**Zadanie 1 (Bazy danych i aplikacje)**

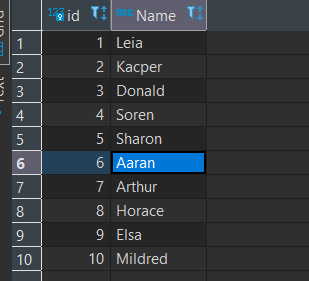
Zaprezentować działanie zapytań realizujących następujące operacje:

* Złączenie wewnętrzne (inner join)
* Złączenie naturalne
* Złączenie (zewnętrzne) lewe
* Złączenie (zewnętrzne) prawe
* Złączenie (zewnętrzne) pełne/obustronne

Zaprojektować tabele i umieścić w nich odpowiednie dane tak, aby było możliwe napisanie i wykonanie odpowiednich zapytań.

Posługujemy się wybranym środowiskiem bazodanowym (preferaby Postgresql).

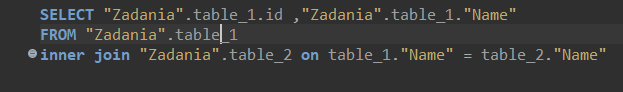
table\_1

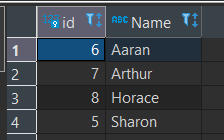


table\_2

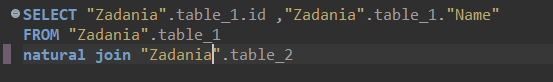


Złączenie wewnętrzne (inner join)



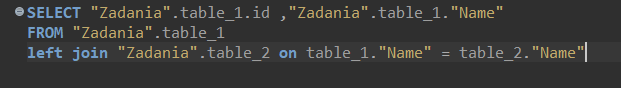


Złączenie naturalne



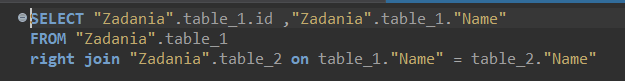


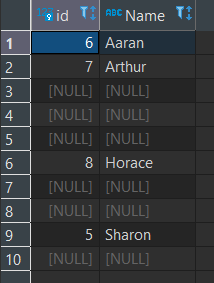
Złączenie (zewnętrzne) lewe



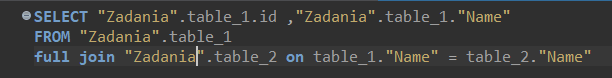


Złączenie (zewnętrzne) prawe





Złączenie (zewnętrzne) pełne/obustronne





INNER JOIN

Jeśli wartości te są równe, złączenie wewnętrzne tworzy nowy wiersz, który zawiera kolumny z obu tabel i dodaje ten nowy wiersz do zbioru wyników.

NATURAL JOIN

Złączenie naturalne to złączenie, które tworzy niejawne złączenie na podstawie tych samych nazw kolumn w łączonych tabelach.

LEFT JOIN

Jeśli wartości te są równe, lewe złączenie tworzy nowy wiersz zawierający kolumny z obu tabel i dodaje ten nowy wiersz do zbioru wyników.

W przypadku, gdy wartości nie są równe, lewe złączenie również tworzy nowy wiersz zawierający kolumny z obu tabel i dodaje go do zbioru wyników. Jednakże kolumny prawej tabeli zostają wypełnione wartością null.

RIGHT JOIN

Jeśli wartości te są równe, to prawe złączenie tworzy nowy wiersz zawierający kolumny z obu tabel.

W przypadku, gdy wartości te nie są równe, złączenie prawe również tworzy nowy wiersz, który zawiera kolumny z obu tabel. Wypełnia jednak kolumny w lewej tabeli wartością NULL.

FULL OUTER JOIN

Pełne złączenie zewnętrzne zwraca zestaw wyników, który zawiera wszystkie wiersze z lewej i prawej tabeli oraz pasujące wiersze z obu stron, jeśli są dostępne. W przypadku braku dopasowania, kolumny tabeli zostaną wypełnione NULL.

