

В соответствии с написанным условием задания сервис, состоящий из 2 различных контейнеров: 1 - веб, 2 - БД

```
alex@alex-VirtualBox:~/Documents/Containers/composeTest$ touch docker-compose.yml
```

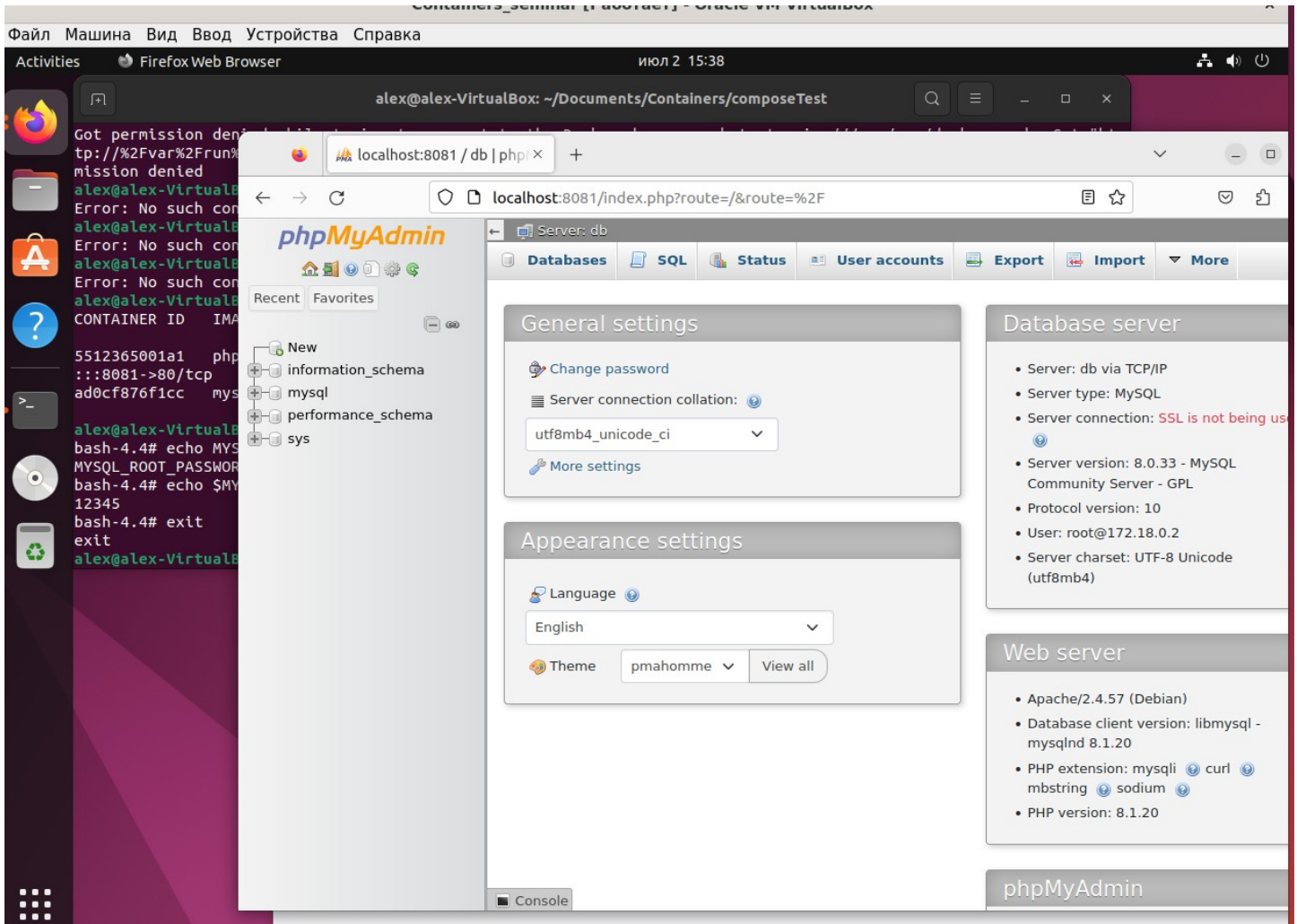
```
alex@alex-VirtualBox:~/Documents/Containers/composeTest$ cat docker-compose.yml
services:
  db:
    image: mysql:latest
    networks:
      - db-net
    environment:
      - MYSQL_ROOT_PASSWORD=12345
  watcherDB:
    image: phpmyadmin:latest
    networks:
      - db-net
    ports:
      - 8081:80
networks:
  db-net:
```

```
sudo apt install docker-compose
```

