# Лабораторная работа № 3

**Тема:** Резервированное копирование и восстановление данных

Цель: Получить практические навыки для работы в среде СУБД MySQL.

**Сведения из теории.**

Резервирование данных важно проводить регулярно, при хорошо спланированном резервировании можно за короткое время восстановить данные почти при любой аварии. Общепринято осуществлять полное резервирование раз в неделю или через небольшие промежутки времени, а ежедневно производить резервирование данных, изменившихся с момента последнего полного резервирования. Это называется инкрементным резервированием.

**Резервное копирование**

Если копируются табличные файлы, следует остановить сервер.

BACKUP TABLE tbl\_name[,tbl\_name...] TO '/path/to/backup/directory'

Копирует в каталог резервного копирования тот минимум табличных файлов, который достаточен для восстановления таблицы, после сброса на диск всех изменений. Для таблиц MyISAM копирует файлы .frm (определений) и .MYD (данных). Индексные файлы могут быть реконструированы по этим двум. В процессе резервного копирования будет установлена блокировка чтения отдельно для каждой таблицы на время ее копирования.

**Восстановление данных**

RESTORE TABLE tbl\_name[,tbl\_name...] FROM '/path/to/backup/directory'

Восстанавливает таблицу(ы) из резервной копии, созданной с помощью BACKUP TABLE. Существующие таблицы не перезаписываются: при попытке восстановления поверх существующей таблицы будет выдана ошибка.

#### Пример: backup table vines,query,client,distributor,delivery to c:/mysql\_bak';

Восстановить таблицы командой restore table.

####  restore table vines,query,client,distributor,delivery from 'c:/mysql\_bak'

Таблицы нужно указывать через ',' (запятую), использовать конструкции типа <имя\_бд>.\* нельзя.

При использовании путей в командах backup и restore, нужно использовать обратный слеш (/), при использовании привычного слеша (\) будет выдаваться ошибка.

#### Пример:'c:/mysql\_bak' – правильно 'c:\mysql\_bak' - ошибка

mysqldump [OPTIONS] database [tables]

или mysqldump [OPTIONS] --databases [OPTIONS] DB1 [DB2 DB3...]

или mysqldump [OPTIONS] --all-databases [OPTIONS]

Данная утилита позволяет получить дамп (``моментальный снимок'') содержимого базы данных или совокупности баз для создания резервной копии. Дамп будет содержать набор команд SQL для создания и/или заполнения таблиц.

Если не указывать имена таблиц или использовать параметры --databases или --all-databases, то будет получен дамп базы данных в целом.

#### mysqldump --opt database > backup-file.sql

Можно, наоборот, прочитать этот файл на MySQL-сервере посредством команды:

#### mysql database < backup-file.sql или mysql -e "source /patch-to-backup/backup-file.sql" database

Вполне возможно получить дамп нескольких баз данных с помощью одной команды:

#### mysqldump --databases database1 [database2 ...] > my\_databases.sql

Если необходим дамп всех баз данных, можно использовать:

#### mysqldump --all-databases > all\_databases.sql

####  C:\mysql\bin>mysqldump vines > vine\_db.sql

#### mysql> drop table client,vines,query;

#### Удалены 3 таблицы. Теперь восстановим их.

#### mysql> source vine\_db.sql;

ВНИМАНИЕ: mysql должен быть запущен из того же каталого, что и mysqldump, то есть в данном случае из C:\mysql\bin, иначе mysql не найдет файл сценария.

Проверка поврежденных таблиц.

Повреждения в таблицах происходят вследствие событий, которые невозможно избежать.

CHECK TABLE tbl\_name[,tbl\_name...] [option [option...]]

option = QUICK | FAST | MEDIUM | EXTENDED | CHANGED

Если опция не указана, используется MEDIUM.

CHECK TABLE работает только на таблицах MyISAM и InnoDB. Проверяет таблицу(ы) на наличие ошибок. По каждой проверяемой таблице может быть выдано много строк информации. Последняя строка будет представлять Msg\_type status и, как правило, должна содержать OK. Если выдается что-либо отличное от OK и Not checked, то обычно следует провести ремонт таблицы. Table already up to date свидетельствует о том, что указанный для таблицы тип (TYPE) вернул информацию о том, что таблица не нуждается в проверке.

 REPAIR TABLE tbl\_name[,tbl\_name...] [QUICK] [EXTENDED] [USE\_FRM]

REPAIR TABLE ремонтирует таблицу, которая, возможно, повреждена.

 OPTIMIZE TABLE name

Обновляет таблицу, удаляя любое неиспользуемое пространство. Оптимизируемая таблица создается заново как отдельная временная таблица, а затем помещается на место текущей таблицы. Все табличные операции в это время происходят как обычно (вносимые изменения записываются во временную таблицу).

#### Пример: OPTIMIZE TABLE mytable

**Утилита myisamchk** проверяет/ремонтирует таблицы типа MyISAM (.MYI и .MYD), ее можно использовать для получения информации о таблицах рабочей базы данных, для их проверки и исправления или же оптимизации.

myisamchk [options] tbl\_name

Если опции не указаны, myisamchk просто проверяет таблицу.

tbl\_name - это таблица базы данных, которую нужно проверить/исправить.

#### myisamchk \*.MYI

Можно даже проверить все таблицы во всех базах данных, если задать шаблон вместе с путем к каталогу данных MySQL:

#### myisamchk /path/to/datadir/\*/\*.MYI

Проверочные опции для myisamchk

 -c или --check Проверить таблицы на ошибки.

 -e или --extend-check Проверить таблицу очень тщательно

 -F или --fast Проверять только таблицы, которые не были корректно закрыты.

 -C или --check-only-changed Проверять только таблицы, изменившиеся с момента последней проверки. и.т.д.

 Задание.

Проверить все таблицы на повреждение.

Обновить таблицы.

Создать резервные копии всех таблиц, используя все команды и утилиты

* 1. Через графическую среду выполнить BACKUP и RESTORE (следить за тем, чтобы таблицы были типа MyISAM).
	2. С помощью утилиты **mysqldump создать копию таблиц.**

Восстановить таблицы (2 таблицы)

В отчете представить все протоколы выполнения и сценарии дампов.