**Zadanie 2 (Bazy danych i aplikacje)**

Napisać program, który realizuje analogiczne do zaprezentowanych na wykładzie operacje z wykorzystaniem JDBC

**Zadanie 3 (Bazy danych i aplikacje)**

Na struktury bazy danych zaprojektowane w ramach rozwiązania Zadania 1 proszę nałożyć indeksy:

* posługując się samodzielnie przeprowadzoną analizą zapytań i ich planów
* proszę również porównać plany zapytań sprzed wprowadzenia indeksów oraz po ich wprowadzeniu
* na podstawie powyższego porównania ocenić, na ile indeksacja jest skuteczna
* Hint: EXPLAIN … oraz EXPLAIN ANALYZE … (w Postgresql), w innych bazach danych - proszę znaleźć odpowiedniki

Postgres automatycznie tworzy indeksy dla kluczy głównych

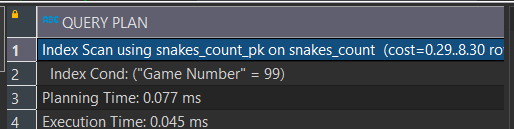
Stworzyłem nową tabelę z danymi z:

<https://people.sc.fsu.edu/~jburkardt/data/csv/csv.html>

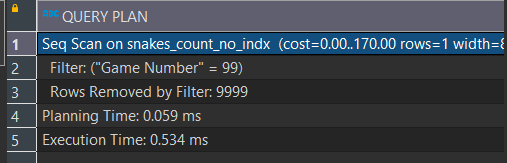
snakes\_count\_10000.csv

SELECT

Indeks

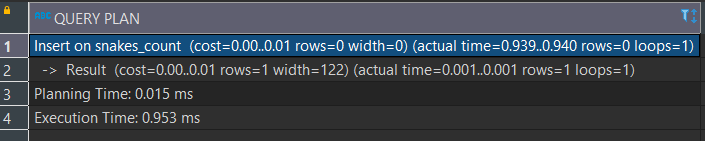


Bez indeksu

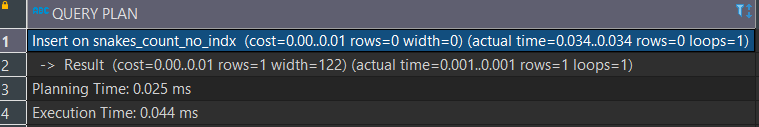


INSERT

Indeks

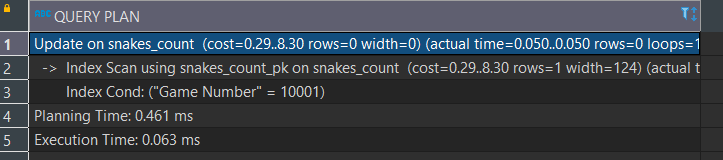


Bez indeksu

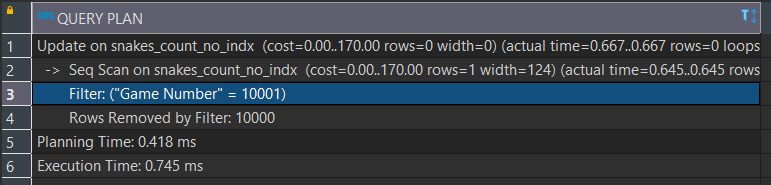


UPDATE

Indeks:

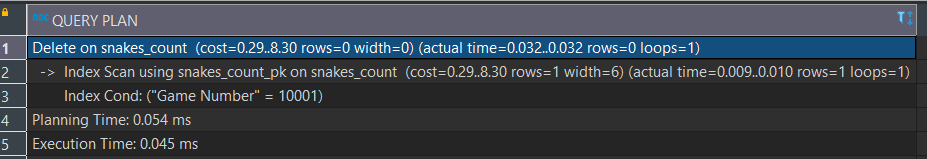


Bez indeksu:



DELETE

Indeks:



Bez indeksu:

