Przetwarzanie danych typu CLOB

oraz przeszukiwanie pełno tekstowe.

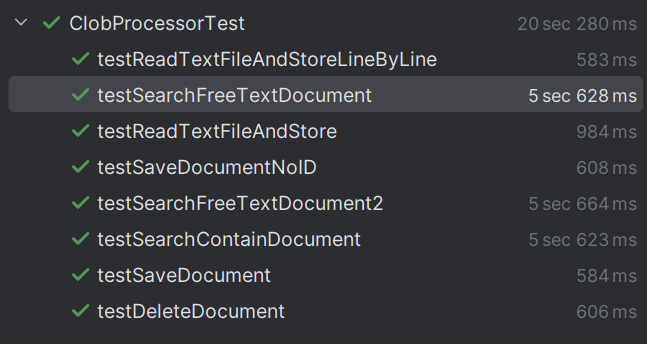
Szymon Kolanko

1. Opis problemu.  
   Celem projektu jest opracowanie i implementacja API do przetwarzania danych typu CLOB (Character Large Object). API ma umożliwiać efektywne zarządzanie dużymi dokumentami tekstowymi oraz zapewniać funkcjonalność wyszukiwania pełnotekstowego (FTS - Full Text Search). W ramach projektu należy stworzyć rozwiązanie pozwalające na zapis, usuwanie oraz wyszukiwanie informacji w dokumentach przechowywanych w repozytorium.
2. Wymagania funkcjonalne
   1. Zapis dokumentu:
      1. API powinno umożliwiać zapis dokumentu tekstowego typu CLOB do bazy danych.
   2. Usunięcie dokumentu:
      1. API powinno umożliwiać usunięcie dokumentu z bazy danych na podstawie unikalnego identyfikatora.
   3. Wyszukiwanie informacji:
      1. API powinno umożliwiać wyszukiwanie pełnotekstowe w dokumentach na podstawie podanych kryteriów wyszukiwania.
3. Opis funkcjonalności.  
   API umożliwia utworzenie bądź podłączenie się pod dowolną tabelę, która posiada jedną unikalną kolumnę primary key, o wartości innej niż null. API umożliwia dodawanie obiektów CLOB w dwóch formach:
   1. Jako pojedyncze, niezależne pliki tekstowe.
   2. Jako ciąg obiektów CLOB zapisanych w jednym pliku tekstowym, oddzielonych znakami nowej linii.

Udostępniona została również możliwość usuwanie wybranego obiektu po kluczu głównym.

W kontekście przeszukiwania pełnotekstowe, interfejs programistyczny przygotowuje katalog pełnotekstowy dla zadanej tabeli. Następnie kolumny wybranej tabeli są indeksowane wg. wybranego języka.

1. Testy:  
   Testy zostały przeprowadzone z użyciem frameworka jUnit4 w środowisku InteliJ. Przeprowadzone zostały podstawowe testy jednostkowe poszczególnych funkcjonalności API.



1. Dokumentacja API
   1. **public void initDatabase() throws SQLException**Metoda inicjalizująca połączenie z bazą danych. Jednocześnie w odpowiedniej kolejności przygotowuje elementy potrzebne do przeszukiwania pełnotekstowego tabeli.
   2. **public Integer saveDocument(Integer ID, String documentContent) throws SQLException**Metoda zapisuje obiekt CLOB (documentContent). Argument ID może przyjmować wartość null, w tym wypadku wartość klucza głównego zostaje automatycznie nadana przez bazę danych.  
      Funkcja zwraca klucz główny wstawionego obiektu.
   3. **public ArrayList<String> searchContainDocument(String searchTerm) throws SQLException**Metoda szuka zadanego wyrazu (**searchTerm**) w aktualnie podłączonej tabeli za pomocą wyrażenia WHERE CONTAINS. Argumentem może być tylko pojedynczy wyraz. Metoda zwraca listę indeksów wierszy zawierających wyszukiwany wyraz.
   4. **public ArrayList<String> searchFreeTextDocument(String searchTerm) throws SQLException**  
      Metoda wyszukuje w tabeli wyrazu (**searchTerm**) w aktualnie podłączonej tabeli za pomocą wyrażenia WHERE FREETEXT, czyli odnajdu wartości, które odpowiadają **znaczeniu**, a nie tylko dokładnemu brzmieniu słów w warunku wyszukiwania. (np. liczba mnoga, deklinacja wyrazu). Metoda zwraca listę indeksów wierszy zawierających wyszukiwany wyraz.

**Przykładowa inicjalizacja klasy:**

