

Федеральное государственное бюджетное образовательное учреждение высшего образования
Казанский государственный энергетический университет
Кафедра информационных технологий и интеллектуальных систем

Задачи с MySQL

Методические указания для выполнения лабораторных работ



Казань, 2024

Лабораторная работа. Задачи с MySQL

Интерфейс СУБД MySQL. Предоставление доступа и привилегий. Утилиты, входящие в состав СУБД

Цель работы: Ознакомиться с приложениями, включенными в состав СУБД MySQL. Получить навыки управления учетными записями пользователей и определения привилегий. Ознакомиться с утилитами, входящими в состав СУБД MySQL, получить навыки работы с ними.

Запуск MySQL

Управление сервером обычно осуществляется из командной строки. Запуск в Windows 95/98/2000/XP осуществляется через сеанс DOS выполнением следующей команды:

```
D:\usr\local\MySQL\bin\mysqld --standalone
```

Эта команда запустит демон mysql в фоновом режиме. В Windows 95/98 не предусмотрен запуск mysqld в виде службы. В Windows 2000 демон mysql запускается в виде службы.

Можно осуществить запуск winmysqladmin.exe, в этом случае все настройки перечисляются в файле my.ini

При запуске mysqld можно указывать следующие опции:

Таблица 1- Опции команды MySQLD

-?, --help	Справка
-b, --basedir=[path]	Путь к каталогу в котором установлен mysql
-h, --datadir [homedir]	Путь к каталогу, в котором хранятся базы данных.
-l, --log=[filename]	Имя журнала транзакций
-L, --language=[language]	Язык по умолчанию (обычно English).
-P, --port=[port]	Порт для соединения.
--skip-grant-tables	Игнорировать таблицы привилегий. Это дает любому ПОЛНЫЙ доступ ко всем таблицам. Не следует предоставлять обычным пользователям разрешений на запуск mysqld.
--skip-name-resolve	Позволяет предоставлять доступ только тем хостам, чьи IP-адреса указаны в таблицах привилегий. Ипользуется для более высокого уровня защиты.

`--skip-networking`

Использовать подключения только через интерфейс localhost.

-V, --version	Вывести информацию о версии.
---------------	------------------------------

Наличие в статусной строке иконки светофора с активным зеленым цветом указывает на то, что сервер запущен (см. рис 1).



Рисунок 4 - Приложение winmysqladmin запущено

Теперь можно попытаться войти в сервер. В случае, если предполагается управление сервером через консоль, то необходимо использовать команду **mysql**. Изначально существует единственный пользователь, которому предоставляется право входа - **root**, которая не имеет пароля. Первое, что нужно сделать войти под именем **root** и зарегистрировать нового пользователя и установить для него пароль. Команда **mysql** может использовать следующие опции:

Таблица 2 - Опции команды MySQL

-?, --help	Справка
-h, --hostname=[hostname]	Имя сервера mysql.
-u, --user=[user]	Имя пользователя для доступа к mysql.
-p, --password=[password]	Пароль пользователя для доступа к mysql.
-P, --port=[port]	Порт для соединения с сервером.
-V, --version	Информация о версии

Примечание. Команды **mysqld** и **mysql** имеют еще некоторые опции, но в данный момент они особого интереса не представляют.

Запуск из сеанса ДОС осуществляется как показано на Рисунок 5 (в указанном случае осуществляется подключение к БД mysql).

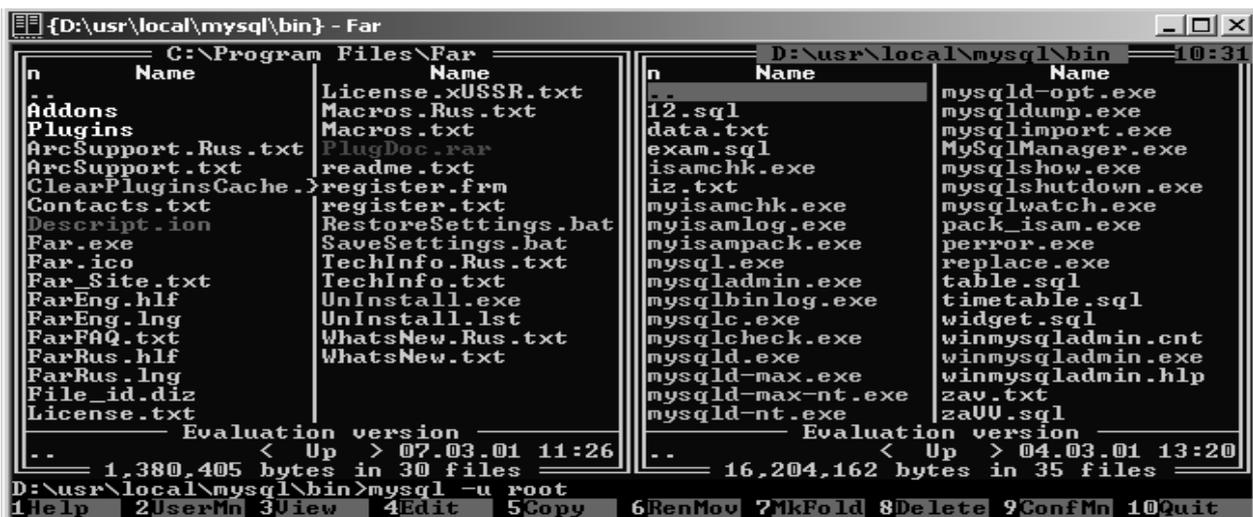


Рисунок 5 - Запуск консоли MYSQL

Для выполнения в строке наберите команду: **mysql -u root**

```
The FAR manager, version 1.70 beta 3 (build 591)
Copyright (C) 1996-2000 Eugene Roshal, Copyright (C) 2000-2001 FAR Group
Evaluation copy, please register.
D:\usr\local\mysql\bin>mysql -u root
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 3 to server version: 3.23.41-nt

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> _
```

Рисунок 6 - Успешный запуск консоли

Если вы это получили, значит вы успешно вошли в консоль mysql, которая используется для администрирования сервера.

Для составления отчета вам понадобятся приведение команд, которые вы будете посылать на сервер. В MySQL имеется возможность ведение протокола выполняемых команд, чтобы запустить ведение протокола необходимо выполнить команду

\T filename

!!! обязательно в верхнем регистре. Filename – имя файла, в который будут записываться команды (создается автоматически при выполнении команды, и действует во время жизни сеанса, т.е. в случае отключения от сервера лог прерывается и для возобновления необходимо повторить команду с выводом в новый файл, так как команда затирает имеющиеся в файле данные).

Просмотр списка БД, доступных на сервере осуществляется командой **SHOW DATABASES**.

Для выполнения в строке наберите команду: **show databases**.

Командой: **USE MYSQL;** – выбираем текущую БД где MYSQL имя БД.

Система привилегий и безопасность в MySQL

- User
- Db
- Host
- Пользовательские привилегии

База данных mysql и таблицы привилегий.

Итак, вы успешно вошли в базу данных mysql, которая используется для администрирования сервера. Что же здесь находится? А находятся здесь 5 таблиц, которые ничем не отличаются от других таблиц баз данных, за исключением того, что эти таблицы используются для предоставления доступа к базам данных и таблицам в них пользователям.

Рассмотрим каждую из них.

Введите следующую команду, `show tables`, которая покажет таблицы в базе данных mysql.

