

Установка Docker

```
sudo apt update
sudo apt install apt-transport-https ca-certificates curl software-properties-common
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu bionic stable"
sudo apt update
apt-cache policy docker-ce
sudo apt install docker-ce
sudo systemctl status docker

sudo usermod -aG docker ${USER}
su - ${USER}
id -nG
sudo usermod -aG docker user1
```

Установка Docker Compose

```
sudo curl -L https://github.com/docker/compose/releases/download/1.29.2/docker-compose-`uname -s`-`uname -m` -o /usr/local/bin/docker-compose
sudo chmod +x /usr/local/bin/docker-compose
```

Установка WordPress

Установка и настройка

```
mkdir wordpress && cd wordpress
```

```
mkdir nginx-conf
```

```
nano nginx-conf/nginx.conf
```

```
server {
```

```
    listen 80;
```

```
    listen [::]:80;
```

```
    server_name example.com www.example.com localhost;
```

```
    index index.php index.html index.htm;
```

```
    root /var/www/html;
```

```
    location ~ /.well-known/acme-challenge {
        allow all;
    }
```

```

        root /var/www/html;
    }

    location / {
        try_files $uri $uri/ /index.php$is_args$args;
    }

    location ~ /\.php$ {
        try_files $uri =404;
        fastcgi_split_path_info ^(.+\.(php|/.+))$;
        fastcgi_pass wordpress:9000;
        fastcgi_index index.php;
        include fastcgi_params;
        fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;
        fastcgi_param PATH_INFO $fastcgi_path_info;
    }

    location ~ /\.ht {
        deny all;
    }

    location = /favicon.ico {
        log_not_found off; access_log off;
    }
    location = /robots.txt {
        log_not_found off; access_log off; allow all;
    }
    location ~* \.(css|gif|ico|jpeg|jpg|js|png)$ {
        expires max;
        log_not_found off;
    }
}

```

nano .env

```

MYSQL_ROOT_PASSWORD=your_root_password
MYSQL_USER=your_wordpress_database_user
MYSQL_PASSWORD=your_wordpress_database_password

```

git init

```
nano .gitignore
```

```
---
```

```
.env
```

```
---
```

```
nano .dockerignore
```

```
---
```

```
.env
```

```
.git
```

```
docker-compose.yml
```

```
.dockerignore
```

```
---
```

```
nano docker-compose.yml
```

```
---
```

```
version: '3'
```

```
services:
```

```
  db:
```

```
    image: mysql:8.0
```

```
    container_name: db
```

```
    restart: unless-stopped
```

```
    env_file: .env
```

```
    environment:
```

```
      - MYSQL_DATABASE=wordpress
```

```
    volumes:
```

```
      - dbdata:/var/lib/mysql
```

```
    command: '--default-authentication-plugin=mysql_native_password'
```

```
    networks:
```

```
      - app-network
```

```
wordpress:
```

```
  depends_on:
```

```
    - db
```

```
  image: wordpress:5.1.1-fpm-alpine
```

```
  container_name: wordpress
```

```
  restart: unless-stopped
```

```
  env_file: .env
```

```
  environment:
```

```
    - WORDPRESS_DB_HOST=db:3306
```

```
    - WORDPRESS_DB_USER=$MYSQL_USER
```

```
    - WORDPRESS_DB_PASSWORD=$MYSQL_PASSWORD
```

```
    - WORDPRESS_DB_NAME=wordpress
```

volumes:

- wordpress:/var/www/html

networks:

- app-network

webserver:

depends_on:

- wordpress

image: nginx:1.15.12-alpine

container_name: webserver

restart: unless-stopped

ports:

- "80:80"

volumes:

- wordpress:/var/www/html
- ./nginx-conf:/etc/nginx/conf.d
- certbot-etc:/etc/letsencrypt

networks:

- app-network

certbot:

depends_on:

- webserver

image: certbot/certbot

container_name: certbot

volumes:

- certbot-etc:/etc/letsencrypt
- wordpress:/var/www/html

command: certonly --webroot --webroot-path=/var/www/html --email sammy@example.com --agree-tos --no-eff-email --staging -d example.com -d www.example.com

volumes:

certbot-etc:

wordpress:

dbdata:

networks:

```
app-network:  
  driver: bridge
```

```
---
```

```
docker-compose up -d  
docker-compose ps
```