

Отчёт по лабораторной работе 6

**Установка и настройка системы управления базами данных
MariaDB**

Суннатилло Махмудов

Содержание

1	Цель работы	5
2	Теоретические сведения	6
2.1	Основные возможности MariaDB	6
2.2	Этапы взаимодействия с MariaDB	7
3	Выполнение лабораторной работы	8
3.1	Установка и первичная настройка MariaDB	8
3.2	Конфигурация кодировки символов	11
3.3	Создание базы данных и таблиц	13
3.4	Управление пользователями и правами	14
3.5	Проверка доступа к базе данных	14
3.6	Резервное копирование и восстановление	15
3.7	Сохранение конфигурации и резервных копий	16
4	Вывод	17
5	Контрольные вопросы	18
6	Список литературы	20

Список иллюстраций

3.1	Установка MariaDB	8
3.2	Запуск и автозагрузка MariaDB	9
3.3	Настройка безопасности MariaDB	9
3.4	Список команд MariaDB	10
3.5	Проверка системных БД	11
3.6	Статус MariaDB до изменения	11
3.7	Создание utf8.cnf	12
3.8	Статус MariaDB после изменения кодировки	12

Список таблиц

1 Цель работы

Приобретение практических навыков по установке и конфигурированию системы управления базами данных на примере программного обеспечения MariaDB.

2 Теоретические сведения

MariaDB — система управления базами данных (СУБД), форк MySQL с открытым исходным кодом. Применяется для хранения, обработки и управления данными в приложениях и информационных системах.

- **Сервер MariaDB** — основной компонент, выполняющий SQL-запросы и управляющий транзакциями.
- **Клиентские утилиты** — программы для взаимодействия с сервером (`mysql`, `mariadb`).
- **Хранилища данных (Storage Engines)** — модули для организации хранения (InnoDB, Aria, MyISAM и др.).
- **Конфигурационные файлы** — задают параметры работы сервера (`my.cnf`).

2.1 Основные возможности MariaDB

- Поддержка языка **SQL**;
- **Многопользовательский доступ** и система прав;
- **Репликация** для отказоустойчивости и масштабируемости;

- Работа с большими объемами данных;
- Совместимость с приложениями MySQL.

2.2 Этапы взаимодействия с MariaDB

1. **Установка** серверного ПО.
2. **Первичная настройка** параметров работы.
3. **Создание баз данных** и учетных записей пользователей.
4. **Конфигурирование безопасности.**
5. **Тестирование и эксплуатация.**

3 Выполнение лабораторной работы

3.1 Установка и первичная настройка MariaDB

1. На виртуальной машине **server** были установлены пакеты **mariadb** и **mariadb-server**.

```
Installing      : mariadb-client-utils-3:10.11.11-1.el10.x86_64      6/11
Installing      : mariadb-backup-3:10.11.11-1.el10.x86_64          7/11
Installing      : mariadb-gssapi-server-3:10.11.11-1.el10.x86_64    8/11
Installing      : mariadb-server-utils-3:10.11.11-1.el10.x86_64    9/11
Running scriptlet: mariadb-server-3:10.11.11-1.el10.x86_64        10/11
Installing      : mariadb-server-3:10.11.11-1.el10.x86_64        10/11
Running scriptlet: mariadb-server-3:10.11.11-1.el10.x86_64        10/11
Installing      : mariadb-3:10.11.11-1.el10.x86_64                11/11
Running scriptlet: mysql-selinux-1.0.14-1.el10.0.noarch            11/11
Running scriptlet: mariadb-3:10.11.11-1.el10.x86_64              11/11

Installed:
mariadb-3:10.11.11-1.el10.x86_64      mariadb-backup-3:10.11.11-1.el10.x86_64      mariadb-client-utils-3:10.11.11-1.el10.x86_64
mariadb-common-3:10.11.11-1.el10.noarch mariadb-errmsg-3:10.11.11-1.el10.noarch      mariadb-gssapi-server-3:10.11.11-1.el10.x86_64
mariadb-server-3:10.11.11-1.el10.x86_64 mariadb-server-utils-3:10.11.11-1.el10.x86_64 mysql-selinux-1.0.14-1.el10.0.noarch
perl-DBD-MariaDB-1.23-10.el10.x86_64    perl-Sys-Hostname-1.25-512.2.el10.0.x86_64

Complete!
[root@server.smahmudov.net ~]#
```