Кратко рассмотрим функции каждой из таблиц:

Таблица User

Определяет, разрешено ли пользователю, пытающемуся подключиться к серверу делать это. Содержит имя пользователя, пароль, а также привилегии. Если ввести команду `show columns from user;` то получим следующее:

Таблица 3- Структура таблицы User

Field	Type	Null	Key	Default	Extra
Host	char(60)		PRI		
User	char(16)		PRI		
Password	char(41)				
Select_priv	enum('N','Y')			N	
Insert_priv	enum('N','Y')			N	
Update_priv	enum('N','Y')			N	
Delete_priv	enum('N','Y')			N	
Create_priv	enum('N','Y')			N	
Drop_priv	enum('N','Y')			N	
Reload_priv	enum('N','Y')			N	
Shutdown_priv	enum('N','Y')			N	
Process_priv	enum('N','Y')			N	
File_priv	enum('N','Y')			N	
Grant_priv ¹	enum('N','Y')			N	
References_priv	enum('N','Y')			N	
Index_priv	enum('N','Y')			N	
Alter_priv	enum('N','Y')			N	
Show_db_priv	enum('N','Y')			N	
Super_priv	enum('N','Y')			N	
Create_tmp_table_priv	enum('N','Y')			N	
Lock_tables_priv	enum('N','Y')			N	
Execute_priv	enum('N','Y')			N	
Repl_slave_priv	enum('N','Y')			N	
Repl_client_priv	enum('N','Y')			N	
Create_view_priv	enum('N','Y')			N	
Show_view_priv	enum('N','Y')			N	
Create_routine_priv	enum('N','Y')			N	
Alter_routine_priv	enum('N','Y')			N	
Create_user_priv	enum('N','Y')			N	
Event_priv	enum('N','Y')			N	
Trigger_priv	enum('N','Y')			N	
ssl_type	enum('','ANY','X509','SPECIFIED')				
ssl_cipher	blob			NULL	
x509_issuer	blob			NULL	
x509_subject	blob			NULL	
max_questions	int(11) unsigned			0	
max_updates	int(11) unsigned			0	
max_connections	int(11) unsigned			0	
max_user_connections	int(11) unsigned			0	

¹Эта и все, описанные ниже команды добавлены начиная с версии 5.12

Изначально эта таблица содержит пользователя root без пароля. По умолчанию root может входить с любого хоста, имеет все привилегии и доступ ко всем базам данных. Также в таблице содержится запись для пользователя '%'.²

В БД MYSQL содержатся таблицы, называемых таблицами привилегий. Система привилегий будет подробно рассмотрена в следующих работах, а пока вы можете выполнить команды на добавления своего пользователя:

Для добавления нового пользователя **your_name**, можно выполнить следующие операторы языка (Insert):

```
Insert into user (host, user, password, ssl_cipher2,  
x509_issuer, x509_subject ) values ('localhost',  
'your_name', password('your_pass'), '', '', '');
```

Выполнением команды

```
Select host, user, password from user;
```

Мы выводим перечисленные поля в виде таблицы

Host	User	Password
%	root	456g879k34df9

Если необходимо выделить все столбцы таблицы, то необходимо набрать * в качестве аргумента команды *select*.

Чтобы изменения вступили в силу нужно перезагрузить сервер, предварительно закончив текущий сеанс работы командой *quit*.

```
mysqladmin -u root reload (эта команда перезагружает сервер)
```

После установки пароля для пользователя нужно перезагрузить сервер командой `mysqladmin reload`, чтобы изменения вступили в силу. После этого можно попробовать войти снова:

```
Mysql/bin/mysql -u your_name -p mysql  
Enter password:*****
```

Если же после этой операции вы не получите приглашение ко входу, то необходимо будет повторить вход в сервер под учетной записью **ROOT** и назначить необходимые права. Т.о., недостаточно добавить сведения о пользователе в системную БД, дополнительно необходимо назначить права пользователю, после чего можно начинать настраивать таблицы привилегий, вводить новых пользователей, создавать базы данных и таблицы, то есть делать все то, что называется администрированием. Назначить права можно указанием ин-

² Атрибуты *ssl_cipher², x509_issuer, x509_subject* обязательны для заполнения для версии сервера 5.12

струкцией *INSERT* для заполнения соответствующие привилегии (перечень привилегий см.

