



**Wydział Elektroniki
i Technik Informatycznych**

POLITECHNIKA WARSZAWSKA

Bazy Danych 1

edycja 21L

Laboratorium 6

Wprowadzenie i przebieg laboratorium

 Podzapytania (zapytania zagnieżdżone) - wprowadzenie

 Podzapytania **nieskorelowane/skorelowane**

→ Omówienie podzapytań zwracających pojedynczy wiersz danych

→ Omówienie podzapytań zwracających wiele wierszy danych

 Podzapytania w klauzulach **WHERE, SELECT, FROM, HAVING**

Podzapytania - wprowadzenie

- Podzapytanie jest poleceniem **SELECT**, które zagnieżdżone jest w ramach klauzul innego polecenia **SELECT** (tzw. zapytania zewnętrznego).
- Podzapytanie na ogół wykonywane jest przed zapytaniem zewnętrznym.
- Składnia dla podzapytania umieszczonego w klauzuli **WHERE**:

```
SELECT column_list  
FROM table  
WHERE [expression] operator  
( SELECT column_list  
  FROM table );
```

expression nie występuje w przypadku operatora EXISTS

Konieczne jest dobranie operatora do rodzaju danych zwracanych przez podzapytanie.

Podzapytanie umieszczane jest w nawiasach.

Podzapytania - wprowadzenie (cd.)

- Możliwe jest zastosowanie różnych tabel w ramach podzapytania i zapytania zewnętrznego, ale również tych samych tabel.
- W ramach podzapytań można zagnieżdżać inne podzapytania.
- Składnia języka nie zakłada limitu na liczbę podzapytań w zapytaniu i ich liczbę zagnieżdżeń.
- Podzapytania mogą być umieszczane w różnych klauzulach zapytania zewnętrznego: WHERE, FROM, HAVING lub w klauzuli SELECT.



Podzapytania nieskorelowane

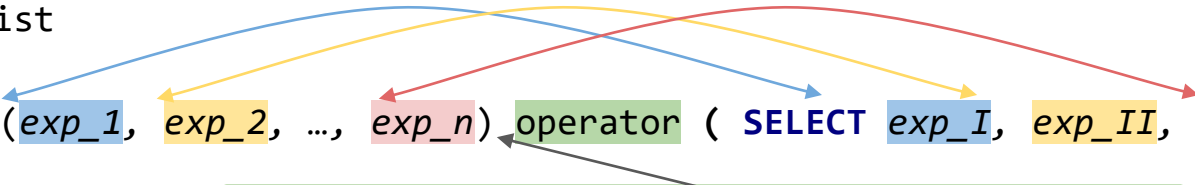
Podzapytania nieskorelowane

- Nie posiadają warunku złączenia pomiędzy zapytaniem zewnętrznym a podzapytaniem umieszczonym w klauzuli **WHERE**.
- Podzapytania nieskorelowane mogą zwracać **zero wierszy danych**, **jeden wiersz danych**, jak i **wiele wierszy danych** (oczywiście trzeba odpowiednio dobrać operator).

Podzapytanie nieskorelowane zwracające jeden wiersz danych

- W przypadku podzapytania nieskorelowanego zwracającego **jeden wiersz danych** możemy wykorzystać operatory arytmetyczne =, !=, >, <, >=, <=.
- Jeśli podzapytanie zwraca dokładnie jedną wartość, to może zostać ona dopasowana tylko do pojedynczego wyrażenia.
- Jeśli podzapytanie zwraca listę wartości (w postaci jednego wiersza), to muszą one zostać dopasowane do listy wyrażeń poprzedzających podzapytanie (tzn. muszą odpowiadać sobie typy danych wyrażeń).

```
SELECT column_list
FROM table
WHERE (exp_1, exp_2, ..., exp_n) operator ( SELECT exp_I, exp_II, ...,
exp_N
FROM table );
```



Jeśli podzapytanie zwraca dokładnie jeden wiersz danych to operatorem może być =, !=, >, <, >=, <=

Podzapytanie nieskorelowane w klauzuli WHERE - podzapytanie zwraca pojedynczą wartość