Рис. 3.1: Установка MariaDB

2. В каталоге **/etc/my.cnf.d/** были просмотрены конфигурационные файлы MariaDB. Основные из них:
 - **client.cnf** — параметры подключения клиента;
 - **mariadb-server.cnf** — настройки сервера (порты, пути хранения данных, лимиты ресурсов);
 - дополнительные конфиги (**auth_gssapi.cnf**, **enable_encryption.preset** и др.) — определяют модули и плагины.
3. Для запуска и включения MariaDB в автозагрузку использовались системные команды.


```
[root@server.smahmudov.net ~]# ls /etc/my.cnf.d/
auth_gssapi.cnf  enable_encryption.preset  mysql-clients.cnf  provider_lz4.cnf  provider_snappy.cnf
client.cnf      mariadb-server.cnf  provider_bzip2.cnf  provider_lzo.cnf  spider.cnf
[root@server.smahmudov.net ~]# systemctl start mariadb
[root@server.smahmudov.net ~]# systemctl enable mariadb
Created symlink '/etc/systemd/system/mysql.service' → '/usr/lib/systemd/system/mariadb.service'.
Created symlink '/etc/systemd/system/mysqld.service' → '/usr/lib/systemd/system/mariadb.service'.
Created symlink '/etc/systemd/system/multi-user.target.wants/mariadb.service' → '/usr/lib/systemd/system/mariadb.service'.
[root@server.smahmudov.net ~]#
[root@server.smahmudov.net ~]#
[root@server.smahmudov.net ~]# ss -tulpen | grep mysql
[root@server.smahmudov.net ~]# ss -tulpen | grep 3306
tcp  LISTEN  0      80          0.0.0.0:3306  0.0.0.0:*    users:((("mariadb",pid=14601,fd=18))
uid:27 ino:60134 sk:13 cgroup:/system.slice/mariadb.service <->
tcp  LISTEN  0      80          [::]:3306    [::]:*      users:((("mariadb",pid=14601,fd=19))
uid:27 ino:60135 sk:1f cgroup:/system.slice/mariadb.service v6only:1 <->
[root@server.smahmudov.net ~]#
```

Рис. 3.2: Запуск и автозагрузка MariaDB

4. Проверка показала, что служба **mariadb** прослушивает порт **3306**, что подтверждает успешный запуск.
5. Для повышения безопасности был запущен скрипт **mysql_secure_installation**:
 - установлен пароль для пользователя **root** базы данных;
 - отключён удалённый root-доступ;
 - удалена тестовая база данных и анонимные пользователи.

```
Normally, root should only be allowed to connect from 'localhost'. This
ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? [Y/n]
... Success!

By default, MariaDB comes with a database named 'test' that anyone can
access. This is also intended only for testing, and should be removed
before moving into a production environment.

Remove test database and access to it? [Y/n]
- Dropping test database...
... Success!
- Removing privileges on test database...
... Success!

Reloading the privilege tables will ensure that all changes made so far
will take effect immediately.

Reload privilege tables now? [Y/n]
... Success!

Cleaning up...

All done! If you've completed all of the above steps, your MariaDB
installation should now be secure.

Thanks for using MariaDB!
[root@server.smahmudov.net ~]# mysql -u root -p
Enter password: █
```

Рис. 3.3: Настройка безопасности MariaDB

6. Был выполнен вход в систему управления БД под пользователем **root**.

