

Домашняя работа №6:

1. УСТАНОВИТЬ Nginx :

`sudo apt install nginx`

Убеждаемся, что сервер запущен и работает:

Таблица сокетов:

`sudo ss -ntlp`

```
fedorsannikov@VMLinux:/etc/nginx$ sudo ss -ntlp
State      Recv-Q    Send-Q    Local Address:Port    Peer Address:Port    Process
LISTEN     0          128       127.0.0.1:631         0.0.0.0:*             users:((("cupsd",pid=745,fd=7))
LISTEN     0          128       0.0.0.0:22           0.0.0.0:*             users:((("sshd",pid=700,fd=3))
LISTEN     0          511       0.0.0.0:80           0.0.0.0:*             users:((("nginx",pid=12174,fd=6),("nginx",pid=12173,fd=6),("nginx",pid=12171,fd=6))
LISTEN     0          4096     127.0.0.1:53         0.0.0.0:*             users:((("systemd-resolve",pid=307,fd=14))
LISTEN     0          128       [::]:22              [::]:*                users:((("sshd",pid=700,fd=4))
LISTEN     0          511       [::]:80              [::]:*                users:((("nginx",pid=12174,fd=7),("nginx",pid=12173,fd=7),("nginx",pid=12171,fd=7))
LISTEN     0          128       [::]:631             [::]:*                users:((("cupsd",pid=745,fd=6))
```

Таблица процессов:

`sudo ps afx`

```
fedorsannikov@VMLinux:/etc/nginx$ sudo ps afx
12171 ?        S      0:00 nginx: master process /usr/sbin/nginx -g daemon on; master_process on;
12173 ?        S      0:00 \_ nginx: worker process
12174 ?        S      0:00 \_ nginx: worker process
```

В терминале:

`curl localhost` или `curl localhost:80`

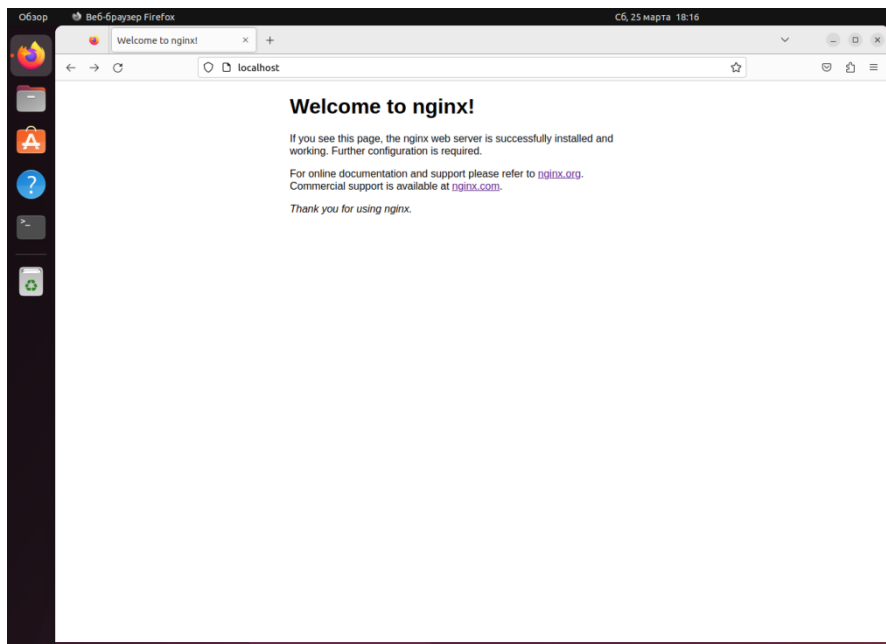
```
fedorsannikov@VMLinux:/etc/nginx$ curl localhost
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
  body {
    width: 35em;
    margin: 0 auto;
    font-family: Tahoma, Verdana, Arial, sans-serif;
  }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>

<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
fedorsannikov@VMLinux:/etc/nginx$
```

В браузере на виртуальной машине:

localhost:80 или localhost



Примечание:

Расположение “общих” настроек nginx:

/etc/nginx/

файл:

nano nginx.conf

Расположение настроек (символической ссылки на настройки) для конкретного сервера по умолчанию:

cd sites-enabled

Переходим по символической ссылке:

default -> /etc/nginx/sites-available/default

sudo nano default

Расположение загружаемой странички по умолчанию:

```
cd /var/www/html
```

файл:

```
index.nginx-debian.html
```

(P.S.: На самом деле там много файлов)

После внесения изменений в конфигурацию проверим синтаксис:

```
sudo nginx -t
```

Если всё в порядке, применить заданную конфигурацию:

```
sudo systemctl reload nginx
```

Эта команда применяет конфигурацию без разрыва соединений и потери запросов от пользователей.

2. Настроить Nginx на работу с PHP-FPM:

PHP-FPM — программный пакет, позволяющий выполнить обработку скриптов, написанных на языке [PHP](#). Включен в состав PHP с версии 5.3.3, для более ранних версий необходима установка отдельно. Является альтернативой FastCGI — протоколу взаимодействия [веб-сервера](#) с программами. FPM расшифровывается как Fastcgi Process Manager.

Часто применяется для настройки вместе с веб-серверами [NGINX](#) или Lighttpd. Последний обрабатывает статические данные, а обработку скриптов отдает PHP-FPM. Такая реализация работает быстрее, по сравнению с, также распространенной, NGINX + [Apache](#) + модуль mod_php.

PHP-FPM запускается как отдельный процесс. По умолчанию, взаимодействует с веб-сервером через порт 9000 или сокетный файл. Есть поддержка всех известных операционных систем, а именно — [FreeBSD](#), Linux (например, [CentOS](#) или [Ubuntu](#)), [Windows](#).