Файл Машина Вид Ввод Устройства Справка

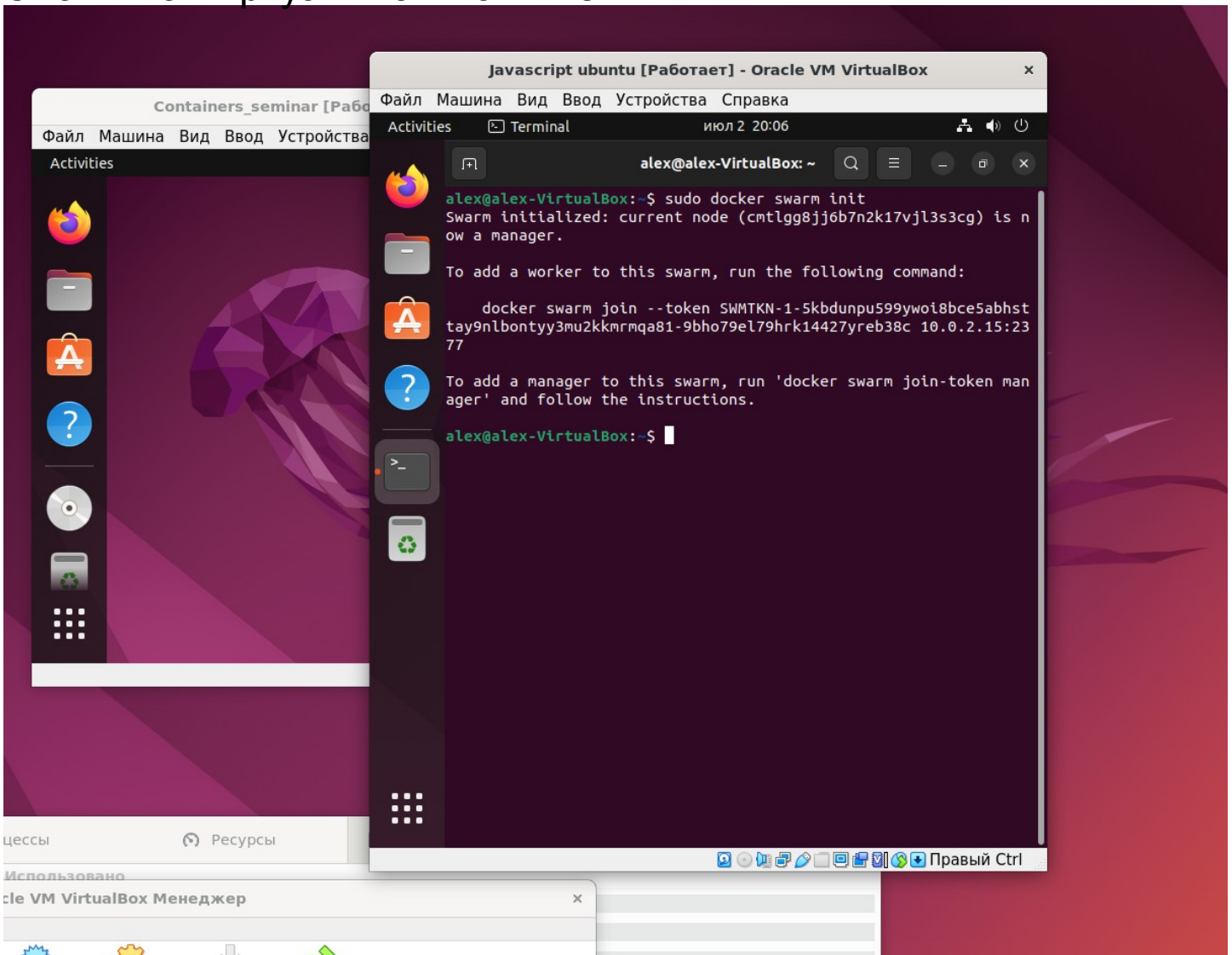
```
alex@alex-VirtualBox:~/Documents/Containers/composeTest$ sudo docker-compose up -d
Creating network "composetest_db-net" with the default driver
Pulling db (mysql:latest)...
latest: Pulling from library/mysql
46ef68baacb7: Pull complete
94c1114b2e9c: Pull complete
ff05e3f38802: Pull complete
41cc3fcd9912: Pull complete
07bbc8bdf52a: Pull complete
6d88f83726a9: Pull complete
cf5c7d5d33f7: Pull complete
9db3175a2a66: Pull complete
feaadedb27fa9: Pull complete
cf91e7784414: Pull complete
b1770db1c329: Pull complete
Digest: sha256:15f069202c46cf861ce429423ae3f8dfa6423306fbf399eaeef36094ce30dd75c
Status: Downloaded newer image for mysql:latest
Pulling watcherDB (phpmyadmin:latest)...
latest: Pulling from library/phpmyadmin
5b5fe70539cd: Pull complete
affe9439d2a2: Pull complete
1684de57270e: Pull complete
dc968f4da64f: Pull complete
9b1cf1bf39c5: Pull complete
719f04c53973: Pull complete
74b24852707c: Pull complete
54373d1fcd4e: Pull complete
27a6a57081a1: Pull complete
d40c0e518300: Pull complete
36b134b9a8c3: Pull complete
91324d697396: Pull complete
9d78e346f9b5: Pull complete
a0afb6c62836: Pull complete
4ed2811642a8: Pull complete
a2fe86a0097f: Pull complete
67ea3c3d96a4: Pull complete
149d5a5803a0: Pull complete
Digest: sha256:99af7a03a368ea0fadb99dd48f4946537df0ca3f5e3001454b9619533ed35eda
Status: Downloaded newer image for phpmyadmin:latest
Creating composetest_db_1 ... done
Creating composetest_watcherDB_1 ... done
alex@alex-VirtualBox:~/Documents/Containers/composeTest$
```