7. Отображён список доступных клиентских команд MariaDB.

```
List of all client commands:
Note that all text commands must be first on line and end with ';'
?          (\?) Synonym for 'help'.
charset    (\C) Switch to another charset. Might be needed for processing binlog with multi-byte charsets.
clear       (\c) Clear the current input statement.
connect     (\r) Reconnect to the server. Optional arguments are db and host.
delimiter  (\d) Set statement delimiter.
edit        (\e) Edit command with $EDITOR.
ego         (\G) Send command to MariaDB server, display result vertically.
exit        (\q) Exit mysql. Same as quit.
go          (\g) Send command to MariaDB server.
help        (\h) Display this help.
nopager     (\n) Disable pager, print to stdout.
notee       (\t) Don't write into outfile.
nowarning   (\w) Don't show warnings after every statement.
pager       (\P) Set PAGER [to_pager]. Print the query results via PAGER.
print       (\p) Print current command.
prompt      (\R) Change your mysql prompt.
quit        (\q) Quit mysql.
rehash      (\#) Rebuild completion hash.
sandbox     (\-) Disallow commands that access the file system (except \P without an argument and \e).
source      (\.) Execute an SQL script file. Takes a file name as an argument.
status      (\s) Get status information from the server.
system      (\!) Execute a system shell command.
tee         (\T) Set outfile [to_outfile]. Append everything into given outfile.
use         (\u) Use another database. Takes database name as argument.
warnings     (\W) Show warnings after every statement.

For server side help, type 'help contents'

MariaDB [(none)]>
```

Рис. 3.4: Список команд MariaDB

8. Для проверки системных баз данных по умолчанию был выполнен запрос **SHOW DATABASES**.

Результат: **information_schema, mysql, performance_schema, sys**.

For server side help, type 'help contents'

```
MariaDB [(none)]> SHOW DATABASES;
```

```
+-----+  
| Database |  
+-----+  
| information_schema |  
| mysql |  
| performance_schema |  
| sys |  
+-----+
```

4 rows in set (0.000 sec)

```
MariaDB [(none)]> █
```

Рис. 3.5: Проверка системных БД

3.2 Конфигурация кодировки символов

1. До изменения параметров команда **status** показала текущую конфигурацию MariaDB, включая используемую кодировку.

```
MariaDB [(none)]> status  
-----  
mysql Ver 15.1 Distrib 10.11.11-MariaDB, for Linux (x86_64) using EditLine wrapper  
  
Connection id:          14  
Current database:         
Current user:           root@localhost  
SSL:                    Not in use  
Current pager:          stdout  
Using outfile:          ''  
Using delimiter:        ;  
Server:                 MariaDB  
Server version:         10.11.11-MariaDB MariaDB Server  
Protocol version:       10  
Connection:             Localhost via UNIX socket  
Server characterset:     latin1  
Db characterset:         latin1  
Client characterset:     utf8mb3  
Conn. characterset:      utf8mb3  
UNIX socket:            /var/lib/mysql/mysql.sock  
Uptime:                 4 min 41 sec  
  
Threads: 1 Questions: 27 Slow queries: 0 Opens: 20 Open tables: 13 Queries per second avg: 0.096  
-----  
MariaDB [(none)]> █
```

Рис. 3.6: Статус MariaDB до изменения

2. В каталоге **/etc/my.cnf.d/** был создан файл **utf8.cnf** со следующими настрой-

ками:

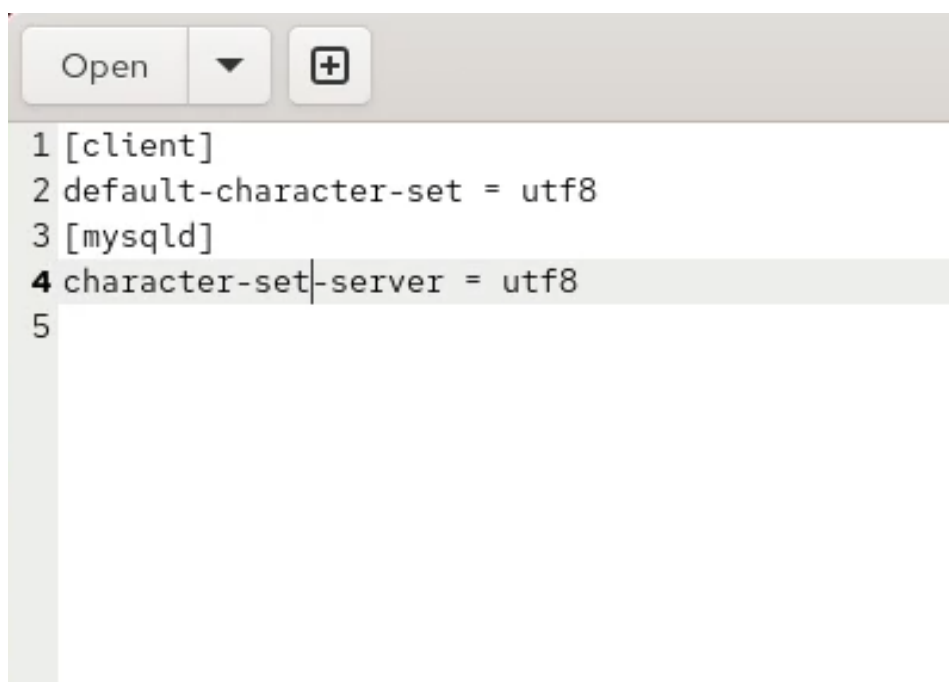


Рис. 3.7: Создание utf8.cnf

3. После перезапуска MariaDB команда **status** показала изменения: клиент, сервер и базы данных теперь используют кодировку **utf8**.

```
MariaDB [(none)]> status
-----
mysql Ver 15.1 Distrib 10.11.11-MariaDB, for Linux (x86_64) using EditLine wrapper

Connection id:          3
Current database:
Current user:           root@localhost
SSL:                   Not in use
Current pager:          stdout
Using outfile:          ''
Using delimiter:        ;
Server:                 MariaDB
Server version:         10.11.11-MariaDB MariaDB Server
Protocol version:       10
Connection:             Localhost via UNIX socket
Server characterset:    utf8mb3
Db characterset:        utf8mb3
Client characterset:    utf8mb3
Conn. characterset:     utf8mb3
UNIX socket:            /var/lib/mysql/mysql.sock
Uptime:                 11 sec

Threads: 1  Questions: 4  Slow queries: 0  Opens: 17  Open tables: 10  Queries per second avg: 0.363
-----
MariaDB [(none)]>
```

Рис. 3.8: Статус MariaDB после изменения кодировки

3.3 Создание базы данных и таблиц

1. В системе управления БД была создана база данных **addressbook** с кодировкой **utf8** и произведён переход к ней.

```
MariaDB [(none)]> USE addressbook;
Database changed
MariaDB [addressbook]> SHOW TABLES;
Empty set (0.001 sec)

MariaDB [addressbook]> CREATE TABLE city(name VARCHAR(40), city VARCHAR(40));
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to
city VARCHAR(40))' at line 1
MariaDB [addressbook]> CREATE TABLE city(name VARCHAR(40), city VARCHAR(40));
Query OK, 0 rows affected (0.004 sec)

MariaDB [addressbook]> INSERT INTO city(name,city) VALUES ('Ivanov', 'Moscow');
Query OK, 1 row affected (0.001 sec)

MariaDB [addressbook]> INSERT INTO city(name,city) VALUES ('Petrov', 'Sochi');
Query OK, 1 row affected (0.001 sec)

MariaDB [addressbook]> INSERT INTO city(name,city) VALUES ('Sidorov', 'Dubna');
Query OK, 1 row affected (0.001 sec)

MariaDB [addressbook]> SELECT * FROM city;
+-----+-----+
| name  | city  |
+-----+-----+
| Ivanov | Moscow |
| Petrov | Sochi  |
| Sidorov | Dubna  |
+-----+-----+
3 rows in set (0.000 sec)

MariaDB [addressbook]> █
```

2. В базе данных были просмотрены существующие таблицы (результат — пусто) и создана таблица **city** с полями *name* и *city*.
3. Таблица была заполнена данными:
 - Иванов, Москва;
 - Петров, Сочи;
 - Сидоров, Дубна.
4. Запрос **SELECT * FROM city** вывел все строки таблицы.
Результат: три записи с фамилиями и городами.

3.4 Управление пользователями и правами

1. Был создан пользователь **smahmudov** с паролем 123456.
2. Пользователю предоставлены права на просмотр, добавление, обновление и удаление данных в базе **addressbook**.
3. Привилегии были обновлены.

```
MariaDB [addressbook]> CREATE USER smahmudov@%' IDENTIFIED BY '123456';
Query OK, 0 rows affected (0.001 sec)

MariaDB [addressbook]> GRANT SELECT,INSERT,UPDATE,DELETE ON addressbook.* TO smahmudov@%';
Query OK, 0 rows affected (0.002 sec)

MariaDB [addressbook]> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.000 sec)

MariaDB [addressbook]> DESCRIBE city;
+-----+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| name  | varchar(40)   | YES  |     | NULL    |       |
| city  | varchar(40)   | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.001 sec)

MariaDB [addressbook]>
```

4. Команда **DESCRIBE city** показала структуру таблицы — два текстовых поля *name* и *city* длиной до 40 символов.

