

Лабораторна робота №2

(два заняття)

Основи адміністрування СУБД MySQL

Мета роботи: здобути базові навички адміністрування СУБД MySQL в ОС UNIX.

Знання, які повиннен здобути студент в результаті виконання лабораторної роботи:

1. Вміння здійснити початкову перевірку стану готовності сервера до роботи, запустити та зупинити сервер, задати параметри при старті, здійснити визначення номеру порта та імені з'єднання-сокету, на яких запуснено сервер.
2. Створення та знищення баз даних. Поняття про різні типи storage engine в MySQL (MyISAM та InnoDB). Доступ до бази даних з командної стрічки.
3. Поняття про схему прав доступу до об'єктів в СУБД MySQL, спеціальна база даних mysql.
4. Маніпулювання правами доступу до даних в СУБД MySQL на як рівні SQL-команди GRANT так і на рівні схеми спеціальної бази даних mysql.
5. Виконання пакетних операцій по відновленню та резервному копіюванню даних. Поняття про реплікацію (тільки визначення).
6. Моніторинг сервера: отримання інформації про завантаженість, статус та активні процеси.
7. Способи відновлення даних в разі пошкодження структури бази даних. Утиліта myisamchk.
8. PhpMyAdmin - засіб доступу до баз даних MySQL через веб-інтерфейс.

Завдання до лабораторної роботи № 2:

1. Залогуватися в системі.
2. Створити піддиректорію *lab2/* в директорії *~/work/*. В неї поміщати всі додаткові файли (окрім *lab2.sh*), створені як результат виконання даної лабораторної роботи.
3. Створити shell-скрипт *lab2.sh* в директорії *~/work/* використовуючи шаблон з */home/common/labs/*, надалі всі команди, файли (з допомогою команди *cat*) а також необхідні пояснення (з допомогою команди *echo*) включати в цей скрипт.
4. Перевірити, чи на сервері запуснено процес СУБД MySQL з допомогою команди *ps -auxwww | grep mysql*
пояснити отриманий результат, звернути увагу на стартові параметри сервера.
5. З допомогою команди *sockstat | grep mysql*
визначити, на якому порті та сокеті запуснено сервер.
6. Дослідити стартовий скрипт сервера */usr/local/etc/rc.d/mysql-server.sh*, включити перших тридцять стрічок цього файлу в свій скрипт *lab2.sh* з допомогою команди *head -n 30 /usr/local/etc/rc.d/mysql-server.sh*
7. Переглянути директиви (змінні), які стосуються роботи mysql-сервера у файлі */etc/rc.conf*
Включити їх у свій скрипт з поясненнями.
8. Передивитися коротку довідку по утиліті *mysqladmin* з допомогою команди *mysqladmin -help*
9. Для того, щоб уникнути процесу введення паролю до mysql-сервера при виконанні кожної утиліти, з командної стрічки виконати команду

```
setenv MYSQL_PWD passwordValue
```

де *passwordValue* - ваш пароль.

Бажано також відредагувати свій локальний файл *~/.cshrc* і внести цю команду туди.

10. Створити базу даних *username* (замість *username* використати свій логін) з допомогою команди *mysqladmin*:

```
mysqladmin create username
```

Спробувати знищити її з допомогою команди

```
mysqladmin drop username
```
11. Зкопіювати в директорію *~/work/lab2/* файл *lab2.sql* з */home/common/labs/*.
Передивитися його вміст з допомогою команди *cat*:

```
cat ~/work/lab2/lab2.sql
```
12. Відкрити інтерактивний сеанс до *mysql*-сервера з допомогою команди

```
mysql username
```
13. Створити в межах сеансу одну-дві таблиці, використовуючи *sql* з зкопійованого вище файлу. Можна також виконати кілька запитів на внесення, видалення та вибірку даних (за бажанням).
14. Передивитися документацію по SQL-командах *mysql* для маніпулювання правами доступу до баз даних (<http://student.mr.lviv.ua/InformacijniTechnologijiVInfrastrukturiiRynku/docs/mysql-5.0/sql-syntax.html#database-administration-statements>)
15. З допомогою SQL-запиту типу GRANT надати права на вибірку із своєї бази даних комусь з інших користувачів-студентів

```
grant select on Object to SomeUsername;
```

Перевірити з допомогою SELECT-запиту доступ до таблиць баз даних користувачів, які надали такі права вам

```
select * from SomeDatabase.tablename;
```
16. Переглянути вміст спеціальної бази даних *mysql*:

```
use mysql;
```

```
show tables;
```

Переглянути структуру і дані таблиць *user* та *db* в цій базі даних:

```
desc user;
```

```
select * from user;
```

Включити вивід в скріпт та пояснити результат.
17. З допомогою команди *Ctrl+z* перевести сеанс до *mysql*-сервера в фоновий режим (не закривати з'єднання).
18. З командної стрічки виконати команду відновлення бази даних з зкопійованого вами *sql*-файлу *~/work/lab2/lab2.sql* :

```
mysql databasename < ~/work/lab2/lab2.sql
```
19. Створити резервну копію бази даних з допомогою команди *mysqldump*:

```
mysqldump --add-drop-table databasename > databasename.dump
```
20. Спробувати відновити базу даних з резервної копії з допомогою команди:

```
mysql databasename < databasename.dump
```
21. Переглянути документацію і зрозуміти, що таке реплікація даних.
22. Вивести на екран коротку довідку по команді *mysqladmin*:

```
mysqladmin --help
```

Включити перших 20 рядків з неї в скріпт з допомогою команди *head*:

```
mysqladmin --help | head -n 20
```
23. Переглянути активні процеси сервера та його статус з допомогою команди

```
mysqladmin proc stat
```

Включити вивід в скріпт.
24. Переглянути параметри роботи сервера з допомогою команди

```
mysqladmin variables
```

25. Переглянути документцію по утиліті *myisamchk* (<http://student.mr.lviv.ua/InformacijniTechnologijiVInfrastrukturiiRynku/docs/mysql-5.0/client-utility-programs.html#myisamchk>).
26. Переглянути довідку по утиліті *myisamchk* з допомогою команди *myisamchk -help*
Внести кілька перших рядків в скрипт.
27. Перевести mysql-сеанс знову в інтерактивний режим з допомогою команди *fg*.
28. Спробувати, як працює *myisamchk* в режимі mysql-сеансу: виконати наступні команди:
ANALYZE TABLE students;
REPAIR TABLE students;
OPTIMIZE TABLE students;
Включити ці команди в скрипт з поясненнями.
29. Закрити mysql-сеанс з допомогою команди *quit*.
30. Ознайомитися з роботою phpMyAdmin - програми доступу до баз даних MySQL через веб-інтерфейс за URL <http://student.mr.lviv.ua/phpmyadmin/> .
31. Оформити і захистити звіт.