Часть 1.

1. Создано 4 виртуальные машины.
2.

```
$: sudo apt install ntp
```

/etc/ntpsec/ntp.conf

```
driftfile /var/lib/ntpsec/ntp.drift
leapfile /usr/share/zoneinfo/leap-seconds.list

pool 0.ubuntu.pool.ntp.org iburst
pool 1.ubuntu.pool.ntp.org iburst
pool 2.ubuntu.pool.ntp.org iburst
pool 3.ubuntu.pool.ntp.org iburst

#server ntp.ubuntu.com
server 127.127.1.0

#restrict default kod nomodify

restrict 127.0.0.1
restrict ::1

restrict 192.168.141.0 mask 255.255.255.0 nomodify notrap nopeer
```

```
$: sudo systemctl stop ntp
$: sudo date -s "2025-02-02 12:00:00"
$: sudo systemctl restart ntp
$: ntpq -p
```

3. [1]

/etc/ntpsec/ntp.conf

```
server 127.127.1.0
```

[2]

/etc/bind/named.conf.local

```
zone "myzone.local"{
        type master;
        file "/etc/bind/db.myzone.local";
};
```

/etc/bind/db.myzone.local

```
$TTL 86400;

@ IN SOA node-2.myzone.local. root.myzone.local. (
        2022103007 ; Serial
        600 ; Refresh
        3600 ; Retry
        1w ; Expirt
        360) ; Negative cache TTL

@ IN NS node-2.myzone.local.
@ IN A 192.168.141.11
node-1 IN A 192.168.141.10
node-2 IN A 192.168.141.11
node-3 IN A 192.168.141.13
node-4 IN A 192.168.141.14
```

/etc/systemd/resolv.conf

```
[Resolve]
# Some examples of DNS server
# Cloudflare: 1.1.1.1#cloudfl
# Google:      8.8.8.8#dns.goo
# Quad9:       9.9.9.9#dns.qua
DNS=192.168.141.11
#FallbackDNS=
Domains=myzone.local
```

/etc/nsswitch.conf

```
hosts:          files mdns4_minimal dns
```

/etc/ntpsec/ntp.conf

```
driftfile /var/lib/ntpsec/ntp.drift
leapfile /usr/share/zoneinfo/leap-seconds.list
logfile /var/log/ntp.log

#server ntp.ubuntu.com
server 192.168.141.10 iburst prefer

#restrict default kod nomodify

restrict 127.0.0.1
restrict ::1

restrict 192.168.141.0 mask 255.255.255.0 nomodify notrap nopeer
```

4. [3]

/etc/systemd/resolv.conf

```
[Resolve]
# Some examples of DNS
# Cloudflare: 1.1.1.1#
# Google:      8.8.8.8#
# Quad9:       9.9.9.9#
DNS=192.168.141.11
#FallbackDNS=
Domains=myzone.local
```

/etc/nsswitch.conf

```
hosts:          files mdns4_minimal dns
```

/etc/ntpsec/ntp.conf

```
driftfile /var/lib/ntpsec/ntp.drift
leapfile /usr/share/zoneinfo/leap-seconds.list

pool node-1.myzone.local iburst
pool node-2.myzone.local iburst

#server ntp.ubuntu.com

restrict 127.0.0.1
restrict ::1
```

[4]

/etc/systemd/resolv.conf

```
[Resolve]
# Some examples of DNS s
# Cloudflare: 1.1.1.1#c
# Google:     8.8.8.8#d
# Quad9:      9.9.9.9#d
DNS=192.168.141.11
#FallbackDNS=
Domains=myzone.local
```

/etc/nsswitch.conf

```
hosts:          files mdns4_minimal dns
```

/etc/ntpsec/ntp.conf

```
driftfile /var/lib/ntpsec/ntp.drift
leapfile /usr/share/zoneinfo/leap-seconds.list

pool node-1.myzone.local iburst
pool node-2.myzone.local iburst

#server ntp.ubuntu.com

restrict 127.0.0.1
restrict ::1
```

Часть 2.

1.

$: sudo apt install mariadb-server mariadb-client

2.