Установка:

```
apt install php8.1-fpm
```

Проверяем, установился или нет:

Таблица процессов:

```
sudo ps afx
```

```
 22432 ?        Ss      0:00 php-fpm: master process (/etc/php/8.1/fpm/php-fpm.conf)
 22433 ?        S        0:00 \_ php-fpm: pool www
 22434 ?        S        0:00 \_ php-fpm: pool www
root@VMLinux:~#
```

Настраиваем NGINX для работы с PHP-FPM добавляя следующий блок в файл настроек:

Расположение настроек:

```
cd /etc/nginx/sites-enabled
```

Переходим по символической ссылке:

```
default -> /etc/nginx/sites-available/default
```

```
sudo nano default
```

Добавляем следующий блок, отвечающий за настройку работы NGINX с PHP-FPM через UNIX-сокеты:

```
location ~ /\.php$ {  
    include snippets/fastcgi-php.conf;  
    root /var/www/html;  
    fastcgi_pass unix:/run/php/php8.1-fpm.sock;  
}
```

Проверяем синтаксис следующей командой:

```
sudo nginx -t
```

Применяем конфигурацию:

```
sudo systemctl reload nginx
```

Создаем info.php (для проверки) в корне основного сайта по умолчанию /var/www/html со следующим содержанием:

```
<?php  
phpinfo();  
?>
```

Настраиваем/проверяем настройки по умолчанию PHP-FPM для прослушивания сокета UNIX:

Примечание:

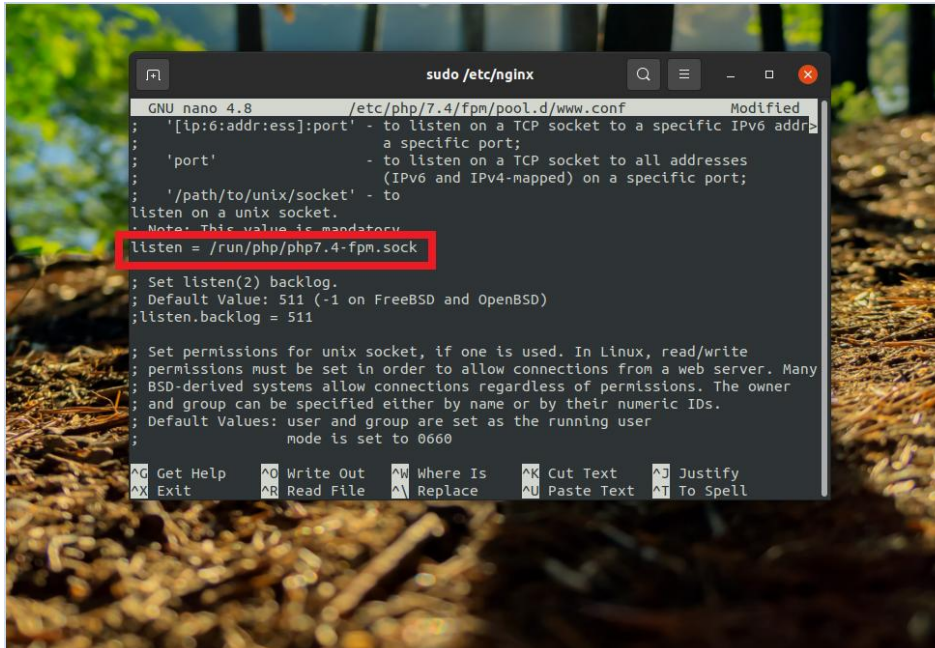
Чтобы принимать запросы FastCGI от Nginx, PHP-FPM может прослушивать сокет TCP/IP или UNIX сокет. Сокеты UNIX являются средством межпроцессного взаимодействия, которое обеспечивает эффективный обмен данными между процессами, работающими в одной и той же операционной системе, в то время как сокеты TCP/IP позволяют процессам обмениваться данными по сети.

Чтобы настроить PHP-FPM на прослушивание сокета UNIX, откройте файл конфигурации пула PHP-FPM по умолчанию:

```
sudo nano /etc/php/8.1/fpm/pool.d/www.conf
```

Находим директиву `listen` и задаем для нее путь к файлу сокета UNIX следующим образом:

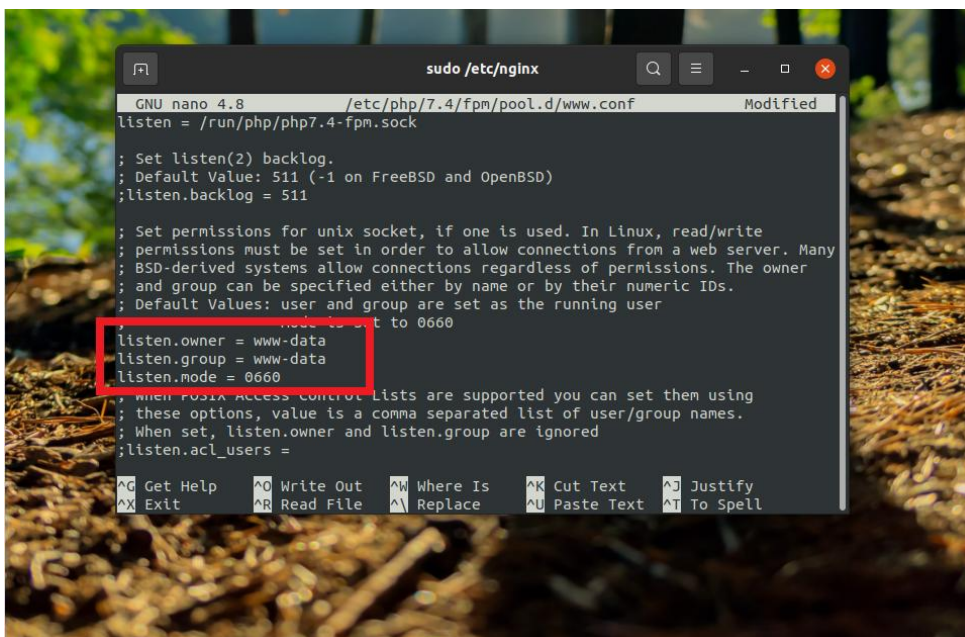
`listen = /run/php/php8.1-fpm.sock`



(эту картинку взял из интернета)

