# Lekcja 16 17 18

## Temat: Język definiowania danych (DDL)

Instrukcje DDL:

* CREATE nazwaObiektu – tworzy nowy obiekt
* ALTER nazwaObiektu – mienia strukturę obiektu
* DROP nazwaObiektu – usuwa istniejący obiektu.

### uTWORZENIE BAZY DANYCH

CREATE DATABASE nazwa\_bazy

### Usunięcie bazy danych

DROP DATABASE nazwa\_bazy

### Tworzenie tabel

CREATE TABLE nazwa\_tabeli

(

Nazwa\_kolumny1 typ\_kolumny1 [atrybuty],

Nazwa\_kolumny2 typ\_kolumny2 [atrybuty],

…………

Nazwa\_kolumnyntyp\_kolumnyn [atrybuty]

);

### Usuwanie tabeli

DROP TABLE nazwa\_tabeli

### Tworzenie schematów

CREATE SCHEMA nazwa\_schematu

Do schematu przydzielamy obiekty bazy danych. Dzięki temu możemy nadawać uprawnienia dla schematów a nie oddzielnie dla tabel. Obiekty, które są tworzone za pomocą instrukcji CREATE SCHEMA znajdują się wewnątrz definiowanego schematu

Jawne przypisanie tabeli do schematu

CREATE TABLE nazwa\_schematu.nazwa\_tabeli

Niejawne przypisanie tabeli to wtedy gdy definiujemy tabelę a nie sprecyzujemy nazwy schematu. Wówczas tabela przypisana będzie domyślnie do schematu na którym aktualnie pracujemy. Nie zaleca się stosowania niejawnego przypisywania tabel do schematów.

Uwaga! Według aktualnej dokumentacji CREATE DATABASE I CREATE SCHEMA wykonują te same operacje

Przykład

Chcemy utworzyć bazę danych o schemacie:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |
| rejestracja |  | uzytkownik |  | artykul |  | ocena\_artykulu |
| id\_rejestracji |  | id\_uzytkownika |  | id\_artykulu |  | id\_oceny |
| imie |  | id\_rejestracji |  | id\_uzytkownika |  | id\_artykulu |
| nazwisko |  |  |  | tytul\_artykulu |  | id\_uzytk\_oceniajacego |
| e-mail |  |  |  | tresc\_artykulu |  | tresc\_oceny |
| login |  |  |  | data\_publikacji |  | dobra\_czy\_zla\_ocena |
| haslo |  |  |  |  |  |  |
| telefon |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

Tworzymy bazę danych:

CREATE DATABASE Artykuly;

Tworzymy tabelę rejestracja:

CREATE TABLE rejestracja

(id\_rejestracji int AUTO\_INCREMENT PRIMARY KEY,

imie varchar(20),

nazwisko varchar(30),

email varchar(20) not null,

login varchar(20) not null,

haslo varchar(30) not null,

telefon varchar(12)

);

Tworzymy tabelę użytkownik i dodaje klucz obcy i prywatny

CREATE TABLE uzytkownik

(

id\_uzytkownika int AUTO\_INCREMENT PRIMARY KEY,

id\_rejestracji int,

FOREIGN KEY (id\_rejestracji)

REFERENCES rejestracja(id\_rejestracji)

ON DELETE CASCADE

);

Tworzymy tabelę artykul

CREATE TABLE artykul

(id\_artykulu int AUTO\_INCREMENT NOT null,

id\_uzytkownikaint not null,

tytul\_artykulu varchar(20),

tresc\_artykulu text,

data\_publikacji date,

PRIMARY KEY(id\_artykulu, id\_uzytkownika)

);

Tworzymy tabele ocena\_artykulu

CREATE TABLE ocena\_artykulu

(

id\_ocenyint AUTO\_INCREMENT PRIMARY KEY NOT null,

id\_artykuluint NOT null,

id\_uzytk\_oceniajacegoint NOT null,

tresc\_ocenytext,

dobra\_czy\_zla\_ocenaboolean not null DEFAULT 1

);

### Zmiana struktury tabeli

ALTER TABLE nazwa\_tabeli zmiana

Przykład

W naszej bazie do tabeli ocena\_artykulu chcemy dodać kolumnę recenzja:

[ALTER](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/alter-table.html) [TABLE](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/alter-table.html) artykuly.ocena\_artykulu ADD recenzja [TEXT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/string-types.html);

A w tabeli rejestracja chcemy poprawić hasło:

ALTER TABLE artykuly.rejestracja

MODIFY COLUMN haslo VARCHAR(10);

Uwaga! Dla serwerów SQL Server/MS Access komenda będzie w postaci:

ALTER TABLE artykuly.rejestracja

ALTER COLUMN haslo VARCHAR(10);

Chcemy usunąć z tabeli ocena\_artykulu kolumnę dobra\_czy\_zla\_ocena

ALTER TABLE artykuly.ocena\_artykulu

DROP COLUMN dobra\_czy\_zla\_ocena;

### Atrybuty kolumn

#### Klucz podstawowy

Może być to jedna kolumna lub grupa kolumn, które określają klucz podstawowy (główny) który jednoznacznie określa wiersz w tabeli.

