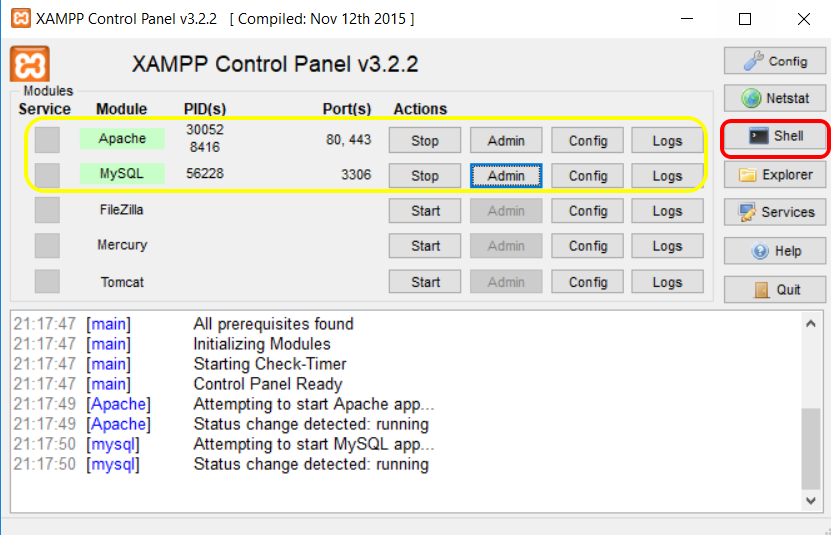
MySQL w konsoli

## Uruchamianie serwera

## 1



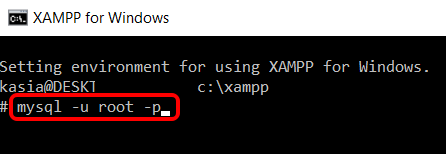
Rysunek 1.Włączany aplikację XAMPP i uruchamiamy serwer MYSQL Klikamy na button SHELL - uruchamiamy konsolę

## 2. Operacje wykonywane na serwerze MySQL poprzez konsolę

### **Logowanie do serwera MySQL**

Do administracji bazami MySQL najczęściej używa się domyślnie tworzonego konta o nazwie **root**, o nieograniczonych możliwościach. Czasami, ze względów bezpieczeństwa, tworzy się osobne konto o tych samych uprawnieniach jednak z inną nazwą użytkownika.

mysql -u root –p - logowanie do konsoli



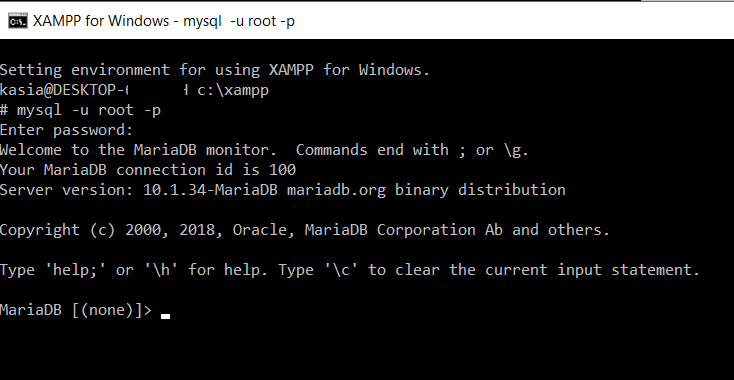
Rysunek 2. Wpisujemy polecenie: mysql -u root –p i enter



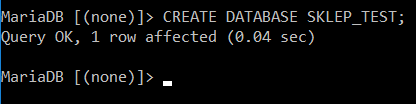
Rysunek 3.hasło dla root-a na localhost jest puste więc naciskamy ENTER



Jesteśmy już zalogowani



CREATE DATABASE - Tworzymy naszą bazę danych "SKLEP\_TEST" poleceniem:

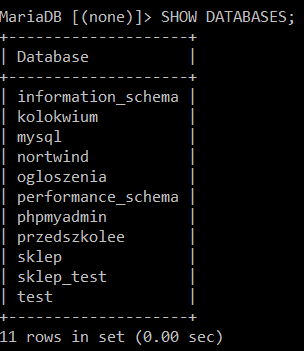
CREATE DATABASE SKLEP\_TEST;

Można użyć polecenia do tworzenia bazy z kodowaniem dla polskich znaków:

**CREATE DATABASE nazwa\_bazy\_danych DEFAULT CHARACTER SET utf8 DEFAULT COLLATE utf8\_general\_ci;**

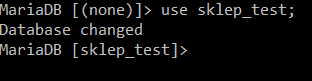
**SHOW DATABASES;** - Wyświetlamy wszystkie bazy danych

**DROP DATABASE nazwa\_bazy\_danych; -**usuwanie bazy danych



Rysunek 4. Polecenie pokazania wszystkich baz.

