania, nazwijmy je terminem „zwykłe”, były wykonywane tylko jeden raz. Tymczasem podzapytania skorelowane wykonują się wielokrotnie – tyle razy, ile rekordów przeładowa zapytanie zewnętrzne. Wyjaśnienia wymaga termin „przeładowa”. Zapytanie zewnętrzne przeładowa rekordy, pobierając je z relacji z bazy danych. Jeśli dla przeładowanego rekordu wszystkie warunki, zdefiniowane w zapytaniu, są spełnione, taki rekord trafia do zbioru wynikowego. Czyli liczba rekordów przeładowanych przez zapytanie nie zawsze jest równa liczbie rekordów w zbiorze wynikowym.

Cechą charakterystyczną podzapytań skorelowanych jest odwołanie wewnątrz podzapytania, najczęściej w warunku, do atrybutu z relacji, którą przeładowa zapytanie zewnętrzne. To odwołanie nosi nazwę korelacji.

Bieżący slajd przedstawia ogólny schemat konstrukcji zapytań z podzapytaniem skorelowanym. Widzimy, że schemat nie różni się zbyt wiele od schematów zapytań ze zwykłymi podzapytaniem, różnicą jest obecność korelacji w podzapytaniu – użycie wyrażenia_A3 w warunku podzapytania. Wyrażenie_A3 pochodzi z rekordu przeładowanego przez zapytanie zewnętrzne.



Podzapytania skorelowane (2)

- Przebieg wykonania:
 1. Pobranie rekordu R1 przez zapytanie zewnętrzne.
 2. Wykonania podzapytania na podstawie wartości rekordu R1, podzapytanie dostarcza wartości dla warunku W1 zapytania zewnętrznego.
 3. Jeśli W1 jest spełniony, R1 trafia do zbioru wynikowego.
 4. Jeśli pozostały jeszcze nie pobrane rekordy, przejdź do punktu 1.

Wykonanie zapytania z podzapytaniem skorelowanym znacznie różni się od wykonania zapytania z podzapytaniem zwykłym. W pierwszym kroku zapytanie zewnętrzne pobiera rekord R1 z relacji. Dla tego rekordu wykonywane jest podzapytanie, dostarczające wartość, która użyta zostaje w warunku sprawdzanym przez zapytanie zewnętrzne dla rekordu R1. Jeśli warunek jest spełniony, rekord R1 trafia do zbioru wynikowego, w przeciwnym razie rekord zostaje odrzucony. Następnie zapytanie zewnętrzne pobiera kolejny rekord, wykonuje dla niego podzapytanie, itd. aż do wyczerpania rekordów w relacji.



Podzapytanie skorelowane (3)

- 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);
```

pracownicy p

Janicki	PROFESOR	2500
Nowicki	PROFESOR	3000
Opolski	ASYSTENT	1000
Kotarski	ASYSTENT	1500

pracownicy

Janicki	PROFESOR	2500
Nowicki	PROFESOR	3000
Opolski	ASYSTENT	1000
Kotarski	ASYSTENT	1500

Ćwiczenie 5 – podzapytania (20)

Omówimy teraz wykonanie zapytania z podzapytaniem skorelowanym na przykładzie. Chcemy odczytać nazwiska pracowników, których płace podstawowe są większe niż średnie płace pracowników w zespołach, do których należą. Ponieważ zarówno w zapytaniu zewnętrznym, jak i w podzapytaniu operujemy na tej samej relacji PRACOWNICY, konieczne jest zdefiniowanie aliasu „p” dla relacji PRACOWNICY, przeglądanej w zapytaniu zewnętrznym. Korelacja jest realizowana przez użycie atrybutu ETAT z relacji PRACOWNICY zapytania zewnętrznego w warunku w podzapytaniu (ETAT = P.ETAT). Załóżmy, że w relacji PRACOWNICY mamy tylko cztery rekordy. Zapytanie zewnętrzne pobiera pierwszy rekord z relacji PRACOWNICY, rekord opisujący pracownika o nazwisku Janicki, zatrudnionego na etacie PROFESOR i zarabiającego 2500. Dla tego rekordu wykonywane jest podzapytanie, wyliczające średnią płacę wśród wszystkich profesorów, a więc średnią płacę pracowników o nazwiskach Janicki i Nowicki. Wynik podzapytania, 2750, jest porównywany z pensją Janickiego. Warunek nie jest spełniony, Janicki zostaje odrzucony. Zapytanie zewnętrzne pobiera kolejny rekord z relacji PRACOWNICY, opisujący profesora o nazwisku Nowicki, zarabiającego 3000. Ponownie wykonywane jest podzapytanie, które znowu liczy średnią pensję wśród profesorów. Wynik oczywiście jest ten sam, co poprzednio (2750), ale tym razem warunek dla pracownika Nowickiego jest spełniony i rekord opisujący Nowickiego trafia do zbioru wynikowego. Analogicznie przebiega wykonanie zapytania dla dwóch następujących rekordów.

Reasumując: zapytanie zewnętrzne przejrzęło cztery rekordy, podzapytanie zostało wykonane czterokrotnie, do zbioru wynikowego trafiły dwa rekordy.



Operator EXISTS

- Przyjmuje wartość prawdy gdy podzapytanie zwróci przynajmniej jeden rekord.
- Przykład: podaj nazwiska pracowników, którzy posiadają podwładnych.

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

- Podzapytanie może zwrócić dowolną wartość, również literał.