- Przykład z podzapytaniem zwracającym **dokładnie jedną wartość**:
skonstruuj zapytanie, które zwróci dane pracownika (pracowników) o największych zarobkach.

```
SELECT name, surname  
FROM Employees  
WHERE salary =  
(  
    SELECT MAX(salary)  
    FROM Employees  
);
```

Podzapytanie zwraca jedną wartość -
możliwe jest użycie operatora =, <>, >,
<, >=, <=

Podzapytanie nieskorelowane w klauzuli WHERE - podzapytanie zwraca jeden wiersz danych

→ Przykład z podzapytaniem zwracającym **jeden wiersz danych**: *skonstruuuj zapytanie, które zwróci dane pracowniczki (pracowniczek) o najmniejszych zarobkach.*

```
SELECT name, surname, gender
FROM employees
WHERE (salary, gender) =
(
    SELECT MIN(salary), 'K'
    FROM Employees
    WHERE Gender = 'K'
);
```

Podzapytanie zwraca jeden wiersz danych - możliwe jest użycie operatorów =, <>, >, <, >=, <=

Ćwiczenie - podzapytanie zwraca jeden wiersz danych

1. Napisz zapytanie, które wyświetli imię, nazwisko oraz nazwy zakładów, w których pracownicy mają większe zarobki niż minimalne zarobki na stanowisku o nazwie 'Konsultant'.
2. Napisz zapytanie, które zwróci dane najmłodszego wśród dzieci pracowników.
(Skorzystaj z podzapytań. Jaki jest inny sposób na osiągnięcie tego wyniku?)
3. Napisz zapytanie, które zwróci dane dzieci najstarszego pracownika z zakładu 102.
4. Napisz zapytanie, które wyświetli wszystkich pracowników, którzy zostali zatrudnieni nie wcześniej niż najwcześniej zatrudniony pracownik w zakładzie o id 101 i nie później niż najpóźniej zatrudniony pracownik w zakładzie o id 107.
5. Wyświetl średnie zarobki dla każdego ze stanowisk, o ile średnie te są większe od średnich zarobków w departamencie "Administracja".

Podzapytanie nieskorelowane w klauzuli WHERE (podzapytanie zwraca wiele wierszy)

- Podzapytanie takie zwraca wiele wierszy danych (a nie pojedynczą wartość albo pojedynczy wiersz danych).
- Nie można w takim przypadku używać operatorów arytmetycznych porównania (=, >, <, <>, !=, <=, >=).
- Należy używać operatorów **IN**, **NOT IN**, **ANY** lub **ALL**. Operatory **ANY** oraz **ALL** łączymy z operatorami arytmetycznymi np. = ANY, > ALL itp.

```
SELECT column_list  
FROM table
```

```
WHERE (exp_1, exp_2, ..., exp_n) operator ( SELECT exp_I, exp_II, ...,  
exp_N  
FROM table );
```

Jeśli podzapytanie może zwrócić więcej niż jeden wiersz danych to używamy operatorów IN, NOT IN, lub ALL, ANY w połączeniu z opt. arytmetycznymi

Podzapytanie nieskorelowane w klauzuli WHERE (podzapytanie zwraca wiele wierszy - przykład)

→ Skonstruuj zapytanie, które zwróci wszystkich pracowników zatrudnionych w zakładach o id managera = 101 lub 112.

```
SELECT name, surname
FROM employees
WHERE department_id IN
(
    SELECT department_id
    FROM departments
    WHERE manager_id IN (101,
112)
);
```

Zastosowanie operatora IN

```
SELECT name, surname
FROM employees
WHERE department_id = ANY
(
    SELECT department_id
    FROM departments
    WHERE manager_id IN (101,
112)
);
```

Zastosowanie operatora ANY

Podzapytanie nieskorelowane w klauzuli WHERE (podzapytanie zwraca wiele wierszy - operator ALL)

→ Skonstruuj zapytanie, które zwróci wszystkich mężczyzn zatrudnionych wcześniej niż pracownicy na stanowiskach = 102 lub 103.

```
SELECT name, surname, gender
FROM employees
WHERE (date_employed, gender) < ALL
(
    SELECT date_employed, 'M'
    FROM employees
    WHERE position_id IN (102, 103)
);
```

Ćwiczenie - podzapytanie zwraca wiele wierszy danych

1. Napisz zapytanie, które zwróci informacje o pracownikach zatrudnionych po zakończeniu wszystkich projektów (tabela projects). *Zapytanie zrealizuj na 2 sposoby i porównaj wyniki*
2. Napisz zapytanie, które wyświetli wszystkich pracowników, których zarobki są co najmniej czterokrotnie większe od zarobków jakiegokolwiek innego pracownika.
3. Korzystając z podzapytań napisz zapytanie które zwróci pracowników departamentów mających siedziby w Polsce.
4. Zmodyfikuj poprzednie zapytania tak, żeby dodatkowo pokazać maksymalną pensję per departament.