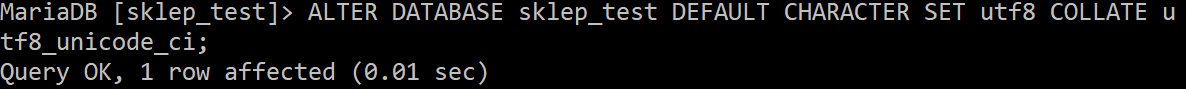
USE sklep\_test; - wybieramy naszą bazę



Kodowanie oraz metodę porównywania napisów dla bazy danych zmieniamy zapytaniem:

**ALTER DATABASE nazwa\_bazy DEFAULT CHARACTER SET utf8 COLLATE utf8\_unicode\_ci;**

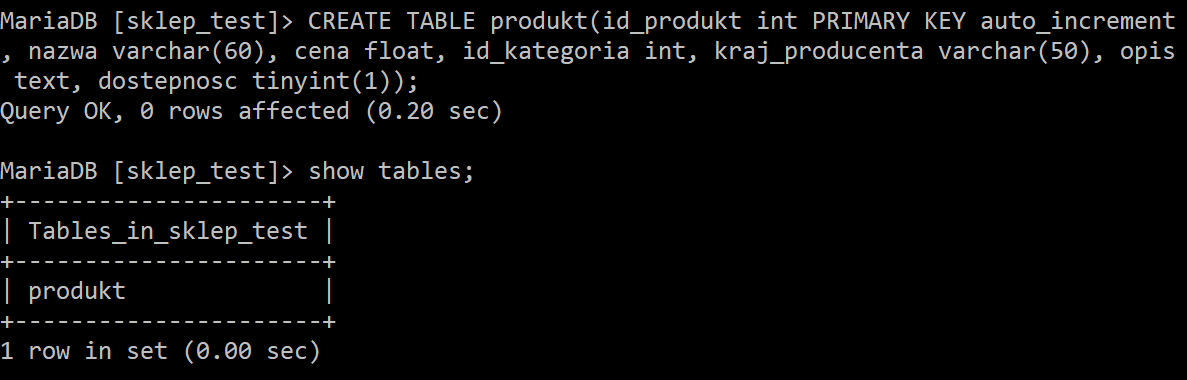
ALTER DATABASE sklep\_test DEFAULT CHARACTER SET utf8 COLLATE utf8\_unicode\_ci;



CREATE TABLE - Tworzymy tabelę produkt:

CREATE TABLE produkt(id\_produkt int PRIMARY KEY auto\_increment, nazwa varchar(60), cena float, id\_kategoria int, kraj\_producenta varchar(50), opis text, dostepnosc tinyint(1));

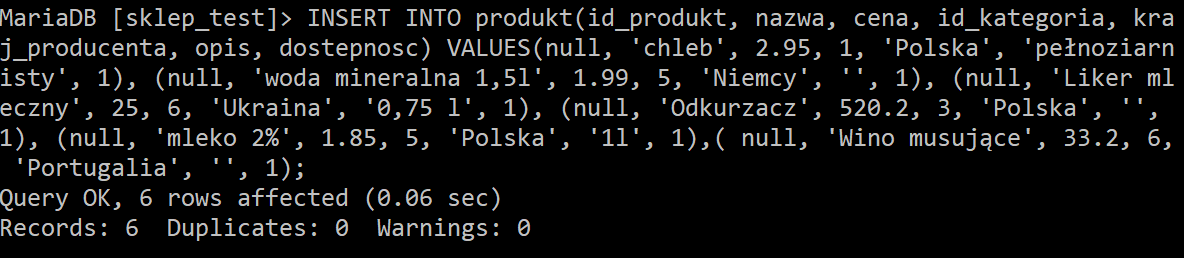
SHOW TABLES - sprawdzamy zawartość naszej bazy