```
SELECT nazwisko FROM pracownicy p WHERE EXISTS  
  (SELECT 1 FROM pracownicy WHERE id_szefa = p.id_prac);
```

Nowym operatorem, używanym jedynie w zapytaniach z podzapytaniem skorelowanym, jest operator EXISTS. Operator ten zwraca wartość prawdy wtedy, gdy następujące zaraz za nim podzapytanie zwróci przynajmniej jeden rekord. Nie są przy tym ważne wartości, jakie ten rekord zawiera. W pierwszym zaprezentowanym na slajdzie przykładzie zapytanie zewnętrzne przegląda rekordy z relacji PRACOWNICY. Dla każdego rekordu wykonywane jest podzapytanie skorelowane, w którym poszukiwane są dane o wszystkich pracownikach, którzy w atrybucie ID_SZEFA mają wartość równą wartości atrybutu ID_PRAC z zapytania zewnętrznego (czyli podzapytanie szuka po prostu podwładnych pracownika, aktualnie przeglądane przez zapytanie zewnętrzne). Jeśli podzapytanie zwróci rekord, operator EXISTS przyjmie wartość prawdy i bieżący rekord zapytania zewnętrznego trafi do zbioru wynikowego. Będzie to rekord opisujący pracownika, który posiada przynajmniej jednego podwładnego.

To samo zapytanie zaprezentowano w drugim przykładzie, zastępując w podzapytaniu symbol * (odczyt wszystkich atrybutów) literałem 1. Nie zmienia to w żaden sposób wyniku zapytania.



Zadania

8. Wyświetl nazwiska i imiona pracowników zarabiających więcej niż 50% maksymalnej pensji w zespołach, do których należą.
9. Stosując podzapytanie skorelowane wyświetl informacje o zespole nie zatrudniającym żadnych pracowników.
10. Wyświetl nazwiska i pensje trzech najlepiej zarabiających pracowników.
11. Wyświetl informacje o pracownikach, którzy mają błędne przypisania do zespołów (wartość id_zesp nie wskazuje na żaden istniejący zespół). Pamiętaj, aby pominąć pracowników bez przydziałów do zespołów!

Ćwiczenie 5 – podzapytania (22)

Bieżący slajd rozpoczyna zbiór zadań, których celem jest utrwalenie wiadomości o konstrukcji zapytań wykorzystujących podzapytania skorelowane.



```
8 SELECT nazwisko, imie FROM pracownicy p
   WHERE placa_pod > 0.5 * (SELECT MAX(placa_pod)
                             FROM pracownicy WHERE id_zesp = p.id_zesp)
```

```
9 SELECT * FROM zespoly z WHERE NOT EXISTS
   (SELECT 1 FROM pracownicy WHERE id_zesp = z.id_zesp);
```

```
10 SELECT nazwisko, placa_pod FROM pracownicy p
     WHERE 3 > (SELECT COUNT(*) FROM pracownicy
               WHERE placa_pod > p.placa_pod)
     ORDER BY 2 desc;
```

```
11 SELECT * FROM pracownicy p WHERE id_zesp is not null
     and NOT EXISTS (SELECT 1 FROM zespoly
                    WHERE id_zesp = p.id_zesp);
```

Bieżący slajd przedstawia rozwiązania zadań (8), (9), (10) i (11), których treść zacytowano poniżej.

- (8) Wyświetl nazwiska i imiona pracowników zarabiających więcej niż 50% maksymalnej pensji w zespołach, do których należą.
- (9) Stosując podzapytanie skorelowane wyświetl informacje o zespole nie zatrudniającym żadnych pracowników.
- (10) Wyświetl nazwiska i pensje trzech najlepiej zarabiających pracowników.
- (11) Wyświetl informacje o pracownikach, którzy mają błędne przypisania do zespołów (wartość ID_ZESP nie wskazuje na żaden istniejący zespół). Pamiętaj, aby pominąć pracowników bez przydziałów do zespołów!



Podzapytania w klauzuli SELECT

- Podzapytanie musi zwrócić dokładnie jedną wartość dla każdego rekordu 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;
```

Ćwiczenie 5 – podzapytania (24)

Rozwiązaniem specyficznym dla niektórych SZBD jest możliwość umieszczenia podzapytania w klauzuli SELECT. Takie podzapytanie jest wówczas źródłem danych dla nowego atrybutu w wyniku zapytania zewnętrznego. Podzapytanie, umieszczone w klauzuli SELECT, musi zwracać co najwyżej jedną wartość dla rekordu, przeglądanego przez zapytanie zewnętrzne. Należy również zdefiniować alias dla atrybutu, definiowanego przez podzapytanie.