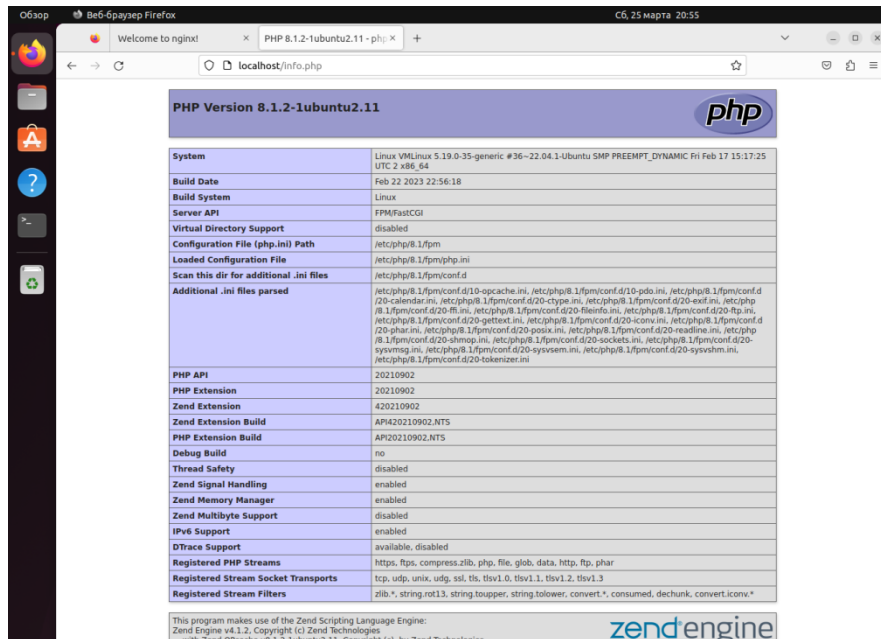
Устанавливаем соответствующие разрешения на чтение/запись для файла, чтобы разрешить подключения с веб-сервера NGINX. По умолчанию Nginx работает как пользователь `www-data` в Ubuntu.

Параметрам `listen.owner` и `listen.group` и задаем значение `www-data`. Также установите режим `0660`, для параметра `listen.mode`.



(эту картинку взял из интернета)

Заходим в браузер по адресу: <http://localhost/info.php>.



The screenshot shows a web browser window with the address bar displaying `localhost/info.php`. The page content is titled "PHP Version 8.1.2-1ubuntu2.11" and features the PHP logo. Below the title is a table of system and configuration information.

System	Linux VMLinux 5.19.0-35-generic #36~22.04.1-Ubuntu SMP PREEMPT_DYNAMIC Fri Feb 17 15:17:25 UTC 2 x86_64
Build Date	Feb 22 2023 22:56:18
Build System	Linux
Server API	FPM/FastCGI
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/8.1/fpm
Loaded Configuration File	/etc/php/8.1/fpm/php.ini
Scan this dir for additional .ini files	/etc/php/8.1/fpm/conf.d
Additional .ini files parsed	/etc/php/8.1/fpm/conf.d/10-opcache.ini, /etc/php/8.1/fpm/conf.d/10-pdo.ini, /etc/php/8.1/fpm/conf.d/10-calendar.ini, /etc/php/8.1/fpm/conf.d/20-ctype.ini, /etc/php/8.1/fpm/conf.d/20-exif.ini, /etc/php/8.1/fpm/conf.d/20-ffi.ini, /etc/php/8.1/fpm/conf.d/20-fileinfo.ini, /etc/php/8.1/fpm/conf.d/20-ftp.ini, /etc/php/8.1/fpm/conf.d/20-gettext.ini, /etc/php/8.1/fpm/conf.d/20-iconv.ini, /etc/php/8.1/fpm/conf.d/20-phar.ini, /etc/php/8.1/fpm/conf.d/20-posix.ini, /etc/php/8.1/fpm/conf.d/20-readline.ini, /etc/php/8.1/fpm/conf.d/20-shmop.ini, /etc/php/8.1/fpm/conf.d/20-sockets.ini, /etc/php/8.1/fpm/conf.d/20-sysvmsg.ini, /etc/php/8.1/fpm/conf.d/20-sysvsem.ini, /etc/php/8.1/fpm/conf.d/20-sysvshm.ini, /etc/php/8.1/fpm/conf.d/20-tokenizer.ini
PHP API	20210902
PHP Extension	20210902
Zend Extension	420210902
Zend Extension Build	API420210902.NTS
PHP Extension Build	API20210902.NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	enabled
Zend Memory Manager	enabled
Zend Multibyte Support	disabled
IPv6 Support	enabled
DTrace Support	available, disabled
Registered PHP Streams	https, ftps, compress.zlib, php, file, glob, data, http, ftp, phar
Registered Stream Socket Transports	tcp, udp, unix, udg, ssl, tls, tlsv1.0, tlsv1.1, tlsv1.2, tlsv1.3
Registered Stream Filters	zlib*, string.rot13, string.toupper, string.tolower, convert.*, consumed, dechunk, convert.iconv*

At the bottom of the page, there is a small text block: "This program makes use of the Zend Scripting Language Engine: Zend Engine v4.2.2, Copyright (c) Zend Technologies" and the Zend Engine logo.

Примечание (здесь можно найти более подробную информацию по настройке):

[https://losst.pro/ustanovka-nginx-i-php-fpm-v-ubuntu-20-](https://losst.pro/ustanovka-nginx-i-php-fpm-v-ubuntu-20-04#4_%D0%9D%D0%B0%D1%81%D1%82%D1%80%D0%BE%D0%B9%D0%BA%D0%B0_%D0%B1%D1%80%D0%B0%D0%BD%D0%B4%D0%BC%D0%B0%D1%83%D1%8D%D1%80%D0%B0)

[04#4_%D0%9D%D0%B0%D1%81%D1%82%D1%80%D0%BE%D0%B9%D0%BA%D0%B0_%D0%B1%D1%80%D0%B0%D0%BD%D0%B4%D0%BC%D0%B0%D1%83%D1%8D%D1%80%D0%B0](https://losst.pro/ustanovka-nginx-i-php-fpm-v-ubuntu-20-04#4_%D0%9D%D0%B0%D1%81%D1%82%D1%80%D0%BE%D0%B9%D0%BA%D0%B0_%D0%B1%D1%80%D0%B0%D0%BD%D0%B4%D0%BC%D0%B0%D1%83%D1%8D%D1%80%D0%B0)

3. Установить Apache:

`sudo apt install apache2`

Таблица сокетов:

`sudo ss -ntlp`

```
fedorsannikov@VMLinux: /etc/nginx/sites-enabled$ sudo ss -ntlp
State      Recv-Q    Send-Q    Local Address:Port    Peer Address:Port    Process
LISTEN     0          128       127.0.0.1:631         0.0.0.0:*             users:((("cupsd",pid=745,fd=7))
LISTEN     0          128       0.0.0.0:22           0.0.0.0:*             users:((("sshd",pid=700,fd=3))
LISTEN     0          511       0.0.0.0:80           0.0.0.0:*             users:((("nginx",pid=22899,fd=6),("nginx",pid=12171,fd=6))
LISTEN     0          4096      127.0.0.53:53        0.0.0.0:*             users:((("systemd-resolve",pid=387,fd=14))
LISTEN     0          128       [::]:22              [::]:*                users:((("sshd",pid=700,fd=4))
LISTEN     0          511       [::]:80              [::]:*                users:((("nginx",pid=22899,fd=7),("nginx",pid=12171,fd=7))
LISTEN     0          128       [::]:631             [::]:*                users:((("cupsd",pid=745,fd=6))
```

Таблица процессов:

`sudo ps afx`

```
22433 ?        S      0:00 \_ php-fpm: pool www
22434 ?        S      0:00 \_ php-fpm: pool www
23722 ?        Ss    0:00 /usr/libexec/fwupd/fwupd
23885 ?        Ss    0:00 /usr/bin/htcacheclean -d 120 -p /var/cache/apache2/mod_cache_disk -l 300M -n
```

Однако:

`sudo apachectl -t`

```
fedorsannikov@VMLinux: /etc/nginx/sites-enabled$ sudo apachectl -t
AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.1.1. Set the 'ServerName'
directive globally to suppress this messageSyntax OK
```

Итог:

Установлен, но не может запуститься так как 80-ый порт занят nginx.

4. Настроить обработку PHP. Добиться одновременной работы с Nginx.

4.1 Добиться одновременной работы Apache (Порт 8080) с Nginx (Порт 80):

Для того что бы Apache и Nginx работали одновременно необходимо после установки внести изменения в файл:

```
sudo nano /etc/apache2/ports.conf
```

Прописав работу Apache с портом 8080 вместо 80 (порт 8080 - альтернативный порт).

Было:

```
GNU nano 6.2 ports.conf
# If you just change the port or add more ports here, you will likely also
# have to change the VirtualHost statement in
# /etc/apache2/sites-enabled/000-default.conf
Listen 80

<IfModule ssl_module>
    Listen 443
</IfModule>

<IfModule mod_gnutls.c>
    Listen 443
</IfModule>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
```

Стало:

```
GNU nano 6.2 ports.conf
# If you just change the port or add more ports here, you will likely also
# have to change the VirtualHost statement in
# /etc/apache2/sites-enabled/000-default.conf
Listen 8080

<IfModule ssl_module>
    Listen 443
</IfModule>

<IfModule mod_gnutls.c>
    Listen 443
</IfModule>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
```

Проверяем синтаксис:

```
sudo apachectl -t
```

Применяем новую конфигурацию:

```
sudo systemctl reload apache2
```

Запускаем вручную (так как до этого был не запущен):

```
sudo systemctl start apache2
```

Проверяем статус:

```
sudo systemctl status apache2
```

```
fedorsannikov@VHLinux: /etc/apache2$ sudo systemctl status apache2
* apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
   Active: active (running) since Sat 2023-03-25 22:37:15 MSK; 23s ago
     Docs: https://httpd.apache.org/docs/2.4/
   Process: 24343 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
    Main PID: 24347 (apache2)
       Tasks: 55 (limit: 1869)
     Memory: 5.1M
        CPU: 25ms
    CGroup: /system.slice/apache2.service
            └─24347 /usr/sbin/apache2 -k start
              └─24348 /usr/sbin/apache2 -k start
                └─24349 /usr/sbin/apache2 -k start

Mar 25 22:37:15 VHLinux systemd[1]: Starting The Apache HTTP Server...
Mar 25 22:37:15 VHLinux apachectl[24346]: AH00558: apache2: could not reliably determine the server's fully qualified domain name, using 127.0.1.1. Set the 'ServerName' directive globally to suppress this message
Mar 25 22:37:15 VHLinux systemd[1]: Started The Apache HTTP Server.
```

Таблица сокетов:

```
sudo ss -ntlp
```

```
fedorsannikov@VHLinux: /etc/apache2$ sudo ss -ntlp
State      Recv-Q    Send-Q    Local Address:Port    Peer Address:Port    Process
LISTEN     0          128      127.0.0.1:631         0.0.0.0:*             users:((("cupsd",pid=745,fd=7))
LISTEN     0          128      0.0.0.0:22           0.0.0.0:*             users:((("sshd",pid=700,fd=3))
LISTEN     0          511      0.0.0.0:80           0.0.0.0:*             users:((("nginx",pid=22899,fd=6),("nginx",pid=22898,fd=6),("nginx",pid=12171,fd=6))
LISTEN     0          4096    127.0.0.53:53        0.0.0.0:*             users:((("systemd-resolve",pid=387,fd=14))
LISTEN     0          128      [::]:22              [::]:*                users:((("sshd",pid=700,fd=4))
LISTEN     0          511      [::]:80              [::]:*                users:((("nginx",pid=22899,fd=7),("nginx",pid=22898,fd=7),("nginx",pid=12171,fd=7))
LISTEN     0          511      *:8080               *:8080                users:((("apache2",pid=24349,fd=4),("apache2",pid=24348,fd=4),("apache2",pid=24347,fd=4))
LISTEN     0          128      [::]:631             [::]:*                users:((("cupsd",pid=745,fd=6))
```

Таблица процессов:

```
sudo ps afx
```

```
24347 ?        Ss      0:00 /usr/sbin/apache2 -k start
24348 ?        Sl      0:00 \_ /usr/sbin/apache2 -k start
24349 ?        Sl      0:00 \_ /usr/sbin/apache2 -k start
fedorsannikov@VHLinux: /etc/apache2$
```

Привязываем сайт по умолчанию к порту 8080:

```
cd /etc/apache2/sites-enabled
```

```
fedorsannikov@VMLinux:/etc/apache2/sites-enabled$ cd /etc/apache2/sites-enabled$ ll
итого 8
drwxr-xr-x 2 root root 4096 мар 25 23:03 ./
drwxr-xr-x 8 root root 4096 мар 25 22:55 ../
lrwxrwxrwx 1 root root   35 мар 25 21:17 000-default.conf -> ../sites-available/000-default.conf
```

Запускаем символьную ссылку:

```
sudo nano 000-default.conf
```

Вносим изменения:

Было:

```
GNU nano 6.2
<VirtualHost *:80>
    # The ServerName directive sets the request scheme, hostname and port that
    # the server uses to identify itself. This is used when creating
    # redirection URLs. In the context of virtual hosts, the ServerName
    # specifies what hostname must appear in the request's Host: header to
    # match this virtual host. For the default virtual host (this file) this
    # value is not decisive as it is used as a last resort host regardless.
    # However, you must set it for any further virtual host explicitly.
    #ServerName www.example.com

    ServerAdmin webmaster@localhost
    DocumentRoot /var/www/html

    # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
    # error, crit, alert, emerg.
    # It is also possible to configure the loglevel for particular
    # modules, e.g.
    #LogLevel info ssl:warn

    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined

    # For most configuration files from conf-available/, which are
    # enabled or disabled at a global level, it is possible to
    # include a line for only one particular virtual host. For example the
    # following line enables the CGI configuration for this host only
    # after it has been globally disabled with "a2disconf".
    #Include conf-available/serve-cgi-bin.conf
</VirtualHost>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
```

Стало:

```
GNU nano 6.2
<VirtualHost *:8080>
# The ServerName directive sets the request scheme, hostname and port that
# the server uses to identify itself. This is used when creating
# redirection URLs. In the context of virtual hosts, the ServerName
# specifies what hostname must appear in the request's Host: header to
# match this virtual host. For the default virtual host (this file) this
# value is not decisive as it is used as a last resort host regardless.
# However, you must set it for any further virtual host explicitly.
#ServerName www.example.com

ServerAdmin webmaster@localhost
DocumentRoot /var/www/html

# Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
# error, crit, alert, emerg.
# It is also possible to configure the loglevel for particular
# modules, e.g.
#LogLevel info ssl:warn

ErrorLog ${APACHE_LOG_DIR}/error.log
CustomLog ${APACHE_LOG_DIR}/access.log combined

# For most configuration files from conf-available/, which are
# enabled or disabled at a global level, it is possible to
# include a line for only one particular virtual host. For example the
# following line enables the CGI configuration for this host only
# after it has been globally disabled with "a2disconf".
#Include conf-available/serve-cgi-bin.conf
</VirtualHost>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
```

Для того что бы при обращении к localhost:8080 (apache) и localhost:80 (nginx) загружались разные страницы исправим файл настроек nginx:

```
cd /etc/nginx/sites-enabled
```

Переходим по символической ссылке:

```
default -> /etc/nginx/sites-available/default
```

```
sudo nano default
```

Было:

```
server {
    listen 80 default_server;
    listen [::]:80 default_server;

    # SSL configuration
    #
    # listen 443 ssl default_server;
    # listen [::]:443 ssl default_server;
    #
    # Note: You should disable gzip for SSL traffic.
    # See: https://bugs.debian.org/773332
    #
    # Read up on ssl_ciphers to ensure a secure configuration.
    # See: https://bugs.debian.org/765782
    #
    # Self signed certs generated by the ssl-cert package
    # Don't use them in a production server!
    #
    # include snippets/snakeoil.conf;

    root /var/www/html;

    # Add index.php to the list if you are using PHP
    index index.html index.htm index.nginx-debian.html;

    server_name _;

    location / {
        # First attempt to serve request as file, then
        # as directory, then fall back to displaying a 404.
        try_files $uri $uri/ =404;
    }

    # pass PHP scripts to FastCGI server

    location ~ \.php$ {
        include snippets/fastcgi-php.conf;
        root /var/www/html;
        fastcgi_pass unix:/run/php/php8.1-fpm.sock;
    }
}
```

Стало:

```
#
server {
    listen 80 default_server;
    listen [::]:80 default_server;

    # SSL configuration
    #
    # listen 443 ssl default_server;
    # listen [::]:443 ssl default_server;
    #
    # Note: You should disable gzip for SSL traffic.
    # See: https://bugs.debian.org/773332
    #
    # Read up on ssl_ciphers to ensure a secure configuration.
    # See: https://bugs.debian.org/765782
    #
    # Self signed certs generated by the ssl-cert package
    # Don't use them in a production server!
    #
    # include snippets/snakeoil.conf;

    root /var/www/html;

    # Add index.php to the list if you are using PHP
    # index index.html index.htm index.nginx-debian.html;
    index index.nginx-debian.html

    server_name _;

    location / {
```

Теперь при обращении к localhost:8080 (apache) и localhost:80 (nginx) будут загружаться разные странички.

Загружаем страницу по умолчанию в терминале:

curl localhost:8080

```
fedorsannikov@VMLinux:/etc/apache2$ curl localhost:8080
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <!--
    Modified from the Debian original for Ubuntu
    Last updated: 2022-03-22
    See: https://launchpad.net/bugs/1966004
  -->
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
    <title>Apache2 Ubuntu Default Page: It works</title>
    <style type="text/css" media="screen">
      * {
        margin: 0px 0px 0px 0px;
        padding: 0px 0px 0px 0px;
      }

      body, html {
        padding: 3px 3px 3px 3px;

        background-color: #D8DBE2;

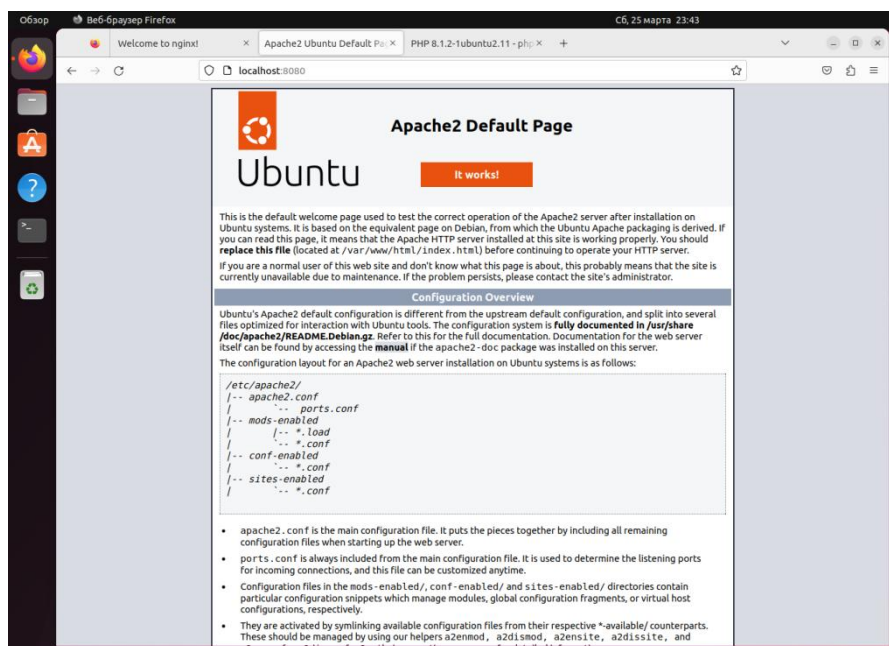
        font-family: Ubuntu, Verdana, sans-serif;
        font-size: 11pt;
        text-align: center;
      }

      div.main_page {
        position: relative;
        display: table;

        width: 800px;
      }
    </style>
  </head>
  <body>
    <div class="main_page">
      <div style="float: left; width: 60px; text-align: center; padding-right: 10px;>
        <img alt="Ubuntu logo" data-bbox="288 503 315 530"/>
      </div>
      <div style="float: right; width: 150px; text-align: center; padding-left: 10px;>
        <div style="background-color: #A8D08F; color: white; padding: 2px 5px; display: inline-block;>It works!</div>
      </div>
      <div style="clear: both;></div>
      <div style="text-align: center; padding-top: 10px;>
        <p>This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should <b>replace this file</b> (located at /var/www/html/index.html) before continuing to operate your HTTP server.</p>
        <p>If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.</p>
        <div style="background-color: #D8DBE2; padding: 5px; margin: 10px 0;>
          Configuration Overview
        </div>
        <p>Ubuntu's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Ubuntu tools. The configuration system is <b>fully documented in /usr/share/doc/apache2/README.Debian.gz</b>. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the <b>manual</b> if the apache2-doc package was installed on this server.</p>
        <p>The configuration layout for an Apache2 web server installation on Ubuntu systems is as follows:</p>
        <pre>/etc/apache2/
|-- apache2.conf
/  |-- ports.conf
|-- mods-enabled
/    |-- *.load
/    |-- *.conf
|-- conf-enabled
/    |-- *.conf
|-- sites-enabled
/    |-- *.conf
-->
      </div>
    </div>
  </body>
</html>
```