$: sudo mariadb
    ➢ Create database library
    ➢ Use library

```
MariaDB [library]> create table if not exists books (b_id INT PRIMARY KEY AUTO_I
NCREMENT, name VARCHAR(100) NOT NULL, author VARCHAR(100) NOT NULL, price INT NO
T NULL, genre VARCHAR(100) NOT NULL);
Query OK, 0 rows affected (0,034 sec)
```

```
MariaDB [library]> create table if not exists orders (o_id INT PRIMARY KEY AUTO_
INCREMENT, b_id INT REFERENCES books(b_id), address VARCHAR(100) NOT NULL, statu
s BOOLEAN);
Query OK, 0 rows affected (0,019 sec)
```

```
MariaDB [library]> insert into books values(1,'name1','author1',200,'romance');
Query OK, 1 row affected (0,022 sec)
```

```
MariaDB [library]> insert into books values(2,'name2','author2',400,'comedy');
Query OK, 1 row affected (0,006 sec)

MariaDB [library]> insert into books values(3,'name3','author3',999,'drama');
Query OK, 1 row affected (0,006 sec)

MariaDB [library]> select * from books;
+------+-------+---------+-------+---------+
| b_id | name  | author  | price | genre   |
+------+-------+---------+-------+---------+
|    1 | name1 | author1 |   200 | romance |
|    2 | name2 | author2 |   400 | comedy  |
|    3 | name3 | author3 |   999 | drama   |
+------+-------+---------+-------+---------+
3 rows in set (0,000 sec)
```

```
MariaDB [library]> insert into orders values(1,1,'address1',false);
Query OK, 1 row affected (0,020 sec)
```

```
MariaDB [library]> insert into orders values(2,2,'address1',false);
Query OK, 1 row affected (0,008 sec)

MariaDB [library]> insert into orders values(3,3,'address1',false);
Query OK, 1 row affected (0,006 sec)

MariaDB [library]> insert into orders values(4,2,'address2',true);
Query OK, 1 row affected (0,018 sec)

MariaDB [library]> insert into orders values(5,3,'address3',false);
Query OK, 1 row affected (0,017 sec)

MariaDB [library]> insert into orders values(6,1,'address4',true);
Query OK, 1 row affected (0,007 sec)

MariaDB [library]> select * from orders;
+------+------+----------+--------+
| o_id | b_id | address  | status |
+------+------+----------+--------+
|    1 |    1 | address1 |      0 |
|    2 |    2 | address1 |      0 |
|    3 |    3 | address1 |      0 |
|    4 |    2 | address2 |      1 |
|    5 |    3 | address3 |      0 |
|    6 |    1 | address4 |      1 |
+------+------+----------+--------+
6 rows in set (0,000 sec)
```

3. [3]
    /etc/mysql/mariadb.conf.d/50-server.cnf

```
bind-address            = 192.168.141.13
server-id               = 1
replicate-do-db         = library
binlog-do-db            = library


#
# * Fine Tuning
#


#key_buffer_size         = 128M
#max_allowed_packet      = 1G
#thread_stack            = 192K
#thread_cache_size       = 8
# This replaces the startup script and checks MyISAM tables if needed
# the first time they are touched
#myisam_recover_options = BACKUP
#max_connections         = 100
#table_cache             = 64



#
# * Logging and Replication
#


# Note: The configured log file or its directory need to be created
# and be writable by the mysql user, e.g.:
# $ sudo mkdir -m 2750 /var/log/mysql
# $ sudo chown mysql /var/log/mysql


# Both location gets rotated by the cronjob.
# Be aware that this log type is a performance killer.
# Recommend only changing this at runtime for short testing periods if
general_log_file        = /var/log/mysql/mysql.log
general_log             = 1


# When running under systemd, error logging goes via stdout/stderr to j
# and when running legacy init error logging goes to syslog due to
# /etc/mysql/conf.d/mariadb.conf.d/50-mysqld_safe.cnf
# Enable this if you want to have error logging into a separate file
log_error = /var/log/mysql/error.log
# Enable the slow query log to see queries with especially long duratio
#log_slow_query_file     = /var/log/mysql/mariadb-slow.log
#log_slow_query_time     = 10
#log_slow_verbosity      = query_plan,explain
#log-queries-not-using-indexes
#log_slow_min_examined_row_limit = 1000


# The following can be used as easy to replay backup logs or for replic
# note: if you are setting up a replica, see README.Debian about other
#       settings you may need to change.
#server-id               = 1
log_bin                 = /var/log/mysql/mysql-bin.log
expire_logs_days        = 10
```

```
sudo mkdir /var/log/mysql
```

```
sudo chown mysql:mysql /var/log/mysql
```

```
user@user:~$ mysqldump -u root -p --databases library > /tmp/mydb_dump.sql
```

```
user@user:~$ scp /tmp/mydb_dump.sql user@192.168.141.14:/tmp
```

```
MariaDB [(none)]> grant replication slave on *.* to "user"@"192.168.141.14" id
entified by "password";
```

```
MariaDB [(none)]> show master status\G;
*************************** 1. row ***************************
            File: mysql-bin.000004
        Position: 532
    Binlog_Do_DB: library
Binlog_Ignore_DB:
1 row in set (0,000 sec)
```

```
MariaDB [(none)]> change master to master_host = "192.168.141.14", master_port
=3306, master_user = "user", master_password = "password", master_log_file = "
mysql-bin.000004", master_log_pos = 3105;
```

```
MariaDB [(none)]> start slave;
```

```
MariaDB [(none)]> show slave status\G;
*************************** 1. row ***************************
               Slave_IO_State: Waiting for master to send event
                  Master_Host: 192.168.141.14
                  Master_User: user
                  Master_Port: 3306
                Connect_Retry: 60
              Master_Log_File: mysql-bin.000004
          Read_Master_Log_Pos: 3295
               Relay_Log_File: mysqld-relay-bin.000002
                Relay_Log_Pos: 745
        Relay_Master_Log_File: mysql-bin.000004
             Slave_IO_Running: Yes
            Slave_SQL_Running: Yes
              Replicate_Do_DB: library
           Replicate Ignore DB:
```