Przykładowe zapytanie zewnętrzne przegląda rekordy relacji ZESPOLY. Dla każdego rekordu wykonywane jest umieszczone w klauzuli SELECT zapytanie zewnętrzne podzapytanie skorelowane, wyliczające średnią płacę podstawową pracowników przeglądanego zespołu.



Podzapytania w klauzuli FROM

- Podzapytanie tworzy zbiór danych dla zapytania zewnętrznego.
- Przykład: dla każdego zespołu podaj jego nazwę i średnią płacę podstawową pracowników w zespole.

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

Kolejnym rozwiązaniem specyficznym dla niektórych SZBD jest możliwość umieszczenia podzapytań w klauzuli FROM zapytania zewnętrznego. Zbiór danych, odczytywanych przez podzapytanie, jest wówczas zbiorem wejściowym dla zapytania zewnętrznego. Nie ma żadnych ograniczeń co do konstrukcji podzapytań w klauzuli FROM.

Zaprezentowany przykład wykonuje połączenie naturalne rekordów z relacji ZESPOLY z rekordami, odczytanymi przez podzapytanie w klauzuli FROM zapytania zewnętrznego, które to wylicza średnie płaca podstawowe pracowników poszczególnych zespołów. Widzimy, że w klauzuli SELECT zapytania zewnętrznego umieszczono atrybut SREDNIA_PLACA, wyliczany przez podzapytanie.



Zadania

12. Dla każdego pracownika (również bez przydziału do zespołu) wyświetl jego nazwisko, nazwę zespołu, do którego należy i średnią pensję w zespole.
13. Zmodyfikuj zapytanie z p.12, aby móc wyświetlić dodatkowo widełki płacowe (atrybuty placa_od i placa_do) z relacji ETATY. Użyj podzapytania w klauzuli FROM.

Bieżący slajd rozpoczyna zbiór zadań, których celem jest utrwalenie wiadomości o konstrukcji zapytań wykorzystujących podzapytania w klauzulach SELECT i FROM .



```
12 SELECT nazwisko, nazwa,  
      (SELECT AVG(placa_pod) FROM pracownicy  
       WHERE id_zesp = p.id_zesp) as srednia  
FROM pracownicy p LEFT JOIN zespoły z ON p. id_zesp = z.id_zesp;
```

```
13 SELECT nazwisko, nazwa,  
      (SELECT AVG(placa_pod) FROM pracownicy  
       WHERE id_zesp = p.id_zesp) as srednia,  
      placa_od, placa_do  
FROM pracownicy p LEFT JOIN zespoły z ON p. id_zesp = z.id_zesp  
JOIN (SELECT nazwa, placa_od, placa_do FROM etaty) e  
ON p.etat = e.nazwa;
```

Bieżący slajd przedstawia rozwiązania zadań (12) i (13), których treść zacytowano poniżej.

- (12) Dla każdego pracownika (również bez przydziału do zespołu) wyświetl jego nazwisko, nazwę zespołu, do którego należy i średnią pensję w zespole.
- (13) Zmodyfikuj zapytanie z p.12, aby móc wyświetlić dodatkowo widełki płacowe (atrybuty `placa_od` i `placa_do`) z relacji `ETATY`. Użyj podzapytania w klauzuli `FROM`.



Podsumowanie

- Podzapytanie jest zapytaniem zagnieżdżonym w innym zapytaniu.
- Podzapytanie może zostać zagnieżdżone w klauzulach WHERE, HAVING, SELECT i FROM.
- W zależności od sposobu wykonania podzapytania dzielimy na podzapytania zwykłe i podzapytania skorelowane.

Ćwiczenie 5 – podzapytania (28)

W zakończonym ćwiczeniu zostało zaprezentowane użycie podzapytań w zapytaniach języka SQL. Podzapytanie jest zapytaniem, umieszczonym w klauzulach: WHERE, HAVING, SELECT lub FROM innego zapytania, tzw. zapytania zewnętrznego. W zależności od sposobu wykonania podzapytania dzielimy na podzapytania zwykłe, wykonywane jednokrotnie, oraz podzapytania skorelowane, wykonywane wielokrotnie, po jednym razie dla każdego rekordu przeglądane przez zapytanie główne.

Każde z omówionych zagadnień zostało utrwalone przez serię zadań.