Таблица 3)

```
Mysql/bin/mysql -u root
```

И выполнить следующий запрос к БД:

```
Mysql>USE MYSQL;
```

```
Mysql>GRANT ALL PRIVILEGES ON *.* TO  
'your_name'@'localhost3' IDENTIFIED BY 'your_pass' WITH  
GRANT OPTION;
```

```
Mysql>FLUSH PRIVILEGES;
```

Если пароль был случайно забыт, чтобы его задать по новой, придется стереть файлы `mysql.frm` `mysql.MYI` и `mysql.MYD` из папки с базами данных, затем запустить скрипт `mysql_install_db` и повторить все по новой. Можно воспользоваться ключом `MYSQL` и ввести `--skip-grant-tables`, при этом все пароли будут имеет пустое поле.

Команда имеет вид *`mysqld --skip-grant-tables`*.

Пояснения:

1. Команда `insert` вставляет данные в таблицу, не забываяте завершать команды `'`.
2. При вводе пароля используйте функцию `password()`, иначе пароль работать не будет!
3. Все пароли шифруются `mysql`, поэтому в поле `Password` вы видите абракадабры. Это делается в целях безопасности.
4. Не есть хорошей практикой назначать привилегии пользователям в таблице `user`, так как в этом случае они являются глобальными и распространяются на все базы данных. Предоставляйте привилегии каждому пользователю к конкретной базе данных в таблице `db`, которая будет рассмотрена далее.
5. При задании имени хоста для входа через сеть рекомендуется явно указывать полное имя хоста, а не `'%'`. В приведенном выше примере пользователю `maru` разрешается вход на сервер со всех машин домена `tomsk.ru`. Можно также указывать IP-адреса машин и маски подсетей для большей безопасности.

Таблица Db

Определяет к каким базам данных каким пользователям и с каких хостов разрешен доступ. В этой таблице можно предоставлять каждому пользователю доступ к базам данных и назначать привилегии. Если выполнить команду *`show columns from db;`* получим следующее:

³ Для случая, если работаете на том же компьютере где запущен сервер БД

Таблица 4 - Структура таблицы Db

Field	Type	Null	Key	Default	Extra
Host	char(60)		PRI		
Db	char(32)		PRI		
User	char(16)		PRI		
Select_priv	char(1)			N	
Insert_priv	char(1)			N	
Update_priv	char(1)			N	
Delete_priv	char(1)			N	
Create_priv	char(1)			N	
Drop_priv	char(1)			N	

- По умолчанию, все привилегии установлены в 'N'. Например, предоставим юзеру mary доступ к базе данных mysql и дадим ему привилегии **select**, **insert** и **update** (описание основных команд mysql будет дано в следующих лабораторных работах, сейчас ваша цель увидеть, как работают таблицы привилегий).
- Для справки:

```
Insert into db (host, user, db, select_priv, insert_priv, update_priv)
Values ('localhost', 'your_name', mysql, 'Y', 'Y', 'Y');
```

- Привилегии, устанавливаемые в таблице db, распространяются только на базу данных library. Если же установить эти привилегии в таблице user, то они будут распространяться и на другие базы данных, даже если доступ к ним и не установлен явно.

Таблица Host

Таблица host используется для расширения диапазона доступа в таблице db. К примеру, если доступ к какой-либо базе данных должен быть предоставлен более чем одному хосту, тогда следует оставить пустой колонку host в таблице db, и внести в таблицу host необходимые имена хостов. Выполним команду

show columns from host;

Таблица 5 - Структура таблиц Host

Field	Type	Null	Key	Default	Extra
Host	char(60)		PRI		
Db	char(32)		PRI		
Select_priv	char(1)			N	
Insert_priv	char(1)			N	
Update_priv	char(1)			N	
Delete_priv	char(1)			N	

Create_priv	char(1)			N	
Drop_priv	char(1)			N	

Как видно из таблицы, здесь также можно задавать привилегии для доступа к базе данных. Они обычно редко используются без необходимости. Все привилегии доступа нужно задавать в таблице db для каждого пользователя, а в таблице host только перечислить имена хостов. Сервер читает все таблицы, проверяет имя пользователя, пароль, имя хоста, имя базы данных, привилегии. Если в таблице db привилегии select, insert установлены в 'Y', а в таблице host в 'N', то в итоге юзер все равно получит 'Y'. Чтобы не вносить путаницы, лучше назначать привилегии в таблице db.