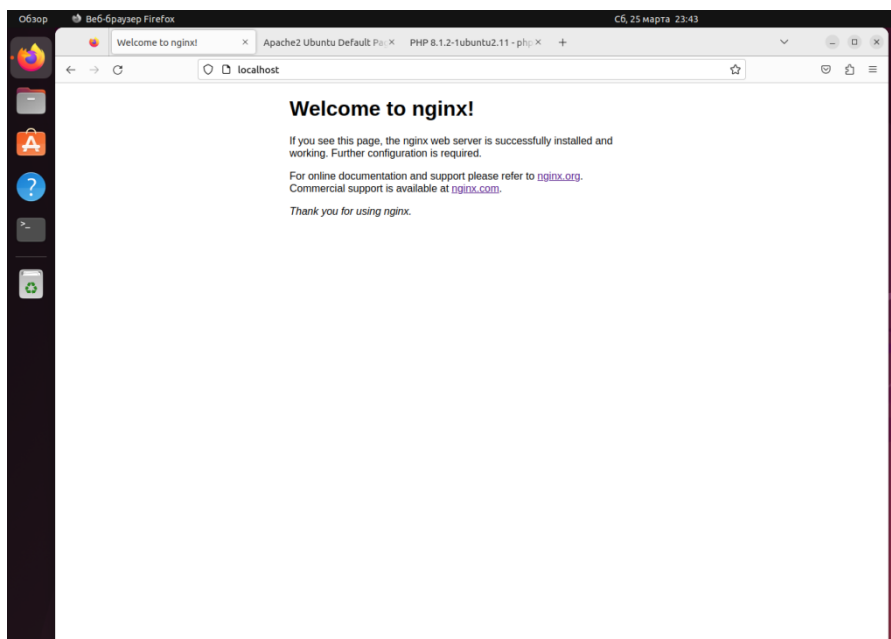
Загружаем страницу по умолчанию в браузере:

localhost:8080



А вот страничка nginx:

localhost:80 или localhost



4.2 Настроить обработку PHP:

Устанавливаем пакеты php8.1 для apache2:

```
apt install libapache2-mod-php8.1 php8.1
```

Файл создан ранее (смотри выше) в директории по умолчанию /var/www/html со следующим содержанием:

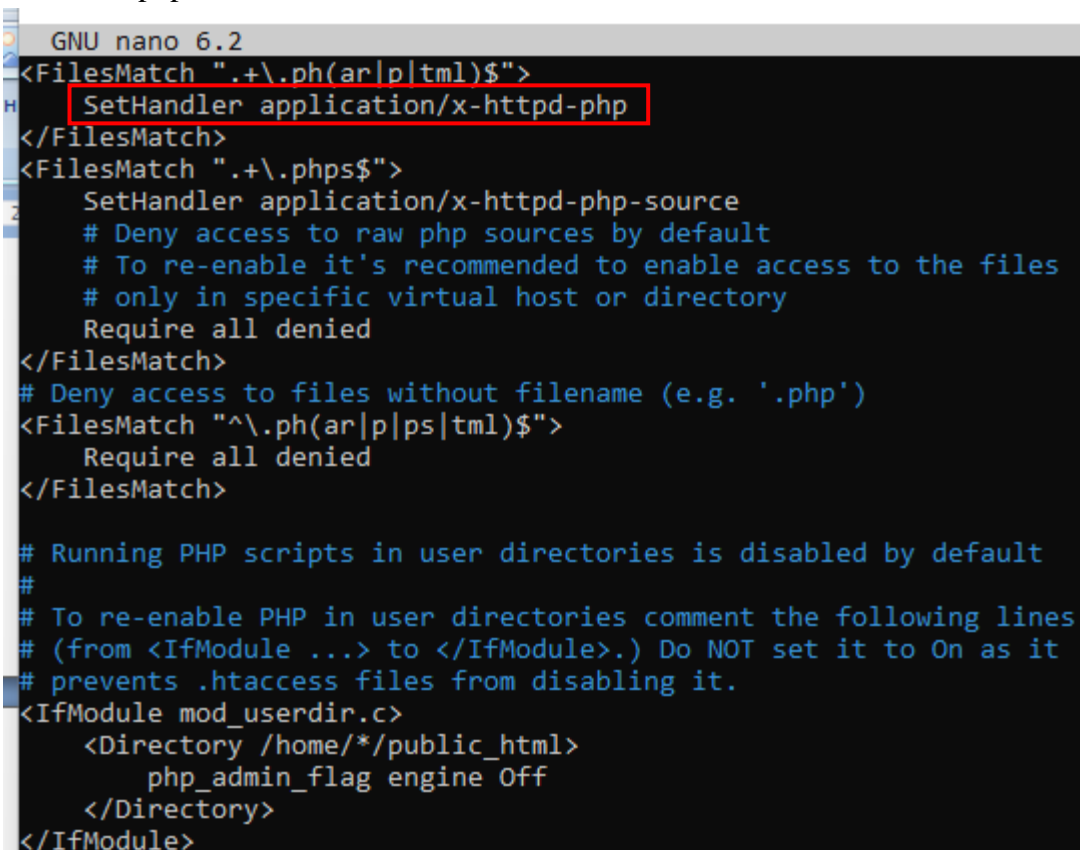
```
<?php  
phpinfo();  
?>
```

Обработка PHP подключается автоматически, по расширению файла:

Можно проверить:

```
cd /etc/apache2/mods-enabled
```

```
sudo nano php8.1.conf
```

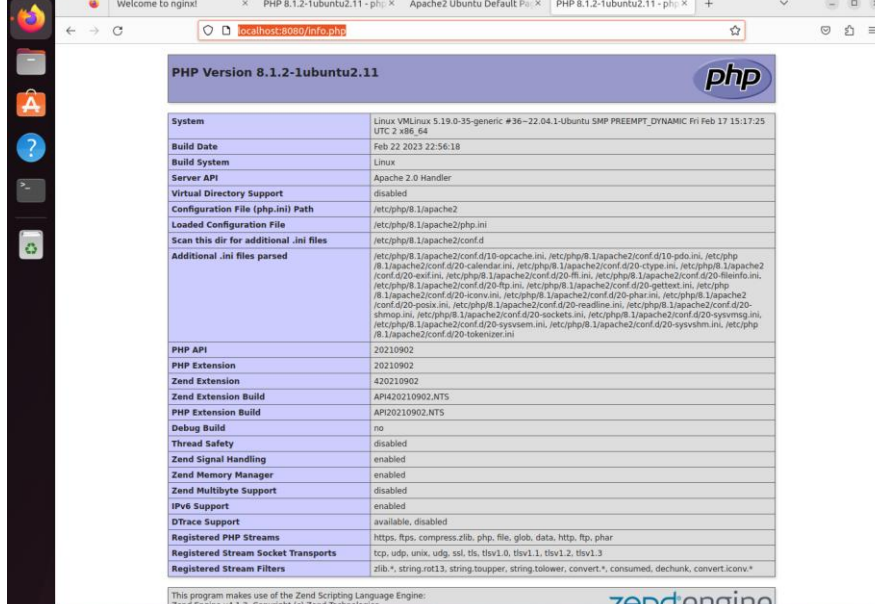


```
GNU nano 6.2  
<FilesMatch "\.+\.(ph|ar|p|tml)$">  
    SetHandler application/x-httpd-php  
</FilesMatch>  
<FilesMatch "\.+\.(phps)$">  
    SetHandler application/x-httpd-php-source  
    # Deny access to raw php sources by default  
    # To re-enable it's recommended to enable access to the files  
    # only in specific virtual host or directory  
    Require all denied  
</FilesMatch>  
# Deny access to files without filename (e.g. '.php')  
<FilesMatch "^\.ph(ar|p|ps|tml)$">  
    Require all denied  
</FilesMatch>  
  
# Running PHP scripts in user directories is disabled by default  
#  
# To re-enable PHP in user directories comment the following lines  
# (from <IfModule ...> to </IfModule>.) Do NOT set it to On as it  
# prevents .htaccess files from disabling it.  
<IfModule mod_userdir.c>  
    <Directory /home/*/public_html>  
        php_admin_flag engine Off  
    </Directory>  
</IfModule>
```

```
</div></body></html>fedorsannikov@VMLinu
```

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"><head>
<style type="text/css">
body {background-color: #fff; color: #222; font-family: sans-serif;}
pre {margin: 0; font-family: monospace;}
a:link {color: #009; text-decoration: none; background-color: #fff;}
a:hover {text-decoration: underline;}
table {border-collapse: collapse; border: 0; width: 934px; box-shadow: 1px 2px 3px #ccc;}
tr.center {text-align: center;}
tr.center table {margin: 1em auto; text-align: left;}
tr.center th {text-align: center !important;}
td, th {border: 1px solid #666; font-size: 75%; vertical-align: baseline; padding: 4px 5px;}
th {position: sticky; top: 0; background: inherit;}
h1 {font-size: 150%;}
h2 {font-size: 125%;}
p {text-align: left;}
div.e {background-color: #ccf; width: 300px; font-weight: bold;}
div.h {background-color: #99c; font-weight: bold;}
div.v {background-color: #ddd; max-width: 300px; overflow-x: auto; word-wrap: break-word;}
div.v i {color: #999;}
img {float: right; border: 0;}
hr {width: 934px; background-color: #ccc; border: 0; height: 1px;}
</style>
<title>PHP 8.1.2-ubuntu2.11 - phpinfo()</title><meta name="ROBOTS" content="NOINDEX,NOFOLLOW,NOARCHIVE" /></head>
<body><div class="center">
<table>
<tr class="h"><td>
<a href="http://www.php.net/">


```



5. Настроить схему обратного прокси для Nginx (динамика - на Apache).

Директория nginx в которой располагается файл default, в который необходимо внести изменения:

```
cd /etc/nginx/sites-available
```

```
sudo nano default
```

```
# Default server configuration
#
server {
    listen 80 default_server;
    listen [::]:80 default_server;

    # SSL configuration
    #
    # listen 443 ssl default_server;
    # listen [::]:443 ssl default_server;
    #
    # Note: You should disable gzip for SSL traffic.
    # See: https://bugs.debian.org/773332
    #
    # Read up on ssl_ciphers to ensure a secure configuration.
    # See: https://bugs.debian.org/765782
    #
    # Self signed certs generated by the ssl-cert package
    # Don't use them in a production server!
    #
    # include snippets/snakeoil.conf;

    root /var/www/html;

    # Add index.php to the list if you are using PHP
    index index.html index.htm index.nginx-debian.html;
    # index index.nginx-debian.html

    server_name _;

    # Динамический запрос (перенадресация на Apache2)
    location / {
        proxy_pass http://localhost:8080;
        proxy_set_header Host $host;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header X-Real-IP $remote_addr;
    }

    # Статический запрос
    location ~* ^.+.(jpg|jpeg|gif|png|ico|css|zip|pdf|txt|tar|js)$ {
        root /var/www/html;
    }