Запускаем контейнеры под
Compose



создать 3 сервиса в каждом окружении (dev, prod, lab)

Проинициализируем Docker swarm на виртуальной машине



Откроем порты

Установим app для работы с портами

```
root@alex-VirtualBox:/home/alex# firewall-cmd
```


Откроем порты на управляющей ноде

```
root@alex-VirtualBox:/home/alex# firewall-cmd --add-port=2376/tcp --permanent;
success
root@alex-VirtualBox:/home/alex# firewall-cmd --add-port=2377/tcp --permanent;
success
root@alex-VirtualBox:/home/alex# firewall-cmd --add-port=7946/tcp --permanent;
success
root@alex-VirtualBox:/home/alex# firewall-cmd --add-port=7946/udp --permanent;
success
root@alex-VirtualBox:/home/alex# firewall-cmd --add-port=4789/udp --permanent;
success
root@alex-VirtualBox:/home/alex#
```

```
root@alex-VirtualBox:/home/alex# systemctl restart docker;
```

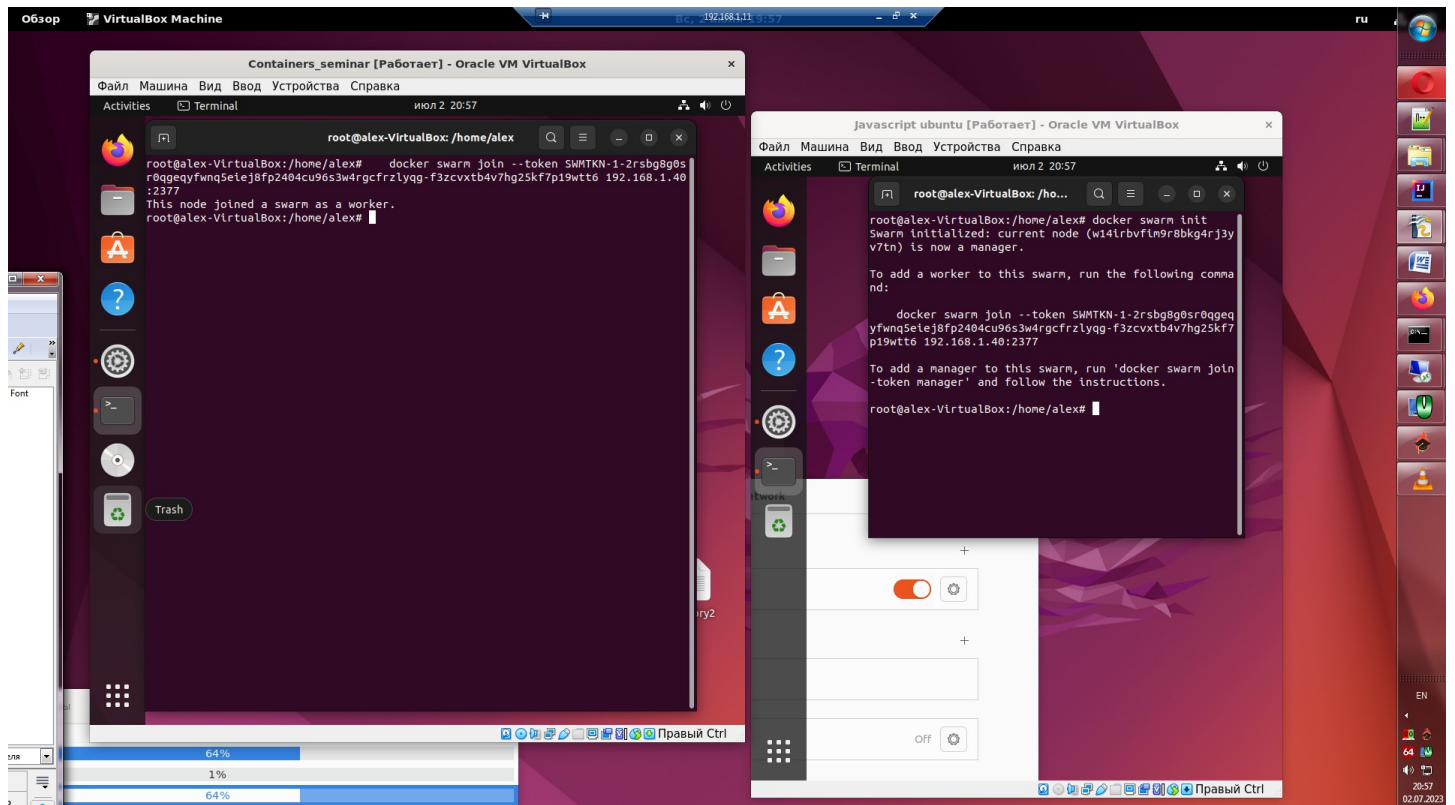
Для worker node

The screenshot displays a VirtualBox window titled "Containers_seminar [Работает] - Oracle VM VirtualBox". Inside, a Linux desktop environment is visible. A terminal window is open, showing the following commands and output:

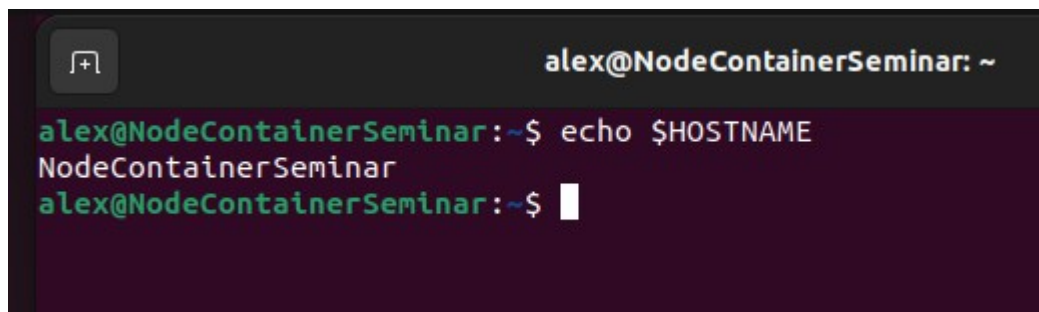
```
root@alex-VirtualBox:/home/alex# firewall-cmd --add-port=2376/tcp --permanent;
firewall-cmd --add-port=7946/tcp --permanent;
firewall-cmd --add-port=7946/udp --permanent;
firewall-cmd --add-port=4789/udp --permanent;
firewall-cmd --reload;
systemctl restart docker;
Warning: ALREADY_ENABLED: 2376:tcp
success
success
success
success
success
root@alex-VirtualBox:/home/alex#
```