Rysunek 5.Tworzenie i wyświetlanie tabeli

INSERT INTO - wstawiamy rekordy do tabeli / jednym poleceniem po przecinku wstawiam kilka rekordów

INSERT INTO produkt(id\_produkt, nazwa, cena, id\_kategoria, kraj\_producenta, opis, dostepnosc) VALUES(null, 'chleb', 2.95, 1, 'Polska', 'pełnoziarnisty', 1), (null, 'woda mineralna 1,5l', 1.99, 5, 'Niemcy', '', 1), (null, 'Liker mleczny', 25, 6, 'Ukraina', '0,75 l', 1), (null, 'Odkurzacz', 520.2, 3, 'Polska', '', 1), (null, 'mleko 2%', 1.85, 5, 'Polska', '1l', 1),( null, 'Wino musujące', 33.2, 6, 'Portugalia', '', 1);

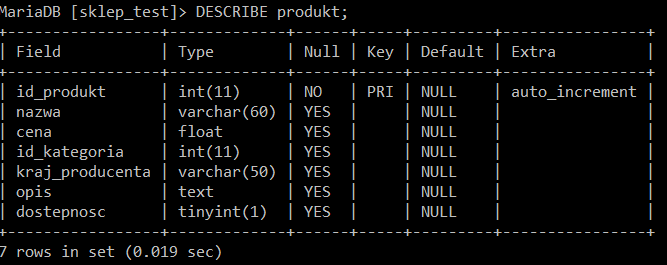


Rysunek 6. Wstawianie rekordów.

SELECT - wypisujemy rekordy z tabeli



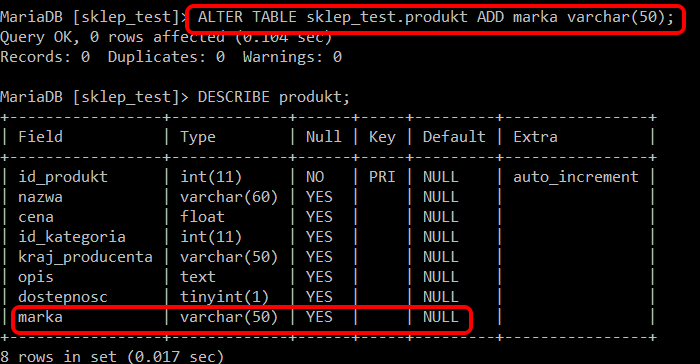
**DESCRIBE nazwa\_tabeli;**



To co powyżej zostało wyświetlone to nazwy pól, typ danych, jakie dane pole przechowuje, czy można wartość **Null** wprowadzić, czy dane pole jest kluczem identyfikacyjnym, domyślna wartość pola i dodatkowe opcje.

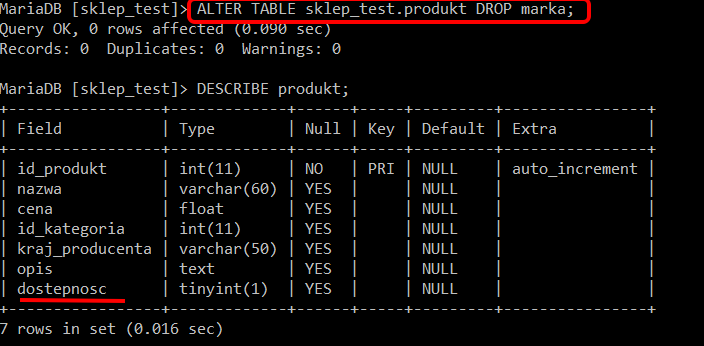
## Dodawanie nowej kolumny danych do tabeli

ALTER TABLE sklep\_test.produkt ADD marka varchar(50);



