

# MySQL

## Administracja

Zalogowanie się do MySQL za pomocą mysql:

```
$ mysql -u mysql -p mysql
Enter password:
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 28
Server version: 5.5.11-log PLD/Linux Distribution MySQL RPM

Copyright (c) 2000, 2010, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input
statement.

mysql>
```

W PLD administrator MySQL'a nazywa się mysql, w innych dystrybucjach może on się nazywać inaczej np: w Debianie będzie to root

Stworzenie użytkownika na bazie danych:

```
CREATE USER 'username'@'localhost' IDENTIFIED BY 'tajne_password';
```

Zmiana hasła dla użytkownika:

```
UPDATE mysql.user SET Password=PASSWORD('nowe_tajne_password') WHERE
User='username';
```

Usunięcie użytkownika z bazy danych:

```
DROP USER 'username'@'localhost';
```

Stworzenie instancji na bazie danych:

```
CREATE DATABASE dbname;
```

Nadanie praw użytkownikowi do bazy danych:

```
GRANT ALL PRIVILEGES ON dbname.* TO username@'localhost';
```

Tu mała uwaga: w MySQL'u użytkownik identyfikowany jest za pomocą nazwa@host, gdzie nazwa@'%' nie jest równa nazwa@localhost, więc gdy nasz użytkownik będzie logował się z zewnątrz (z poza localhost) to musimy nadać jemu do tego uprawnienia:

```
GRANT ALL PRIVILEGES ON dbname.* TO username@'%';
```

Gdy pierwszy raz nadawaliśmy użytkownikowi nazwa@'%' uprawnienia to MySQL tak naprawdę stworzył tego użytkownika i trzeba jemu nadać też hasło:

```
UPDATE mysql.user SET Password=PASSWORD('tajne_password') WHERE  
User='username';
```

To samo tyczy się usuwania, jeśli istnieje użytkownik name@localhost i name@'%' podczas usuwania name@localhost użytkownik name@'%' zostanie na bazie danych! Jeśli nie chcemy tego to też musimy go usunąć:

```
DROP USER 'username'@'%';
```

Usunięcie instancji na bazie danych:

```
DROP DATABASE dbname;
```

Po nadaniu uprawnień należy odświeżyć uprawnienia komendą:

```
FLUSH PRIVILEGES;
```

Wyświetlenie listy użytkowników:

```
mysql> select host, user from mysql.user;
```

```
+-----+-----+  
| host      | user          |  
+-----+-----+  
| %         | ocs           |  
| 127.0.0.1 | mysql         |  
| 127.0.0.1 | mysql_sysadmin |  
| localhost |               |  
| localhost | mysql         |  
| localhost | mysql_sysadmin |  
| localhost | ocs           |  
| localhost | pong          |  
+-----+-----+  
8 rows in set (0.01 sec)
```

```
mysql>
```

Wyświetlenie listy instancji bazodanowych:

```
mysql> show databases;
```

```
+-----+
| Database          |
+-----+
| information_schema |
| mysql              |
| ocsweb             |
| pong               |
+-----+
4 rows in set (0.00 sec)

mysql>
```

## Backup

Sam czysty SQL:

```
$ data=`date +%Y_%m_%d`
mysqldump -u dbuser --password=tajnepass dbname > /backup/dbname_$data.sql
```

Jeśli baza danych posiada funkcję to należy też dumpować – służy do tego opcja -R:

```
mysqldump -u dbuser --password=tajnepass -R dbname >
/backup/dbname_$data.sql
```

Z kompresją:

```
mysqldump -u dbuser --password=tajnepass dbname | bzip2 >
/backup/dbname_$data.bz2
```

Bez danych – sam schemat:

```
mysqldump -u dbuser --password=tajnepass --no-data dbname >
/backup/dbname_$data.sql
```

Same dane – bez schematu:

```
mysqldump -u dbuser --password=tajnepass --skip-triggers --compact --no-
create-info dbname > /backup/dbname_$data.sql
```

Wszystkie bazy:

```
mysqldump -u mysql --password=tajnepass --all-databases >
/backup/dump_$data.sql
```

Odtwarzanie bazy danych:

```
mysql -u dbuser --password=tajnepass dbname < /backup/dump.sql
```

## Użytkowanie

Pokazanie struktury bazy danych:

```
mysql> show tables;
+-----+
| Tables_in_ocsweb |
+-----+
| accesslog         |
| accountinfo       |
| bios              |
| blacklist_macaddresses |
| blacklist_serials |
| .                 |
| .                 |
| .                 |
| slots             |
| softwares          |
| softwares_name_cache |
| sounds            |
| storages           |
| subnet            |
| tags              |
| videos            |
| virtualmachines   |
+-----+
54 rows in set (0.00 sec)

mysql>
```

Pokazanie struktury tabeli:

```
mysql> desc files;
+-----+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| NAME  | varchar(150)  | NO   | PRI | NULL    |       |
| VERSION | varchar(100) | NO   | PRI | NULL    |       |
| OS     | varchar(80)   | NO   | PRI | NULL    |       |
| CONTENT | longblob      | NO   |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.02 sec)

mysql>
```

From:

<https://kamil.orchia.pl/> - **kamil.orchia.pl**

Permanent link:

<https://kamil.orchia.pl/doku.php?id=mysql&rev=1364299175>

Last update: **2018/07/16 11:47**