Эти 3 таблицы являются основными. В новых версиях MySQL, начиная с 3.22 добавлены еще 2 таблицы - tables_priv и columns_priv, которые позволяют задать права доступа к определенной таблице в базе данных и даже к определенной колонке. Они работают подобно таблице db, только ссылаются на таблицы и колонки. Также, начиная с версии 3.22 можно использовать команду GRANT для предоставления доступа к базам данных, таблицам и колонкам таблиц, что избавляет от необходимости вручную модифицировать таблицы db, tables_priv и columns_priv. Команда GRANT будет подробно рассмотрена в следующих разделах.

Привилегии, предоставляемые MySQL

Таблица 6 - Привилегии пользователя⁴

Привилегия	Колонка	Где используется
select	Select_priv	таблицы
insert	Insert_priv	таблицы
Update	Update_priv	таблицы
delete	Delete_priv	таблицы
index	Index_priv	таблицы
alter	Alter_priv	таблицы
create	Create_priv	БД, таблицы, индексы
drop	Drop_priv	БД или таблицы
grant	Grant_priv	БД или таблицы
References	References_priv	БД или таблицы
reload	Reload_priv	администрирование сервера
Shutdown	Shutdown_priv	администрирование сервера
Process	Process_priv	администрирование сервера
file	File_priv	доступ к файлам на сервере

⁴ Даны для справки, для текущей версии сервера может быть существенно расширены

Основные утилиты MySQL.

В состав дистрибутива MySQL входят следующие утилиты:

- `mysqld`
- `mysql`
- `mysqladmin`
- `mysqlaccess`
- `mysqlshow`
- `mysqldump`
- `isamchk`

Утилиты **mysqld** и **mysql** были подробно рассмотрены ранее, поэтому возвращаться к ним не будем. Кратко рассмотрим остальные.

MySQLadmin

Утилита для администрирования сервера. Может использоваться администратором, а также некоторыми пользователями, которым предоставлены определенные привилегии, например – **Reload_priv**, **Shutdown_priv**, **Process_priv** и **File_priv**. Данная команда может использоваться для создания баз данных, изменения пароля пользователя (администратор может изменить пароль любому пользователю, а рядовой пользователь – только свой собственный), перезагрузки и остановки сервера, просмотра списка процессов, запущенных на сервере. MySQLadmin поддерживает следующие команды:

Таблица 7 - Опции команды MySQLadmin

Create [database_name]	Создает базу данных
Drop [database_name]	Удаляет базу данных и все таблицы в ней
Reload	Перезагружает сервер
Shutdown	Останавливает работу сервера MySQL
Processlist	Выводит список процессов на сервере
Status	Выводит сообщение о статусе сервера

Пример использования `mysqladmin` для изменения пароля:

```
mysqladmin -u your_name password your_pass
```

Следует заметить, что в случае использования `mysqladmin` для установки пароля, не требуется использование функции `password()`. MySQLadmin сам заботится о шифровании пароля.

MySQLaccess

Используется для проверки привилегий пользователя для доступа к конкретной базе данных. Общий синтаксис:

```
mysqlaccess [host] [user] [db] опции
```

Полезная утилита для проверки прав доступа пользователя, если он получает сообщение Access denied, при попытке соединиться с базой данных.