3.5 Проверка доступа к базе данных

1. Список баз данных был просмотрен под пользователем **root**, где присутствует **addressbook**.
2. Были отображены таблицы базы **addressbook** под пользователями **root** и **smahmudov** — в обоих случаях доступна таблица **city**.

```

[root@server.smahmudov.net my.cnf.d]# mysqlshow -u root -p
Enter password:
+-----+
| Databases |
+-----+
| addressbook |
| information_schema |
| mysql |
| performance_schema |
| sys |
+-----+
[root@server.smahmudov.net my.cnf.d]# mysqlshow -u root -p addressbook
Enter password:
Database: addressbook
+-----+
| Tables |
+-----+
| city |
+-----+
[root@server.smahmudov.net my.cnf.d]# mysqlshow -u smahmudov -p addressbook
Enter password:
Database: addressbook
+-----+
| Tables |
+-----+
| city |
+-----+
[root@server.smahmudov.net my.cnf.d]#

```

3.6 Резервное копирование и восстановление

1. На виртуальной машине **server** был создан каталог для резервных копий:
/var/backup.
2. Выполнено резервное копирование базы **addressbook** в файл **addressbook.sql** и его сжатая версия **addressbook.sql.gz.**
3. Также была создана копия с автоматическим добавлением даты в имя файла.

```
[root@server.smahmudov.net my.cnf.d]#
[root@server.smahmudov.net my.cnf.d]# mkdir -p /var/backup
[root@server.smahmudov.net my.cnf.d]# mysqldump -u root -p addressbook > /var/backup/addressbook.sql
Enter password:
[root@server.smahmudov.net my.cnf.d]# mysqldump -u root -p addressbook | gzip > /var/backup/addressbook.sql.gz
Enter password:
[root@server.smahmudov.net my.cnf.d]# mysqldump -u root -p addressbook | gzip > $(/var/backup/addressbook.%Y%m%d.%H%M%S.sql.gz)
-bash: /var/backup/addressbook.%Y%m%d.%H%M%S.sql.gz: No such file or directory
-bash: $(/var/backup/addressbook.%Y%m%d.%H%M%S.sql.gz): ambiguous redirect
Enter password:
mysqldump: Got errno 32 on write
[root@server.smahmudov.net my.cnf.d]#
[root@server.smahmudov.net my.cnf.d]# mysqldump -u root -p addressbook | gzip > $(date+var/backup/addressbook.%Y%m%d.%H%M%S.sql.gz)
-bash: date+var/backup/addressbook.%Y%m%d.%H%M%S.sql.gz: No such file or directory
-bash: $(date+var/backup/addressbook.%Y%m%d.%H%M%S.sql.gz): ambiguous redirect
Enter password:
mysqldump: Got errno 32 on write
[root@server.smahmudov.net my.cnf.d]# mysqldump -u root -p addressbook | gzip > $(date +var/backup/addressbook.%Y%m%d.%H%M%S.sql.gz)
Enter password:
[root@server.smahmudov.net my.cnf.d]# ls /var/backup/
addressbook.20250926.123550.sql.gz  addressbook.sql  addressbook.sql.gz
[root@server.smahmudov.net my.cnf.d]# mysql -u root -p addressbook < /var/backup/addressbook.sql
Enter password:
[root@server.smahmudov.net my.cnf.d]#
[root@server.smahmudov.net my.cnf.d]# zcat /var/backup/addressbook.sql.gz | mysql -u root -p addressbook
Enter password:
[root@server.smahmudov.net my.cnf.d]#
[root@server.smahmudov.net my.cnf.d]#
```

4. Выполнено восстановление базы данных из обычной и сжатой резервной копии.

