

**Sprawozdanie z przedmiotu**  
**Administrowanie internetowymi serwerami**  
**baz danych**

Laboratorium № 8

**Khoruzhyi Oleksii**

**grupa nr.1**

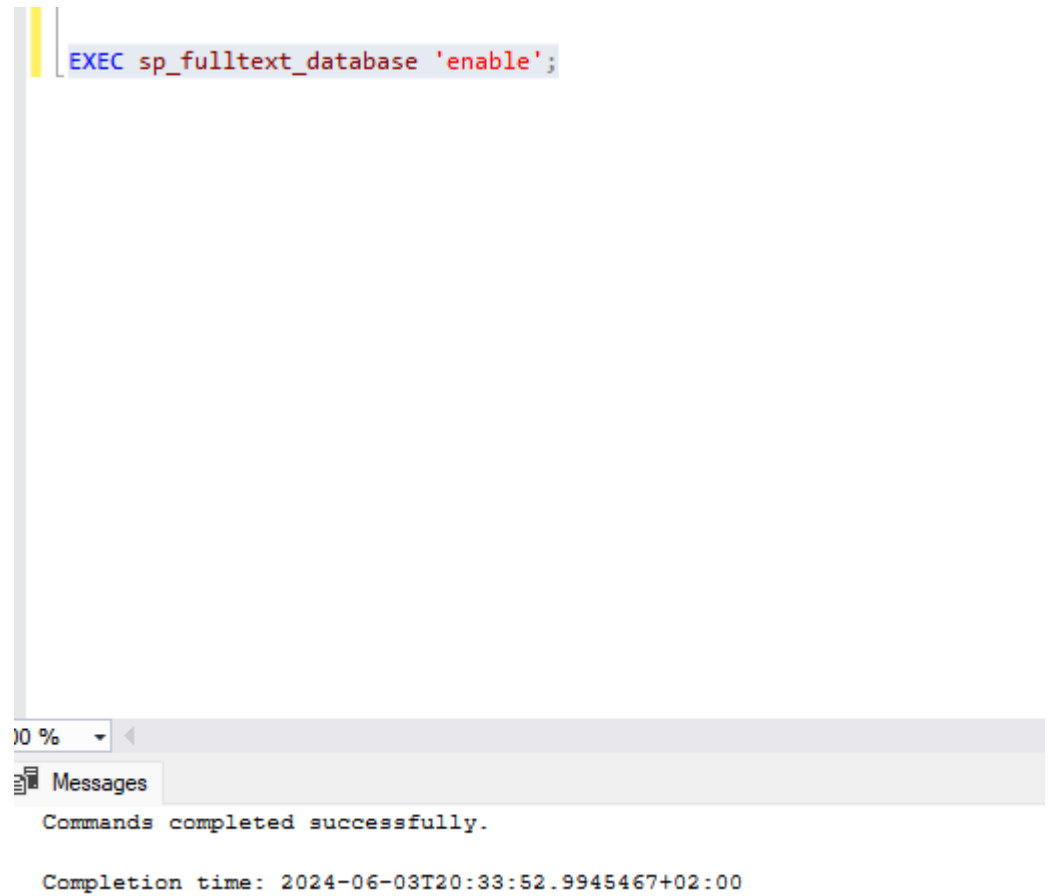
PAI WiMiI

**Zadania:**

- Utwórz na tabeli z poprzednich zajęć laboratoryjnych Full Text Index na dwóch kolumnach
- Przedstaw na przykładzie kilku zapytań jak działają polecenia Freetext oraz Contains
- Proszę napisz jaka jest różnica pomiędzy poleceniami: FREETEXT <->CONTAINS <-> LIKE
- Załóż FTS na tabeli typu FileTable Skopiuj do tej tabeli kilka plików z różną zawartością tekstową Przebuduj indeks Spróbuj zadać zapytania FTS tak by wybrać pliki o konkretnej zawartości Teraz skopiuj kilka plików MsOffice do FileTable Zainstaluj Microsoft Office filter pack Spróbuj zadać zapytania FTS do plików typu docx

## Zadanie a) Utwórz Full Text Index na dwóch kolumnach

### 1. Full-Text Search na serwerze:



The screenshot displays the SQL Server Enterprise Manager interface. The top pane shows a query window with the command `EXEC sp_fulltext_database 'enable';` entered. The bottom pane, titled "Messages", shows the output: "Commands completed successfully." and "Completion time: 2024-06-03T20:33:52.9945467+02:00".

```
EXEC sp_fulltext_database 'enable';
```

00 %

Messages

Commands completed successfully.