The desktop environment includes a sidebar with application icons (Firefox, Files, App Store, Help, Terminal, etc.) and a taskbar at the bottom. A file named "history2" is visible on the desktop. The terminal window title is "root@alex-VirtualBox: /home/alex".

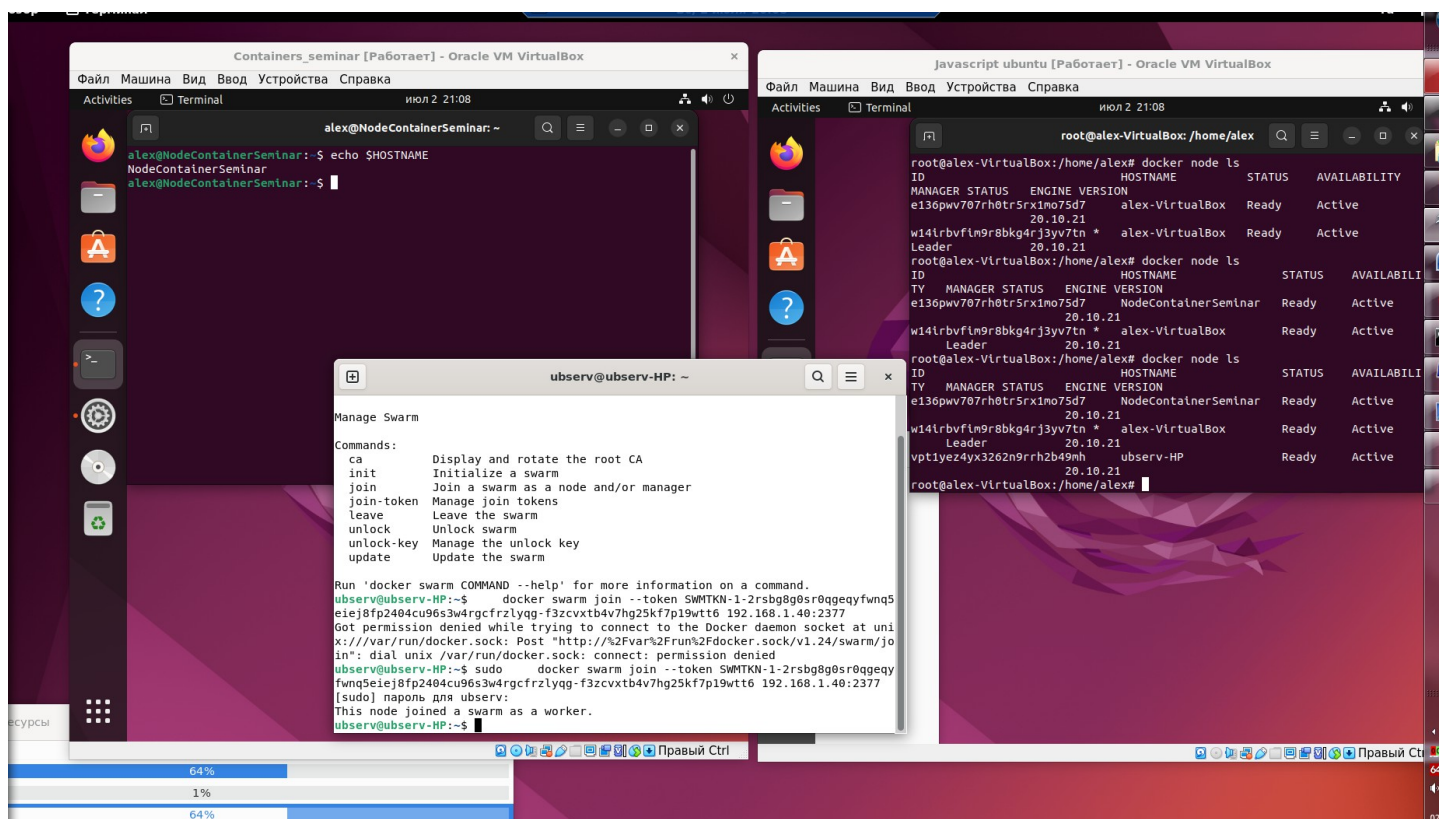
Поменяем на основной ноде Virtual Box тип подключения на мост , для создания разных ip