3.7 Сохранение конфигурации и резервных копий

1. В каталоге **/vagrant/provision/server/** был создан подкаталог **mysql**, содержащий:
 - конфигурацию MariaDB из **/etc/my.cnf.d/utf8.cnf**;
 - резервные копии базы из каталога **/var/backup/**.
2. Для автоматизации операций был подготовлен исполняемый файл

mysql.sh.

```
[root@server.smahmudov.net my.cnf.d]#
[root@server.smahmudov.net my.cnf.d]# cd /vagrant/provision/server/
[root@server.smahmudov.net server]# mkdir -p /vagrant/provision/server/mysql/etc/my.cnf.d
[root@server.smahmudov.net server]# mkdir -p /vagrant/provision/server/mysql/var/backup
[root@server.smahmudov.net server]# cp -R /etc/my.cnf.d/utf8.cnf > /vagrant/provision/server/mysql/etc/my.cnf.d/
-bash: /vagrant/provision/server/mysql/etc/my.cnf.d/: Is a directory
[root@server.smahmudov.net server]# cp -R /etc/my.cnf.d/utf8.cnf /vagrant/provision/server/mysql/etc/my.cnf.d/
[root@server.smahmudov.net server]# cp -R /var/backup/* /vagrant/provision/server/mysql/var/backup/
[root@server.smahmudov.net server]# touch mysql.sh
[root@server.smahmudov.net server]#
```


4 Вывод

В ходе лабораторной работы была установлена и настроена СУБД **MariaDB**. Были выполнены операции по установке и запуску сервиса, настройке параметров безопасности, проверке доступных системных баз данных. Создана база данных **addressbook**, в ней — таблица **city** с добавлением тестовых записей. Реализовано создание отдельного пользователя с выдачей ему прав доступа к базе. Проверены команды для отображения баз данных, таблиц и структуры таблиц. Настроена кодировка **utf8** на уровне сервера и клиента. Дополнительно выполнено резервное копирование базы данных с использованием утилиты **mysqldump**, восстановление базы из резервных копий, а также сохранение конфигурационных файлов в каталог для автоматизированного развертывания.

5 Контрольные вопросы

1. **Какая команда отвечает за настройки безопасности в MariaDB?**

Команда **mysql_secure_installation**.

2. **Как настроить MariaDB для доступа через сеть?**

В конфигурационном файле **/etc/my.cnf.d/mariadb-server.cnf** в секции **[mysqld]** необходимо изменить параметр **bind-address**, указав IP-адрес сервера или **0.0.0.0**. Также следует открыть порт **3306** в файерволе.

3. **Какая команда позволяет получить обзор доступных баз данных после входа в среду оболочки MariaDB?**

Команда **SHOW DATABASES;**

4. **Какая команда позволяет узнать, какие таблицы доступны в базе данных?**

Команда **SHOW TABLES;**

5. **Какая команда позволяет узнать, какие поля доступны в таблице?**

Команда **DESCRIBE имя_таблицы;** или сокращённо **DESC имя_таблицы;**

6. **Какая команда позволяет узнать, какие записи доступны в таблице?**

Команда **SELECT * FROM имя_таблицы;**

7. **Как удалить запись из таблицы?**

С помощью команды **DELETE FROM имя_таблицы WHERE условие;**

8. **Где расположены файлы конфигурации MariaDB? Что можно настроить с их помощью?**

Основные конфигурационные файлы находятся в каталогах **/etc/my.cnf** и **/etc/my.cnf.d/**.

С их помощью можно настроить параметры сервера: пути хранения данных, сетевые параметры, кодировки, плагины, лимиты ресурсов, права доступа.

9. Где располагаются файлы с базами данных MariaDB?

По умолчанию базы данных хранятся в каталоге **/var/lib/mysql/**.

10. Как сделать резервную копию базы данных и затем её восстановить?

- Резервная копия: **mysqldump -u root -p имя_базы > файл.sql**
- Сжатая копия: **mysqldump -u root -p имя_базы | gzip > файл.sql.gz**
- Восстановление: **mysql -u root -p имя_базы < файл.sql**
- Восстановление из сжатой копии: **zcat файл.sql.gz | mysql -u root -p имя_базы**

6 Список литературы

1. MariaDB Foundation. — URL: <https://mariadb.org> (дата обр. 13.09.2021).
2. Документация по MariaDB. — URL: <https://mariadb.com/kb/ru/5306/>.
3. Основы языка SQL. — URL: <http://citforum.ru/programming/32less/les44.shtml> (дата обр. 13.09.2021).