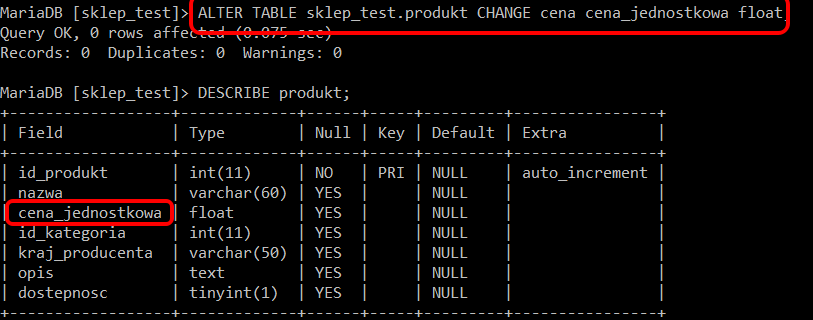
## Usuwanie kolumn danych z tabeli

ALTER TABLE sklep\_test.produkt DROP marka;



Modyfikacja nazwy pól tabeli:

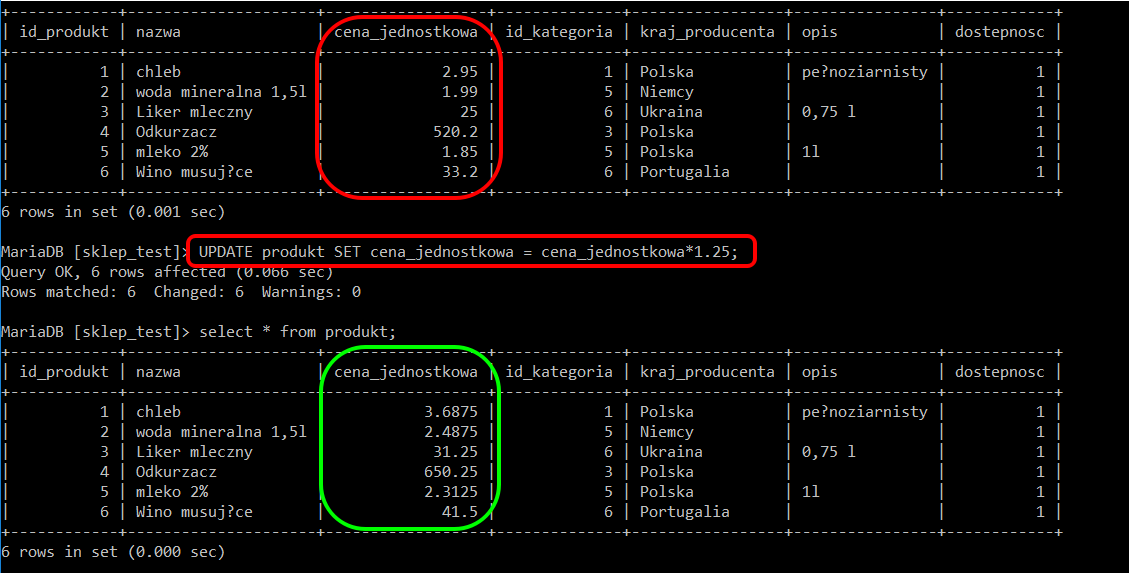
ALTER TABLE sklep\_test.produkt CHANGE cena cena\_jednostkowa float;



## Update- zmiana wartości rekordów:

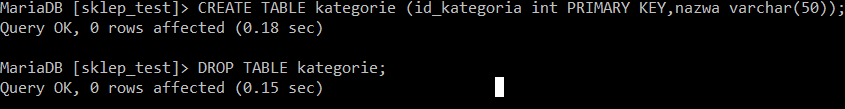
Zwiększenie ceny jednostkowej o 25%

UPDATE produkt SET cena\_jednostkowa = cena\_jednostkowa\*1.25;



## Tworzę tabelę kategorie i wstawiam do niej rekordy.

CREATE TABLE kategorie (id\_kategoria int PRIMARY KEY,nazwa varchar(50));