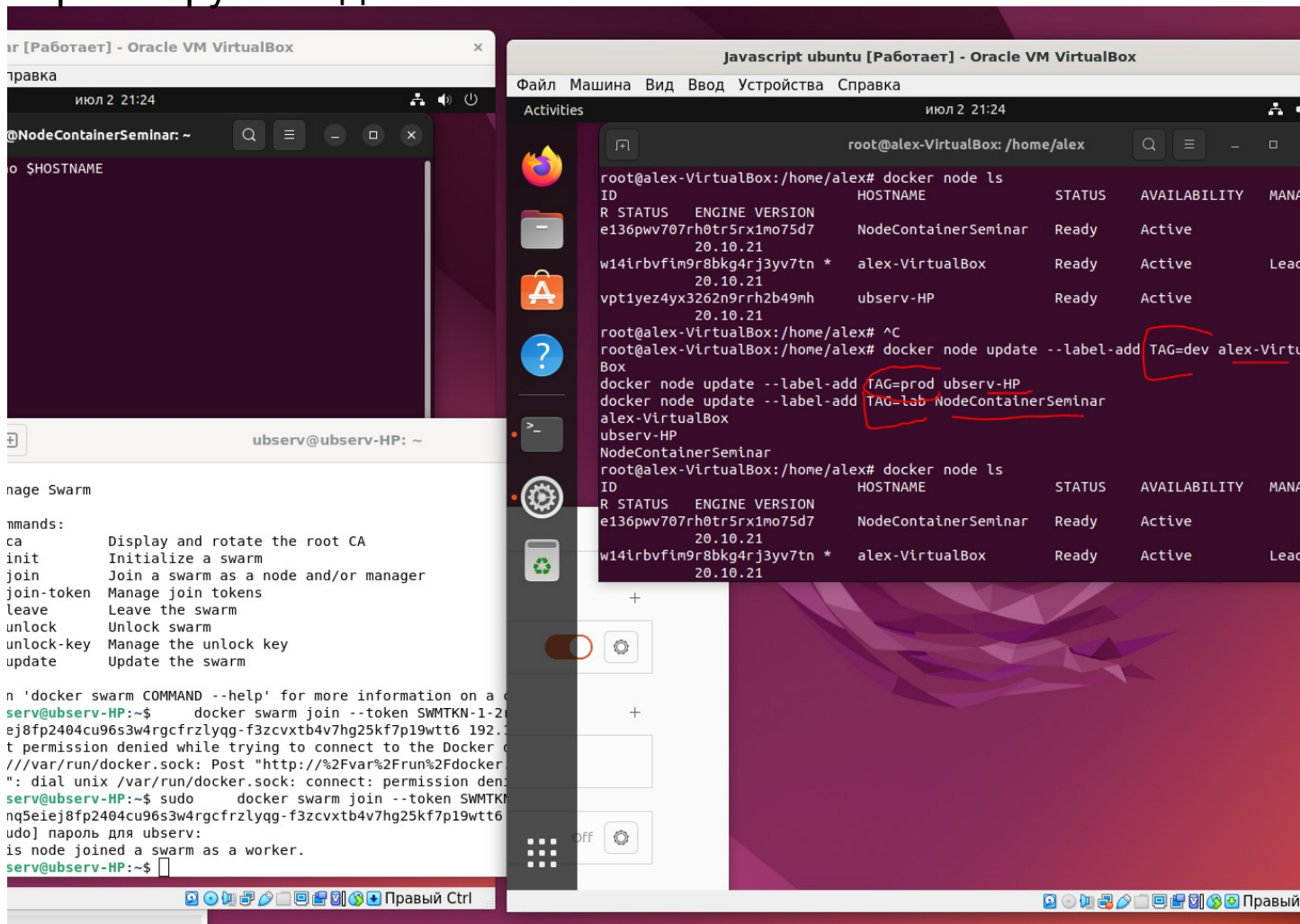
Переименуем одну Worker ноду