Przykład

id\_uzytkownikaint AUTO\_INCREMENT PRIMARY KEY

Przykład

PRIMARY KEY(id\_artykulu, id\_uzytkownika)

#### Klucz obcy

To jest pole, które odwołuje się do podobnego pola w innej tabeli, dzięki czemu pozawala łączyć tabele. Składnia:

nazwa\_kolumny\_w\_tworzonej\_tabelityp\_pola,

FOREIGN KEY (nazwa\_kolumny\_w\_tworzonej\_tabeli)

REFERENCES nazwa\_tabeli\_nadrzednej(nazwa\_kolumny\_w\_tabeli\_nadrzednej)

[ON DELETE opcja]

[ON UPDATE opcja]

Przykład

…

id\_rejestracjiint,

FOREIGN KEY (id\_rejestracji)

REFERENCES rejestracja(id\_rejestracji)

ON DELETE CASCADE

…

#### Not null

Ta opcja pilnuje by nie można było wstawić w kolumnę wartości pustej.

#### Identity i auto\_Increment

Obydwie właściwości pozwalają na automatyczną inkrementację wartości

Składnia Identity (dla MSSQL)

IDENTITY(wartosc\_poczatkowa, wielkość\_kroku)

Sposób użycia AUTO\_INCREMENT (MySQL):

Id INT NOT NULL AUTO\_INCREMENT,

#### Default

Używane by określić wartość domyślną dla pola

Przykład

Liczba INT DEFAULT 1

#### Unique

Właściwość, która wymusza by wszystkie wartości w kolumnie nie powtarzały się.

Przykład

Nazwa varchar(10) NOT NULL UNIQUE,

#### warunek logiczny check

Atrybut do sprawdzania warunków logicznych. Oprócz znaków: <,<=,>,>= można używać, BETWEEN wart1 AND wart2, NOT, AND, OR.

Przykład 8

CREATE TABLE osoba (  
 ...  
    Wiek int,  
    CHECK (Wiek>=18)  
);

### ZADANIA

Zadanie

Wykonaj ćwiczenia tworzenia bazy danych z tabelami (oraz importem oraz eksportem) dostępne na stronie:

<http://www.teacher.webd.pl/kurs_mysql/k_3_4_0.php>

Zadanie

Na podstawie schematu utwórz bazę danych.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Klienci |  | Faktury |  | Szczegóły\_faktury |  | Towary |
| nip |  | numer\_faktury |  | numer\_faktury |  | identyfikator |
| pesel/regon |  | dzien |  | identyfikator\_towaru |  | kod\_towaru |
| imie |  | miesiac |  | ilosc |  | nazwa\_towaru |
| nazwisko/nazwa |  | rok |  | cena\_jednostkowa |  | jednostka\_miary |
| adres |  | klient |  | adres |  | ilosc\_w\_pakowaniu |
| telefon |  | sposob\_platnosci |  | telefon |  | podatek\_vat |
|  |  | termin\_platnosci |  |  |  | stan\_w\_magazynie |

Uwzględnij następujące założenia:

* Zdefiniuj klucze główne i obce dla przedstawionego schematu
* Przy definicji tabeli należy sprawdzać czy cena, ilość towaru na fakturze jest wartością różną od zera
* Pole nip nie może być puste,
* Pole numer\_faktury ma automatycznie wzrastać o 1
* Pole ilość powinno mieć wartość domyślną 100
* Pole rok ma mieć wartość domyślną 2018
* Pole kod towaru nie może się powtarzać
* Pole ilość\_w\_opakowaniu powinno się zawierać między 1 a 100

Następnie zmodyfikuj wyżej stworzone tabele, w sposób:

* W tabeli towary było dodatkowe pole opis\_towaru
* Aby pole nazwisko mogło mieć maksymalnie 40 znaków,
* Aby identyfikator w tabeli towary miało nazwę id\_towaru
* W tabeli Faktury zamiast kolumn dzien, miesiąc i rok należy wstawić jedną kolumnę data\_wystawienia

Na koniec wyeksportuj bazę danych pliku.

## Bibliografia

1. Tomasz Klekot, Krzysztof Pytel, Repetytorium+Testy Egzamin zawodowy, Kwalifikacja E.14 Tworzenie aplikacji internetowych i baz danych oraz administrowanie bazami, WSiP, Warszawa 2015
2. Tomasz Klekot, Krzysztof Pytel, Pracownia aplikacji internetowych, WSiP, Warszawa 2016
3. [Jolanta Pokorska](https://helion.pl/autorzy/jolanta-pokorska), Kwalifikacja E14. Część 2. Tworzenie baz danych i administrowanie bazami. Podręcznik do nauki zawodu technik informatyk, Helion Edukacja, Gliwice 2013
4. Dokumentacja MySQL, <https://dev.mysql.com/doc/refman/8.0/en/create-database.html>, data dostępu: 16.09.2018
5. Tutorial SQL W3schools, <https://www.w3schools.com>, data dostępu: 17.09.2018