Rysunek 7. Tworzenie i usuwanie tabeli kategorie

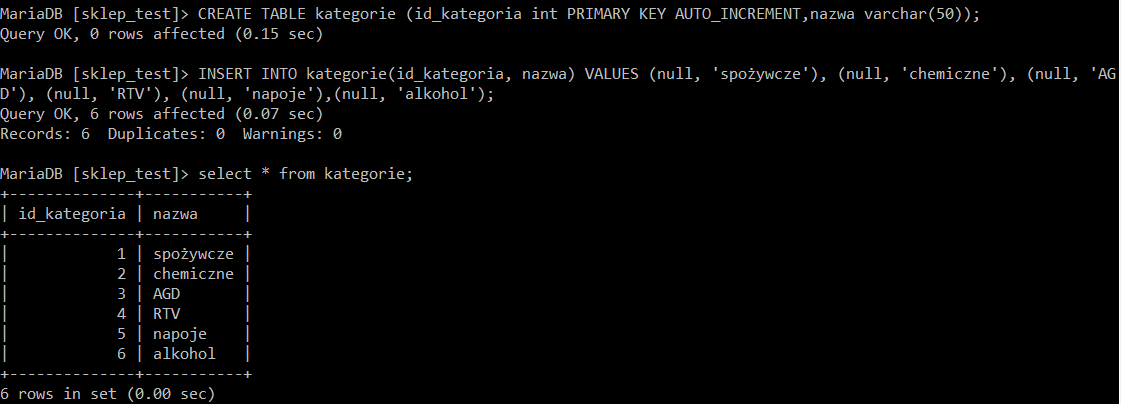
DROP - usuwamy tabelę kategorie;

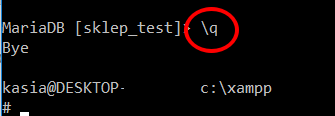
Modyfikuje polecenie o auto\_increment

CREATE TABLE kategorie (id\_kategoria int PRIMARY KEY AUTO\_INCREMENT,nazwa varchar(50));

A następnie wstawiam rekordy do tabeli kategrie:

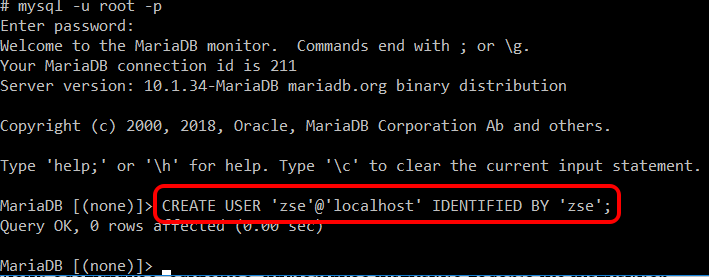
INSERT INTO kategorie(id\_kategoria, nazwa) VALUES (null, 'spożywcze'), (null, 'chemiczne'), (null, 'AGD'), (null, 'RTV'), (null, 'napoje'),(null, 'alkohol');