Podzapytania skorelowane

Podzapytania skorelowane

- Pozwalają połączyć zapytania zewnętrzne z podzapytaniem warunkiem złączenia.
- Wykonywane w nieco inny sposób niż podzapytania nieskorelowane: (i) dla każdego wiersza zapytania nadrzędnego (ii) przeszukiwane są wszystkie wiersze podzapytania.
- Podzapytania skorelowane mogą zwracać zarówno **zero wierszy**, **jeden wiersz danych**, jak i **wiele wierszy danych** (oczywiście trzeba odpowiednio dobrać operator - reguły są takie same jak w przypadku podzapytań nieskorelowanych).

Podzapytania skorelowane - przykład 1

→ Wyświetl imię, nazwisko i zarobki wszystkich pracowników, którzy zarabiają więcej niż średnia na stanowisku które zajmują.

```
SELECT e1.name, e1.surname, e1.salary
FROM employees e1
WHERE e1.salary >
(
    SELECT avg(e2.salary)
    FROM employees e2
    WHERE e1.position_id =
e2.position_id
);
```

Używamy aliasów dla zbiorów danych (tabel) o ile mają te same nazwy

Podzapytanie zwraca jeden wiersz danych dla każdego *position_id* z tabeli *e1*, dlatego możliwe jest użycie operatora *>*.

Warunek złączenia cechujący podzapytanie skorelowane.

Podzapytania skorelowane - przykład 2

→ Wyświetl imię, nazwisko i zarobki wszystkich pracowników, którzy zarabiają co najmniej tyle samo co inni pracownicy pracujący na tych samych stanowiskach i zatrudnieni w zakładach o id = 101, 102, 103 lub 104.

```
SELECT e1.name, e1.surname, e1.salary
FROM employees e1
WHERE e1.salary >= ALL
(
    SELECT e2.salary
    FROM employees e2
    WHERE e1.position_id = e2.position_id
    AND e2.department_id IN (101, 102, 103, 104)
    AND e2.position_id IS NOT NULL
);
```

Zastosowanie operatora ALL (podzapytanie może zwrócić więcej niż jeden wiersz danych).

Warunek złączenia.

Podzapytania skorelowane - operator EXISTS

- Operator EXISTS (NOT EXISTS) pozwala na przeszukanie rezultatów podzapytania ze względu na wiersze obecne w zapytaniu nadrzędnym.
- Może być stosowany zarówno w przypadku podzapytań skorelowanych, jak i nieskorelowanych.
- Nie jest poprzedzony wyrażeniem.

Operator EXISTS - przykład

→ *Napisz zapytanie, które zwróci wszystkich pracowników będących kierownikami (managerami).*

```
SELECT name, surname  
FROM employees e1  
WHERE EXISTS
```

Operator EXISTS nie jest poprzedzony wyrażeniem

```
(  
    SELECT e2.employee_ID  
    FROM employees e2  
    WHERE e1.employee_id = e2.manager_id  
);
```

Warunek złączenia. Jeśli się nie pojawi (podzapytanie nieskorelowane), to EXISTS zwróci wt. prawdę, jeśli podzapytanie zwróci co najmniej jeden wiersz danych.

Ćwiczenia - podzapytania skorelowane

1. Napisz zapytanie, które zwróci pracowników zarabiających więcej niż średnia w ich departamencie.
2. Napisz zapytanie które zwróci regiony nieprzypisane do krajów
3. Napisz zapytanie które zwróci kraje nieprzypisane do regionów
4. Napisz zapytanie, które zwróci wszystkich pracowników niebędących managerami.
5. Napisz zapytanie, które zwróci dane pracowników, którzy zarabiają więcej niż średnie zarobki na stanowisku, na którym pracują
6. Za pomocą podzapytania skorelowanego sprawdź, czy wszystkie stanowiska zdefiniowane w tabeli Positions są aktualnie zajęte przez pracowników.

