

Bazy Danych

LABORATORIUM 1

Operatory warunkowe:

W SQL mamy do dyspozycji operatory warunkowe:

- $=$ - równości
- \neq lub $<>$ - nierówności
- $>$ - ostrej większości
- \geq - nieostrej większości
- $<$ - ostrej mniejszości
- \leq - nieostrej mniejszości
- *BETWEEN* $< wartosc > AND < wartosc >$ - przynależności do przedziału
- *LIKE* $< wzorzec >$ - spełnianie wzorca
- *IN* ($< wartosc >, < wartosc >, \dots$) - przynależności do zadanego zbioru
- *IS NULL* - zawierania wartości NULL

Operatory logiczne:

W SQL mamy dostęp do operatorów logicznych:

- AND
- OR
- NOT

Operacje agregujące (statystyczne):

Operacje wykonywane na kolumnach. W SQL dostępne są operacje statystyczne:

- *SUM(kolumna)* - oblicza sumę wartości w danej kolumnie

- *AVG(kolumna)* - oblicza średniej wartości w danej kolumnie
- *MIN(kolumna)* - służy do wyznaczenia najmniejszej wartości w danej kolumnie
- *MAX(kolumna)* - służy do wyznaczenia największej wartości w danej kolumnie
- *COUNT(kolumna)* - służy do zliczenia liczby wierszy w danej kolumnie

Komentarze w SQL:

Aby wykomentować linię w SQL należy wprowadzić dwa znaki minusa po sobie.

```
1 -- komentarz jednolinijkowy
```

Komentarz wieloliniowy otwiera się znakiem `/*` a zamyka znakiem `*/`.

```
1 /*
2   Komentarz
3   wieloliniowy
4 */
```

Typy danych w SQL:

W języku SQL mamy dostęp do poniższych typów danych:

- *INTEGER* - liczby całkowite
- *FLOAT* - liczba zmiennoprzecinkowa
- *DECIMAL(precyzja, skala)* - liczba zmiennoprzecinkowa, której precyzję (ilość cyfr w liczbie) oraz skalę (ilość miejsc po przecinku) można dowolnie ustalić.
- *CHAR(długość)* - ciąg znaków o stałej długości podanej
- *VARCHAR(maksymalna długość)* - przechowuje ciąg znaków o zmiennej długości (ale zawsze nie większej niż długość maksymalna)
- *BOOLEAN* - wartości logiczne
- *DATE* - daty
- *TIME* - godziny
- *TIMESTAMP* - daty i godziny

- *BLOB* - dane binarne czyli obrazy, pliki dźwiękowe itd.

Tworzenie nowej tabeli:

Aby stworzyć nową tabelę należy użyć słowa kluczowego CREATE TABLE.

```
1 CREATE TABLE <nazwa tabeli> (
2   <nazwa kolumny> <typ danych> [opcjonalna flaga kolumny],
3   <nazwa kolumny> <typ danych> [opcjonalna flaga kolumny],
4   ...
5   <nazwa kolumny> <typ danych> [opcjonalna flaga kolumny],
6   [opcjonalne flagi tabeli]
7 );
```

Usuwanie tabeli:

Aby usunąć tabelę należy użyć słowa kluczowego DROP TABLE.

```
1 DROP TABLE <nazwa tabeli>;
```

Dodawanie rekordu do tabeli:

Dodać rekord do tabeli można za pomocą słowa kluczowego INSERT INTO. W komendzie specyfikuje się nazwy kolumn jakie mają zostać uzupełnione oraz wartości, którymi będą uzupełnione.

```
1 INSERT INTO <nazwa tabeli>
2 ( <nazwa kolumny>, ... , <nazwa kolumny> )
3 VALUES
4 ( <wartosc>, ... , <wartosc> );
```

Polecenie DESCRIBE:

Polecenie DESCRIBE służy do wyświetlenia informacji o strukturze tabeli i jej kolumnach. Zwraca ona dane:

- *Name* - nazwa kolumny
- *Type* - typ danych przechowywanych w kolumnie
- *Null* - informacja, czy kolumna może przyjmować wartość NULL
- *Key* - informacja, czy kolumna jest kluczem (PRIMARY lub FOREIGN KEY)
- *Default* - wartość domyślna kolumny

```
1 DESCRIBE <nazwa tabeli>;
```

Wyrażenie SELECT:

Wyrażenie SELECT służy do pobierania danych z bazy danych. Jest to narzędzie do wyszukiwania, wybierania i pobierania określonych wartości z jednej lub wielu tabel w bazie danych.

```
1 SELECT <nazwa kolumny>, ... , <nazwa kolumny>
2 FROM <nazwa tabeli>
3 WHERE [warunek]
4 GROUP BY [nazwa kolumny], ... , [nazwa kolumny]
5 HAVING [warunek]
6 ORDER BY [kolumna + ASC|DESC], ... , [kolumna + ASC|DESC];
```

Warunek WHERE:

Warunek WHERE służy do wybrania wyłącznie tych rekordów, które spełniają określone kryteria. W warunku występującym po warunku WHERE można zawrzeć wyrażenie logiczne zawierające operatory porównania, porównania liczbowego. Można również użyć operatorów logicznych AND, OR i NOT, które pozwalają łączyć wielu warunków w jedną całość.

```
1 SELECT * -- gwiazdka oznacza wybranie wszystkich kolumn
2 FROM produkty
3 WHERE cena > 10;
```

Klauzula ORDER BY:

Klauzulę ORDER BY używa się do posortowania wyników zapytania SELECT. Sortowanie może odbyć się względem jednej lub wielu kolumn. W wypadku sortowania po wielu kolumnach priorytet sortowania jest największy po kolumnie najbardziej po lewej stronie. Do nazwy kolumny można dodać flagę ASC lub DESC. Oznaczają one kolejna sortowanie rosnąco i malejąco.

```
1 SELECT *
2 FROM produkty
3 ORDER BY cena ASC;
```

Polecenie SELECT DISTINCT:

Polecenie służące do wyszukania i wyświetlenia wszystkich unikalnych wartości lub kombinacji wartości ze wskazanej kolumny lub grupy kolumn. Przydatne, gdy chcemy uniknąć niepotrzebnych, powtarzających się wartości przy wyszukiwaniu danych z tabeli.

```
1 SELECT DISTINCT <nazwa kolumny>, ... , <nazwa kolumny>
2 FROM <nazwa tabeli>
3 WHERE [warunek]
4 GROUP BY [nazwa kolumny], ... , [nazwa kolumny]
```

```
5 HAVING [warunek]
6 ORDER BY [kolumna + ASC|DESC], ... , [kolumna + ASC|DESC];
```

Klauzula AS:

Klauzulę AS można użyć do zmiany nazwy wyświetlanej w wyniku zapytania SELECT kolumny.

```
1 SELECT
2   <nazwa kolumny> AS <nowa nazwa>,
3   ... ,
4   <nazwa kolumny> AS <nowa nazwa>
5 FROM <nazwa tabeli>;
```

Można jej także użyć do ustalenia nazwy wyrażenia użytego w zapytaniu SELECT. W przeciwnym wypadku będzie ono nazwane tak jak użyty do jego wyliczenia wzór.

```
1 SELECT
2   <wyrażenie> AS <nazwa wyrażenia>,
3   ... ,
4   <wyrażenie> AS <nazwa wyrażenia>
5 FROM <nazwa tabeli>;
```