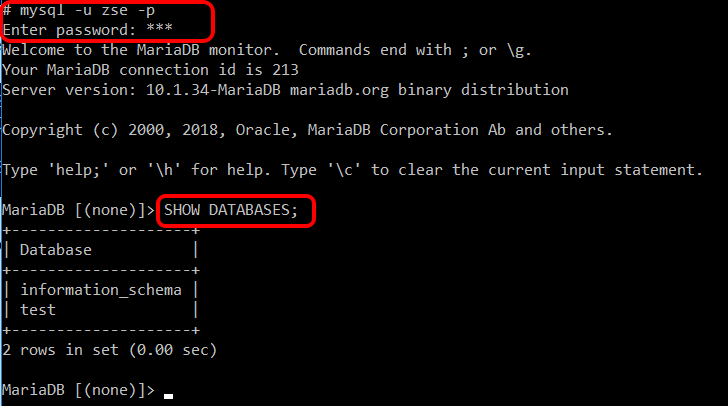
\q Wylogowanie z serwera MySQL

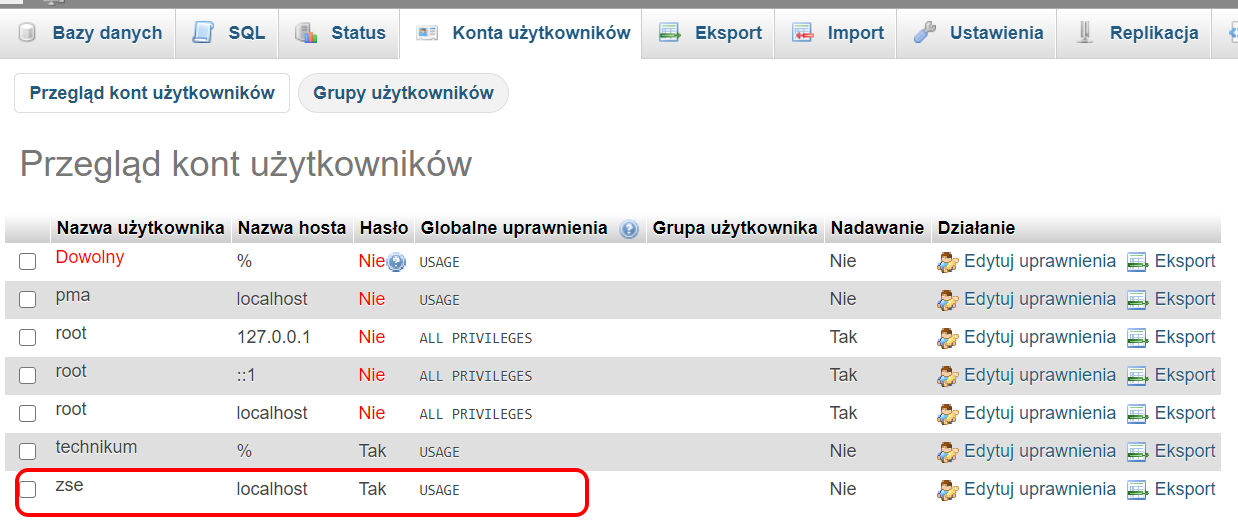
# Jak utworzyć nowego użytkownika MySQL**[[1]](#endnote-1)**

Po zalogowaniu się do konsoli jako root należy dodać nowego użytkownika poleceniem:



Niestety, w tym miejscu nowy użytkownik nie jest upoważniony do wniesienia jakichkolwiek zmian w bazę danych. W rzeczywistości, jeśli nowy użytkownik próbuje się zalogować (z hasłem zse), nie będzie mógł zobaczyć wszystkich baz danych- tylko testową.



****

Rysunek 8. W phpMyAdmin widać utworzonego użytkownika zse USAGE -czyli bez przypisanych uprawnień

Dlatego, najpierw trzeba zapewnić użytkownikowi dostęp do informacji, do której będzie chciał się dostać.  
**GRANT ALL PRIVILEGES ON \* . \* TO 'zse'@'localhost';**

**Przykład ustawiania uprawnień:**

GRANT ALL PRIVILEGES ON nortwind.klienci TO 'zse'@'localhost';

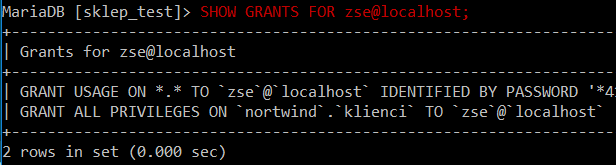
Gwiazdki w tym poleceniu odnoszą się do bazy danych oraz tabeli (odpowiednio), do których mają dostęp. Ta komenda pozwala użytkownikowi odczytywać, edytować, wykonywać wszystkie zadania we wszystkich bazach danych i tabelach.

Kiedy skończysz z uprawieniami, które chcesz konfigurować pod swoich nowych użytkowników, upewnij się, że ponownie załadowałeś wszystkie przywileje.  
**FLUSH PRIVILEGES;** Przeładowanie uprawnień, twoje zmiany już weszły w życie.

**SHOW GRANTS; - pokazanie uprawnień**

Możliwe jest również wyświetlenie uprawnień dla dowolnego użytkownika, a można to osiągnąć w następujący sposób:

**SHOW GRANTS FOR zse@localhost;**

****

# Jak dodać uprawnienia do użytkownika

Oto krótka lista innych możliwych uprawień, z których użytkownicy mogą korzystać.

* ALL PRIVILEGES –te uprawienie umożliwi użytkownikom MySQL dostęp do całej bazy danych
* CREATE – pozwala tworzyć tabele i bazy danych
* DROP – pozwala usuwać tabele i bazy danych
* DELETE – pozwala usuwać wierszy z tabel
* INSERT – pozwala wklejać wierszy w tabele
* SELECT – pozwala używać komendy Select do odczytywania zawartości bazy danych
* UPDATE – pozwala aktualizować wiersze w tabelach
* GRANT OPTION – pozwala nadawać lub pozbawić przywilej innych użytkowników

Żeby upoważnić określonego użytkownika, należy:  
**GRANT [uprawnienie] ON [nazwa bazy danych].[nazwa tabeli] TO ‘[username]’@'localhost’;**

Jeżeli chcesz nadać im dostęp do każdej bazy danych czy tabeli, wpisz gwiazdkę (\*) w miejscu nazwy bazy danych lub tabeli.