Опции:

Таблица 8 - Опции команды MySQLAccess

-?, --help	Справка
-u, --user=[username]	Имя пользователя
-p, --password=[password]	Пароль пользователя
-h, --host=[hostname]	Имя хоста для проверки прав доступа
-d, --db=[dbname]	Имя базы данных для проверки прав доступа
-U, --superuser=[susername]	Имя суперпользователя(root)
-P, --spassword=[spassword]	Пароль администратора
-b, --brief	Выводит краткие сведения о таблице

Mysqlshow

Используется, чтобы показать, с какими базами данных работает сервер, какие таблицы содержит каждая БД и какие колонки есть в каждой таблице. Синтаксис:

mysqlshow [опции] [database [table [field]]]

Mysqlshow может использовать следующие параметры:

Таблица 9 - Параметры команды Mysqlshow

-?, --help	Справка
-h, --host=[hostname]	Имя сервера
-k, --key	Показать ключи для таблицы
-p, --password=[password]	Пароль пользователя
-u, --user=[username]	Имя пользователя
-P, --port=[port]	Порт для связи
-V, --version	Вывести информацию о версии

Если ввести mysqlshow без аргументов, будут показаны все базы данных, если указать имя БД, будут показаны все таблицы в ней.

Команды

mysqlshow

mysqlshow mysql

Mysqldump

Программа mysqldump используется для создания дампа содержания базы данных MySQL. Она пишет инструкции SQL в стандартный вывод. Эти инструкции SQL могут быть переназначены в файл. Можно резервировать базу

данных MySQL, используя `mysqldump`, но при этом Вы должны убедиться, что в этот момент с базой данных не выполняется никаких других действий. А то `mysqldump` Вам такого нарезервирует...

Программа `mysqldump` поддерживает следующие параметры (Вы можете использовать короткую или подробную версию):

Таблица 10 - Опции команды `MySQLdump`

<code>-#, --debug=[options]</code>	Вывести в протокол отладочную информацию. В общем виде <code>'d:t:o, filename'</code> .
<code>-?, --help</code>	Справка.
<code>-c, --compleat-insert</code>	Генерируйте полные инструкции <code>insert</code> (не исключая значений, которые соответствуют значениям столбца по умолчанию).
<code>-h, --host=[hostname]</code>	Соединиться с сервером <code>hostname</code> .
<code>-d, --no-data</code>	Экспорт только схемы информации (исключая данные).
<code>-t, --no-create-info</code>	Экспорт только данных, исключая информацию для создания таблицы. Противоположность <code>-d</code> .
<code>-p, --password=[password]</code>	Пароль пользователя, для соединения с сервером MySQL. Обратите внимание, что не должно быть пробела между <code>-p</code> и паролем.
<code>-q, --quick</code>	Не буферизовать результаты запроса, дампы выдать непосредственно к <code>STDOUT</code> .
<code>-u, --user=[username]</code>	Имя пользователя. Если не задано, используется текущий логин.
<code>-v, --verbose</code>	Вывести подробную информацию относительно различных стадий выполнения <code>mysqldump</code> .
<code>-P, --port=[port]</code>	Порт для связи.
<code>-V, --version</code>	Информация о версии.

Вы можете направить вывод `mysqldump` в клиентскую программу MySQL, чтобы копировать базу данных. ПРИМЕЧАНИЕ: Вы должны убедиться, что база данных не изменяется в это время, иначе Вы получите противоречивую копию!

Для справки:

```
mysqldump -u root -p mysql user>mysql-1.sql
mysqldump -u root mysql>mysql-2.sql
```

Примечание флаг `-p` используется в случае, если пользователь наделен паролем.

После выполнения этой команды у нас появился файл `mysql-1.sql` и `mysql- 2.sql`. Загрузим их в текстовый редактор, чтобы поподробнее изучить, и, возможно, немного поправить.

Задание

1. Запустите сервер MySQL. Зарегистрируйте своего пользователя в консольном приложении, задайте ему права.
2. С помощью утилиты `Mysqlshow` выполните команду на просмотр структуры и состав таблиц базы `Mysql`. Приведите в отчете её схему.
3. С помощью утилиты `Mysqldump` получите полный дамп базы `Mysql` (данные и таблицы), а также отдельные дампы таблиц и данных.

Контрольные вопросы

1. Каким способом возможен запуск серверной части СУБД?
2. Что такое привилегия? Каково её предназначение?
3. Какие основные утилиты входят в состав СУБД, ка
кие функции они выполняют?