Podzapytania w klauzuli SELECT

Podzapytania w klauzuli SELECT

→ *Przykład: korzystając z jednego zapytania (i podzapytań) wyświetl liczbę wierszy w tabelach Employees, Positions i Departments.*

Każde z podzapytań musi zwracać tyle samo wierszy

SELECT

```
(SELECT COUNT(*) FROM EMPLOYEES) emp_row_count,  
(SELECT COUNT(*) FROM DEPARTMENTS) dept_row_count,  
(SELECT COUNT(*) FROM POSITIONS) pos_row_count
```

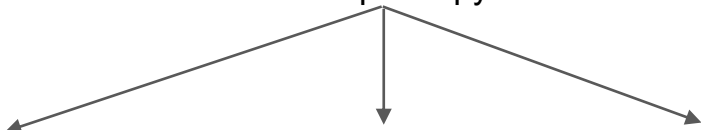
FROM DUAL;

Podzapytania w klauzuli FROM

Podzapytania w klauzuli FROM

→ *Przykład: korzystając z jednego zapytania (i podzapytań) wyświetl liczbę wierszy w tabelach Employees, Positions i Departments.*

Używamy aliasów dla oznaczenia podzapytań
i kolumn w ramach podzapytań



```
SELECT Sq1.emp_row_count, Sq2.dept_row_count, Sq3.pos_row_count
FROM (SELECT COUNT(*) emp_row_count FROM EMPLOYEES) Sq1,
     (SELECT COUNT(*) dept_row_count FROM DEPARTMENTS) Sq2,
     (SELECT COUNT(*) pos_row_count FROM POSITIONS) Sq3;
```

Ćwiczenia - podzapytania w SELECT/FROM

1. Napisz zapytanie, które dla wszystkich pracowników posiadających pensję zwróci informację o różnicy między ich pensją, a średnią pensją pracowników. Różnicę podaj jako zaokrągloną wartość bezwzględną.
2. Korzystając z poprzedniego rozwiązania, napisz zapytanie, które zwróci tylko tych pracowników, którzy są kobietami i dla których różnica do wartości średniej jest powyżej 1000.
3. Zmodyfikuj poprzednie zapytanie tak aby obliczyć liczbę pracowników.
(skorzystaj z podzapytania)
4. Napisz zapytanie które zwróci informacje o pracownikach zatrudnionych po zakończeniu wszystkich projektów (tabela projects). W wynikach zapytania umieść jako kolumnę datę graniczną.
5. Napisz zapytanie które zwróci pracowników którzy uzyskali w 2019 oceny wyższe niż średnia w swoim departamencie. Pokaż średnią departamentu jako kolumnę.



Praca domowa

1. Skonstruuj po jednym zapytaniu, które będzie zawierać w klauzuli WHERE:
 - a. podzapytanie zwracające tylko jedną wartość;
 - b. podzapytanie zwracające jeden wiersz danych, ale wiele kolumn;
 - c. podzapytanie zwracające jedną kolumnę danych;
 - d. podzapytanie zwracające tabelę danych.
2. Napisz zapytanie, które zwróci pracowników będących kierownikami zakładów, o ile ich zarobki są większe niż średnia zarobków dla wszystkich pracowników.
3. Zmodyfikuj powyższe zapytanie tak, aby wyświetlało wszystkich pracowników będących kierownikami zakładów, o ile ich zarobki są większe niż średnia zarobków na stanowisku które zajmują.
4. Wyszukaj informacje w Internecie, dokumentacji bazy danych Oracle lub w dostarczonych materiałach Oracle Academy o sposobie wykonywania podzapytań skorelowanych.



Praca domowa

5. W których klauzulach polecenia SELECT możemy wykorzystać podzapytania nieskorelowane?
6. W których klauzulach polecenia SELECT możemy wykorzystać podzapytania skorelowane?

⚡ Następne zajęcia - sprawdzian ⚡

Zakres:

- język SQL
 - polecenia SELECT, wszystkie klauzule
 - podzapytania
 - grupowanie
 - operacje na zbiorach (UNION, INTERSECT, MINUS, ...)
 - złączenia