Sure, I'd be happy to explain Full-Text Search, including CONTAINS and FREETEXT, as well as Full-Text Indexes. Let's dive into each of these aspects.

### 1. CONTAINS and FREETEXT:

#### CONTAINS:
`CONTAINS` is a predicate used in Full-Text Search to find rows that contain a specified sequence of words. It is commonly used in the `WHERE` clause of a SQL query. The syntax typically involves specifying the column to search, the search condition, and the full-text index to use.

Example in T-SQL (Transact-SQL, used in Microsoft SQL Server):

```sql
-- Assuming you have a full-text indexed column named 'content'
SELECT *
FROM documents
WHERE CONTAINS(content, 'search_term');
```

This query will retrieve all rows from the "documents" table where the "content" column contains the specified search term.

#### FREETEXT:
`FREETEXT` is another predicate used in Full-Text Search, but it is more flexible than `CONTAINS`. It allows you to perform a free-text query against a column, searching for words and phrases that are similar to the specified search terms. It's often used when you want to find documents that contain similar content.

Example in T-SQL:

```sql
-- Assuming you have a full-text indexed column named 'content'
SELECT *
FROM documents
WHERE FREETEXT(content, 'search_term');
```

This query will retrieve all rows from the "documents" table where the "content" column contains content similar to the specified search term.

### 2. Full-Text Indexes:

Full-Text Indexes are database indexes built on columns that have been configured for Full-Text Search. These indexes improve the speed of searching large amounts of text data.

#### Example 1: Creating a Full-Text Index in SQL Server:

```sql
-- Creating a Full-Text Index on the 'content' column in the 'documents' table
CREATE FULLTEXT INDEX ON documents(content);
```

This statement creates a Full-Text Index on the "content" column in the "documents" table.

#### Example 2: Using Full-Text Index in a Query:

```sql
-- Searching using the Full-Text Index
SELECT *
FROM documents
WHERE CONTAINS(content, 'search_term');
```

In this query, the `CONTAINS` function is used to search the "content" column using the Full-Text Index created earlier.

These examples are specific to Microsoft SQL Server, and the syntax may vary for other database management systems that support Full-Text Search.

Certainly! Let's continue with more information on Full-Text Search and provide additional examples.

### 3. Full-Text Search Configurations:

Full-Text Search can be configured to support various language-specific settings, noise words, and other options. The configuration settings influence how the search engine processes and interprets text.

#### Example: Configuring Full-Text Search for a Specific Language:

```sql
-- Configuring Full-Text Search for English language
ALTER FULLTEXT INDEX ON documents
SET STOPLIST = SYSTEM, LANGUAGE = 'English';
```

This example shows how to configure the Full-Text Search index on the "documents" table to use the English language and a specific stoplist (a list of commonly used words that are ignored in searches).

### 4. Advanced Full-Text Search:

Advanced Full-Text Search involves using additional features like proximity searches, weighted searches, and customizing the search behavior.

#### Example: Proximity Search with NEAR in SQL Server:

```sql
-- Searching for documents where 'apple' and 'fruit' appear near each other
SELECT *
FROM documents
WHERE CONTAINS(content, 'apple NEAR fruit');
```

The `NEAR` operator allows you to find documents where the specified terms are close to each other.

#### Example: Weighted Full-Text Search:

```sql
-- Searching for documents with weighted terms
SELECT *
FROM documents
WHERE CONTAINS(content, 'apple WEIGHT (0.8) OR fruit WEIGHT (0.2)');
```

This query assigns different weights to the terms, indicating their relative importance in the search.

### 5. Full-Text Search in PostgreSQL:

Full-Text Search is not exclusive to Microsoft SQL Server; it's also supported in other database systems like PostgreSQL. Below is an example in PostgreSQL:

```sql
-- Creating a Full-Text Search index in PostgreSQL
CREATE INDEX documents_content_fts_idx
ON documents
USING gin(to_tsvector('english', content));
```

```
-- Performing a Full-Text Search in PostgreSQL
SELECT *
FROM documents
WHERE to_tsvector('english', content) @@ to_tsquery('english', 'search_term');
```

In this PostgreSQL example, a Full-Text Search index is created, and then a Full-Text Search is performed using the `to_tsvector` and `to_tsquery` functions.

Remember, the syntax and features can vary between database systems, so it's essential to consult the documentation for the specific database you are using.