Completion time: 2024-06-03T20:33:52.9945467+02:00

## 2. Czy Full-Text Search jest zainstalowany:

```
SELECT
CASE FULLTEXTSERVICEPROPERTY('IsFullTextInstalled')
WHEN 1 THEN 'Full-Text installed.'
ELSE 'Full-Text is NOT installed.'
END;
```

Results Messages

(No column name)

Full-Text installed.

## 3. Full Text Index na tabeli FileStreamTable:

```
DESKTOP-Q8SJN7U (SQL Server 16.0.1115.1 - DESKTOP-Q8SJN7U\Ane...
Databases
├── System Databases
├── Database Snapshots
├── Lab_DB
├── lab1
├── Lab4Db
├── lab5
├── Security
├── Server Objects
├── Replication
├── Always On High Availability
├── Management
├── Integration Services Catalogs
├── SQL Server Agent (Agent XPs disabled)
└── XEvent Profiler

-- Wywołaj użycie danych
USE FileStreamDB;
GO

-- Utwórz unikalny indeks potrzebny do Full-Text Index
CREATE UNIQUE INDEX UIX_FileStreamTable_Id ON FileStreamTable(Id);
GO

-- Utwórz Full-Text Catalog
CREATE FULLTEXT CATALOG ftCatalog AS DEFAULT;
GO

-- Utwórz Full-Text Index
CREATE FULLTEXT INDEX ON FileStreamTable
(
    FileStreamData1,
    FileStreamData2
)
KEY INDEX UIX_FileStreamTable_Id
ON ftCatalog
WITH CHANGE_TRACKING AUTO;
GO
```

**Zadanie b) Przedstaw na przykładzie kilku zapytań jak działają polecenia Freetext oraz Contains**

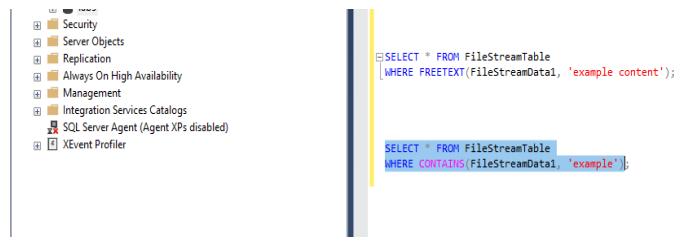
## 4. Przykład zapytania z użyciem FREETEXT:

```
Lab4Db
lab5
Security
Server Objects
Replication
Always On High Availability
Management
Integration Services Catalogs
SQL Server Agent (Agent XPs disabled)
XEvent Profiler

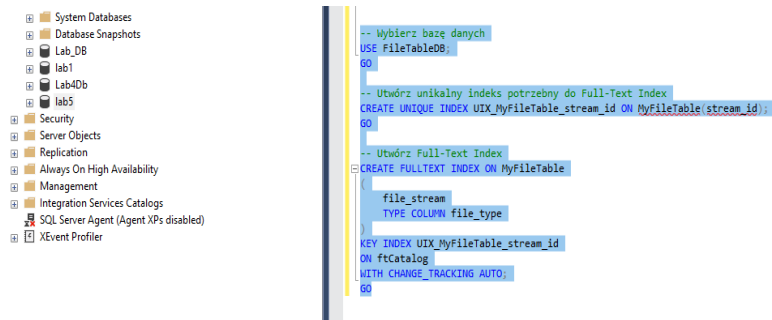
WITH CHANGE_TRACKING AUTO;
GO

SELECT * FROM FileStreamTable
WHERE FREETEXT(FileStreamData1, 'example content');
```

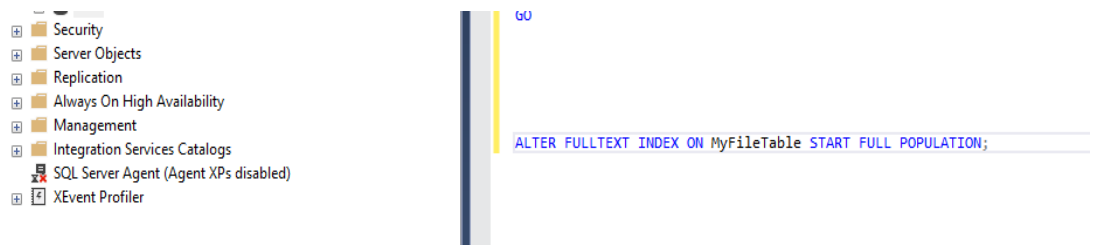
## 5. Przykład zapytania z użyciem CONTAINS



## 6. Utworzono Full Text Index na tabeli MyFileTable



## 7. Skopiowano pliki z różną zawartością tekstową do MyFileTable i przebudowano indeks:



## 8. Zadanano zapytania FTS do plików o konkretnej zawartości:



## 9. Skopiowano pliki .docx do katalogu FileTable, zainstalowano Microsoft Office Filter Pack i zadanano zapytania FTS do plików .docx:

SQL Server Agent (Agent XPs disabled)  
XEvent Profiler

```
SELECT name FROM MyFileTable  
WHERE CONTAINS(file_stream, 'office document content');
```

## Podsumowanie wykonanych zadań

### 1. Utworzenie tabeli z polami FILESTREAM

- **Utworzono bazę danych FileStreamDB:**

- Baza danych została utworzona z obsługą FILESTREAM, zawierającą główną przestrzeń danych oraz grupę FILESTREAM.
- Wykorzystano polecenia T-SQL do zdefiniowania struktury bazy danych i ścieżek do plików.