    # location / {
    #     # First attempt to serve request as file, then
    #     # as directory, then fall back to displaying a 404.
    #     #try_files $uri $uri/ =404;
    # }

    # pass PHP scripts to FastCGI server

    location ~ \.php$ {
        include snippets/fastcgi-php.conf;
        root /var/www/html;
        fastcgi_pass unix:/run/php/php8.1-fpm.sock;
    }
}
```

proxy_pass - задаёт протокол и адрес проксируемого сервера;

proxy_set_header - позволяет переопределять или добавлять поля заголовка запроса, передаваемые проксируемому серверу;

proxy_set_header Host \$host; — в запросе к сайту донору выставляет правильное поле HOST.

proxy_set_header X-Forwarded-For \$proxy_add_x_forwarded_for; - для распознавания заголовка X-Forwarded-For.

X-Forwarded-For является де-факто стандартным заголовком для идентификации происхождения IP-адреса клиента, подключающегося к веб-серверу через HTTP-прокси или балансировщик нагрузки.

proxy_set_header X-Real-IP \$remote_addr; - отправляем реальный адрес посетителя приложению в заголовке X-Real-IP header:

6. Установить MySQL. Создать новую базу данных и таблицу в ней.

Установка сервера MySQL:

```
apt install mysql-server-8.0
```

Таблица сокетов:

```
sudo ss -ntlp
```

State	Recv-Q	Send-Q	Local Address:Port	Peer Address:Port	Process
LISTEN	0	151	127.0.0.1:3306	0.0.0.0:*	users:(("mysqld",pid=28511,fd=23))
LISTEN	0	128	127.0.0.1:631	0.0.0.0:*	users:(("cupsd",pid=27538,fd=7))
LISTEN	0	128	0.0.0.0:22	0.0.0.0:*	users:(("sshd",pid=700,fd=3))
LISTEN	0	511	0.0.0.0:80	0.0.0.0:*	users:(("nginx",pid=25741,fd=6),("nginx",pid=25740,fd=6))
LISTEN	0	4096	127.0.0.53:53	0.0.0.0:*	users:(("systemd-resolve",pid=387,fd=14))
LISTEN	0	70	127.0.0.1:33060	0.0.0.0:*	users:(("mysqld",pid=28511,fd=21))
LISTEN	0	128	:::22	:::*	users:(("sshd",pid=700,fd=4))
LISTEN	0	511	:::80	:::*	users:(("nginx",pid=25741,fd=7),("nginx",pid=25740,fd=7))
LISTEN	0	511	:::8080	:::*	users:(("apache2",pid=27401,fd=4),("apache2",pid=27310,fd=4),("apache2",pid=27309,fd=4),("apache2",pid=27308,fd=4),("apache2",pid=27307,fd=4),("apache2",pid=27306,fd=4),("apache2",pid=27078,fd=4))
LISTEN	0	128	:::631	:::*	users:(("cupsd",pid=27538,fd=6))

Входим в администратора MySQL (пользователь root):

```
sudo mysql
```

Смотрим, какие базы данных уже есть:

```
show databases;
```

```
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| sys |
+-----+
4 rows in set (0,03 sec)
```

Создадим свою базу данных:

CREATE DATABASE instruction_data_database;

```
mysql> CREATE DATABASE instruction_data_database;
Query OK, 1 row affected (0,02 sec)

mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| instruction_data_database |
| mysql |
| performance_schema |
| sys |
+-----+
5 rows in set (0,00 sec)
```

Наполняем созданную базу данных:

Подключаемся к созданной базе данных:

use instruction_data_database

```
mysql> use instruction_data_database
Database changed
```

Создаем таблицу:

create table test(col1 int, col text);

```
mysql> create table test(col1 int, col text);
Query OK, 0 rows affected (0,03 sec)
```

Заполняем таблицу:

insert into test(col1, col) values(7, '-text-');

```
mysql> insert into test(col1, col) values(7, '-text-')
-> ;
Query OK, 1 row affected (0,02 sec)
```

Просматриваем/выводим таблицу:

```
mysql> select * from test;
+-----+-----+
| col1 | col  |
+-----+-----+
|    7 | -text- |
+-----+-----+
1 row in set (0,00 sec)
```

select * from test;

Выходим из mysql:

```
mysql> exit
Bye
fedorsannikov@VMLinux:/etc/apache2/mods-enabled$
```

7. Установить пакет `phpmyadmin` и запустить его веб-интерфейс для управления MySQL.

Ссылки на используемый материал:

<https://www.digitalocean.com/community/tutorials/how-to-install-and-secure-phpmyadmin-on-ubuntu-20-04-ru>

Установим `php` и другие пакеты, необходимые для веб-разработки:

```
sudo apt install phpmyadmin php-mbstring php-zip php-gd php-json php-curl
```

```
sudo phpenmod mbstring
```

```
sudo systemctl restart apache2
```

Настройка доступа по паролю для учетной записи `root` в MySQL:

```
sudo mysql
```

```
ALTER USER 'root'@'localhost' IDENTIFIED WITH caching_sha2_password BY 'мой пароль для входа';
```

Проверяем метод аутентификации `root`:

```
SELECT user,authentication_string,plugin,host FROM mysql.user;
```

Должно быть: `caching_sha2_password`

Данный метод необходим для аутентификации по паролю при входе в `mysql`.

Теперь вход в `mysql` только с помощью и при вводе заданного ранее пароля:

```
mysql -u root -p
```

Создаем нового пользователя и придумываем ему пароль (для работы ну и что бы попрактиковаться):

```
CREATE USER 'fedor'@'localhost' IDENTIFIED WITH caching_sha2_password BY '1';
```

Даем `fedor` все возможные права (`GRANT ALL PRIVILEGES ON`):

```
GRANT ALL PRIVILEGES ON *.* TO 'fedor'@'localhost' WITH GRANT OPTION;
```



```
mysql> CREATE USER 'fedor'@'localhost' IDENTIFIED WITH caching_sha2_password BY '1';
Query OK, 0 rows affected (0,03 sec)

mysql> GRANT ALL PRIVILEGES ON *.* TO 'fedor'@'localhost' WITH GRANT OPTION;
Query OK, 0 rows affected (0,01 sec)
```

Выходим.

Exit

Можно начинать работать через вебинтерфэйс.

Для входа в вебинтерфэйс необходимо набрать в адресной строке браузера:

`http://localhost:8080/phpmyadmin`