[4]

/etc/mysql/mariadb.conf.d/50-server.cnf

```
bind-address            = 192.168.141.14
server-id               = 2
replicate-do-db         = library
binlog-do-db            = library


#
# * Fine Tuning
#

#key_buffer_size         = 128M
#max_allowed_packet      = 1G
#thread_stack            = 192K
#thread_cache_size       = 8
# This replaces the startup script and checks MyISAM tables if need
# the first time they are touched
#myisam_recover_options = BACKUP
#max_connections         = 100
#table_cache             = 64


#
# * Logging and Replication
#


# Note: The configured log file or its directory need to be created
# and be writable by the mysql user, e.g.:
# $ sudo mkdir -m 2750 /var/log/mysql
# $ sudo chown mysql /var/log/mysql

# Both location gets rotated by the cronjob.
# Be aware that this log type is a performance killer.
# Recommend only changing this at runtime for short testing periods
general_log_file        = /var/log/mysql/mysql.log
general_log             = 1

# When running under systemd, error logging goes via stdout/stderr
# and when running legacy init error logging goes to syslog due to
# /etc/mysql/conf.d/mariadb.conf.d/50-mysqld_safe.cnf
# Enable this if you want to have error logging into a separate fil
log_error = /var/log/mysql/error.log
# Enable the slow query log to see queries with especially long dur
#log_slow_query_file     = /var/log/mysql/mariadb-slow.log
#log_slow_query_time     = 10
#log_slow_verbosity      = query_plan,explain
#log-queries-not-using-indexes
#log_slow_min_examined_row_limit = 1000


# The following can be used as easy to replay backup logs or for re
# note: if you are setting up a replica, see README.Debian about ot
#       settings you may need to change.
#server-id               = 1
log_bin                 = /var/log/mysql/mysql-bin.log
expire_logs_days        = 10
#max_binlog_size         = 100M
```

```
sudo mkdir /var/log/mysql
```
```
 sudo chown mysql:mysql /var/log/mysql
```

```
user@user:~$ sudo mariadb < /tmp/mydb_dump.sql
```

```
MariaDB [(none)]> grant replication slave on *.* to "user"@"192.168.141.13" identif
ied by "password";
```

```
MariaDB [(none)]> show master status\G;
*************************** 1. row ***************************
            File: mysql-bin.000004
        Position: 3295
    Binlog_Do_DB: library
Binlog_Ignore_DB:
1 row in set (0,017 sec)
```

```
MariaDB [(none)]> change master to master_host = "192.168.141.13", master_port=3306
, master_user = "user", master_password = "password", master_log_file = "mysql-bin.
000004", master_log_pos = 342;

MariaDB [(none)]> start slave;

MariaDB [(none)]> show slave status\G;
*************************** 1. row ***************************
               Slave_IO_State: Waiting for master to send event
                  Master_Host: 192.168.141.13
                  Master_User: user
                  Master_Port: 3306
                Connect_Retry: 60
              Master_Log_File: mysql-bin.000004
          Read_Master_Log_Pos: 532
               Relay_Log_File: mysqld-relay-bin.000002
                Relay_Log_Pos: 745
        Relay_Master_Log_File: mysql-bin.000004
             Slave_IO_Running: Yes
            Slave_SQL_Running: Yes
              Replicate_Do_DB: library
          Replicate_Ignore_DB:
```