- **Utworzono tabelę FileStreamTable:**

- Tabela zawiera dwa pola typu FILESTREAM (**FileStreamData1** i **FileStreamData2**) oraz pole typu **uniqueidentifier** z automatycznym wypełnianiem wartości **newid()**.

### 2. Utworzenie tabeli FileTable

- **Utworzono bazę danych FileTableDB:**

- Baza danych została utworzona z obsługą FileTable oraz skonfigurowana do pełnego dostępu bez transakcji.

- **Utworzono tabelę MyFileTable:**

- Tabela została zdefiniowana jako FileTable z katalogiem **MyFileTableDirectory**.

### 3. Dodanie danych do obu tabel

- **Dodano dane do tabeli FileStreamTable:**

- Dodano przykładowe dane do pól FILESTREAM przy użyciu polecenia **INSERT**.

- **Dodano dane do tabeli MyFileTable:**

- Dodano przykładowy plik z zawartością tekstową do tabeli FileTable.

### 4. Przeglądanie katalogu FileTable

- **Skorzystano z opcji Explore FileTable Directory w SSMS:**

- Opcja pozwala na przeglądanie zawartości katalogu FileTable w systemie plików bezpośrednio z poziomu SQL Server Management Studio.

### 5. Wykonanie zapytania dotyczącego ścieżki pliku

- **Wykonano zapytanie SQL, aby uzyskać ścieżkę pliku:**

sql

Copy code

```
SELECT [1] [0] AS  
FROM [AS]
```

### 6. Utworzenie dodatkowej tabeli i powiązanie jej kluczem obcym

- **Utworzono tabelę RelatedTable:**

- Tabela zawiera dwa klucze obce powiązane z tabelami **FileStreamTable** i **MyFileTable**.

### 7. Weryfikacja zachowania tabeli FileTable przy usuwaniu powiązanego pliku

- **Przeprowadzono próbę usunięcia pliku z tabeli MyFileTable:**

- Sprawdzono, że usunięcie rekordu w tabeli **MyFileTable**, który jest powiązany kluczem obcym, spowoduje naruszenie ograniczenia klucza obcego.

## Zadania z notatki FTS

### Zadanie a) Utworzenie Full Text Index na dwóch kolumnach

- **Włączono Full-Text Search na serwerze:**
  - Użyto polecenia **sp\_fulltext\_database** oraz zweryfikowano instalację Full-Text Search.
- **Utworzono Full Text Index na tabeli FileStreamTable:**
  - Utworzono unikalny indeks oraz katalog Full-Text.
  - Dodano indeks Full-Text na kolumnach **FileStreamData1** i **FileStreamData2**.

### Zadanie b) Przykłady zapytań z użyciem FREETEXT i CONTAINS

- **Przykłady zapytań:**
  - Zapytania z użyciem FREETEXT oraz CONTAINS pokazują różnice w elastyczności i precyzji wyszukiwania.

### Zadanie c) Różnice między FREETEXT, CONTAINS a LIKE

- **FREETEXT:** Szuka dowolnego tekstu, który pasuje do znaczenia wyszukiwanego terminu.
- **CONTAINS:** Szuka dokładnych fraz lub prefiksów fraz.
- **LIKE:** Szuka dokładnych dopasowań wzorca z użyciem symboli wieloznacznych.

### Zadanie d) Założenie FTS na tabeli typu FileTable

- **Utworzono Full Text Index na tabeli MyFileTable:**
  - Utworzono unikalny indeks oraz Full-Text Index na kolumnie **file\_stream** typu FileTable.



- **Skopiowano pliki i przebudowano indeks:**

- Skopiowano pliki z różną zawartością tekstową oraz pliki typu **.docx**.
- Przebudowano indeks Full-Text.

- **Zadano zapytania FTS do plików o konkretnej zawartości.**

## **Tezaurus**

- **Edytowano plik tezaurya i dodano wpisy:**

- Plik tezaurya skonfigurowano zgodnie z instrukcjami, dodano wpisy i wykonano zapytania FTS, aby sprawdzić działanie tezaurya.