Każdego razu, gdy aktualizujesz albo zmieniasz uprawnienia, używaj komendy **Flush Privileges.**

Czasami trzeba nadać użytkownikowi pewne większe uprawnienia, aby po chwili zastanowienia odebrać mu część przypisanych mu uprawnień. Oto jak można ograniczyć uprawnienia użytkownikowi **zse**,

REVOKE ALL PRIVILEGES ON nortwind.klienci FROM 'zse'@'localhost'; GRANT SELECT, INSERT, UPDATE, REFERENCES ON nortwind.klienci TO 'zse'@'localhost';

Usuwanie rekordów:

**DELETE FROM klienci WHERE ID\_klienta = '1';**

****

Jeśli chcesz cofnąć uprawnienia – polecenie jest prawie identyczne do ich udzielenia:  
**REVOKE [uprawnienie] ON [nazwa bazy danych].[nazwa tabeli] FROM ‘[username]’@‘localhost’;**

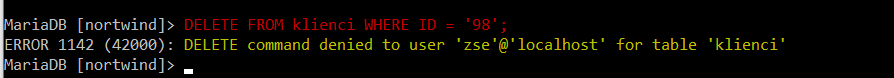
Za pomocą komendy DROP możesz usunąć nie tylko bazę danych, ale i użytkownika:  
**DROP USER ‘tester’@‘localhost’;**

Żeby sprawdzić nowego użytkownika – wyloguj się, wpisując  
**quit** lub **\q;**

Potem zaloguj się ponownie za pomocą tej komendy w terminalu:

**mysql -u [login] -p**

Po przelogowaniu na użytkownika zse, użytkownik ten nie może usunąć rekordu



Polecenie DELETE odmówiono użytkownikowi zse usunięcia rekordu.

## Usuwanie całej zawartości tabeli

Kasowanie rekordu po rekordzie przy milionie pozycji może być nieco męczące. Twórcy SQL pomyśleli również i o tym aspekcie dodając instrukcję TRUNCATE TABLE. Ogólna składnia:

**TRUNCATE TABLE nazwaTabeli;**

Czasami konieczne jest wykonanie zrzutu danych zapisanych w bazie danych do pliku **\*.sql** w celu np. przeniesienia zapisanych w pliku danych do bazy danych umieszczonej na serwerze. Można też utworzyć kopię zapasową bazy danych w ten właśnie sposób. Oto jak można z poziomu konsoli systemowej to systemu **Windows**,**Linux**:

**mysqldump -u root -p nazwa\_bazy > plik.sql**

Po zatwierdzeniu powyższej linijki kodu wyświetlony zostanie monit z prośbą o podanie hasła do bazy danych, jeżeli baza danych istnieje, **Apache** oraz **mysql** działają, wtedy utworzony zostanie plik z danymi tejże bazy danych.

Można również określić, które tabele mają zostać wyeksportowane z bazy danych do pliku **\*.sql** a to z kolei za sprawą następującego polecenia:

**mysqldump -u root -p nazwa\_bazy tabela1 tabela2 > plik.sql**

W wyniku działania powyższego kodu utworzony zostanie plik **\*.sql** zawierający dane z tabel **table1** i **table2** bazy danych **nazwa\_bazy**.

**Zadanie**

1. utwórz użytkownika uczen1na localhost z hasłem uczen123;
2. nadaj użytkownikowi uczen1 prawo do usuwania, dodawania oraz przeglądania danych w bazie nortwind na tabeli klienci.
3. nadaj użytkownikowi uczen1 wszystkie uprawnienia w tabeli produkty;
4. odbierz użytkownikowi uczen1 w tabeli produkty możliwość usuwania i modyfikacji rekordów.

1. [Jak utworzyć nowego użytkownika z odpowiednimi uprawnieniami w MySQL - Hostovita.pl](https://hostovita.pl/blog/jak-utworzyc-uzytkownika-mysql/) [↑](#endnote-ref-1)