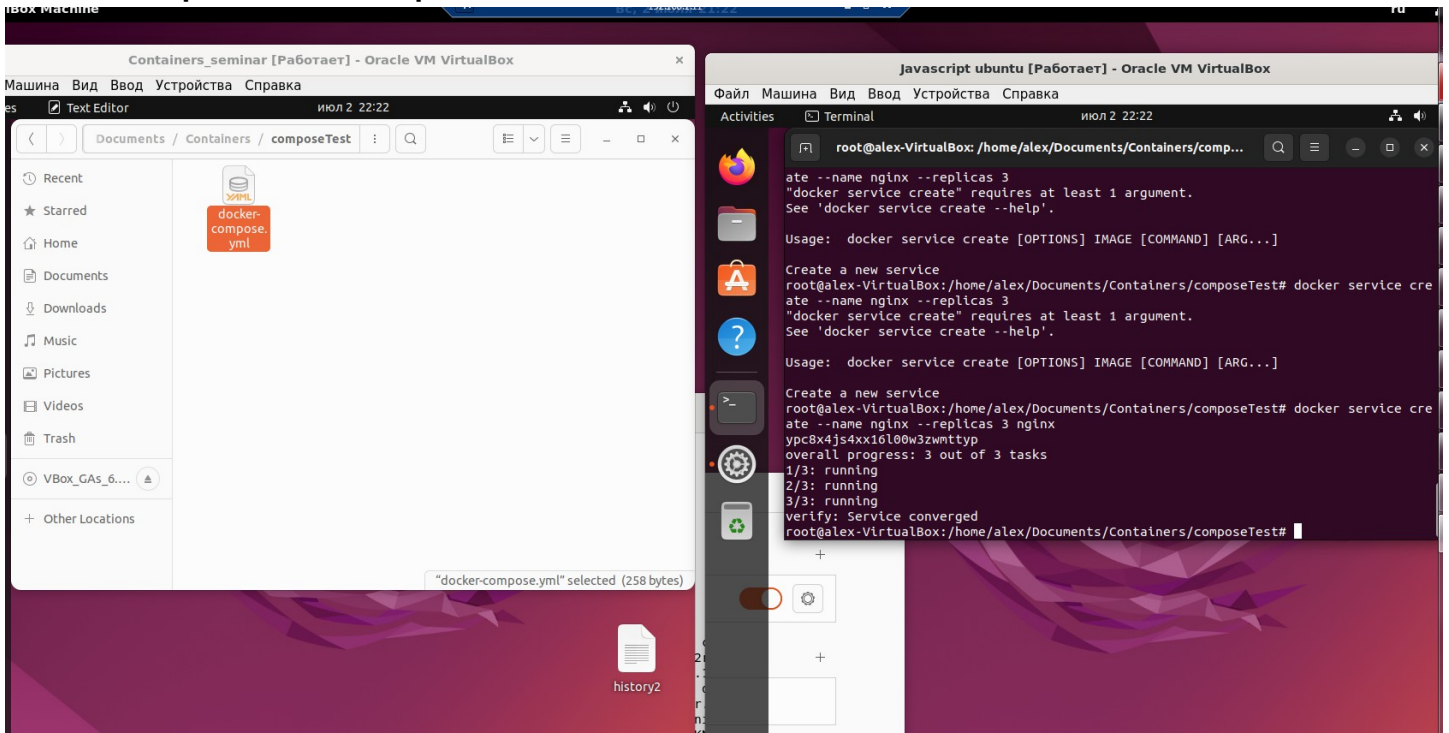
Подключим третью ноду



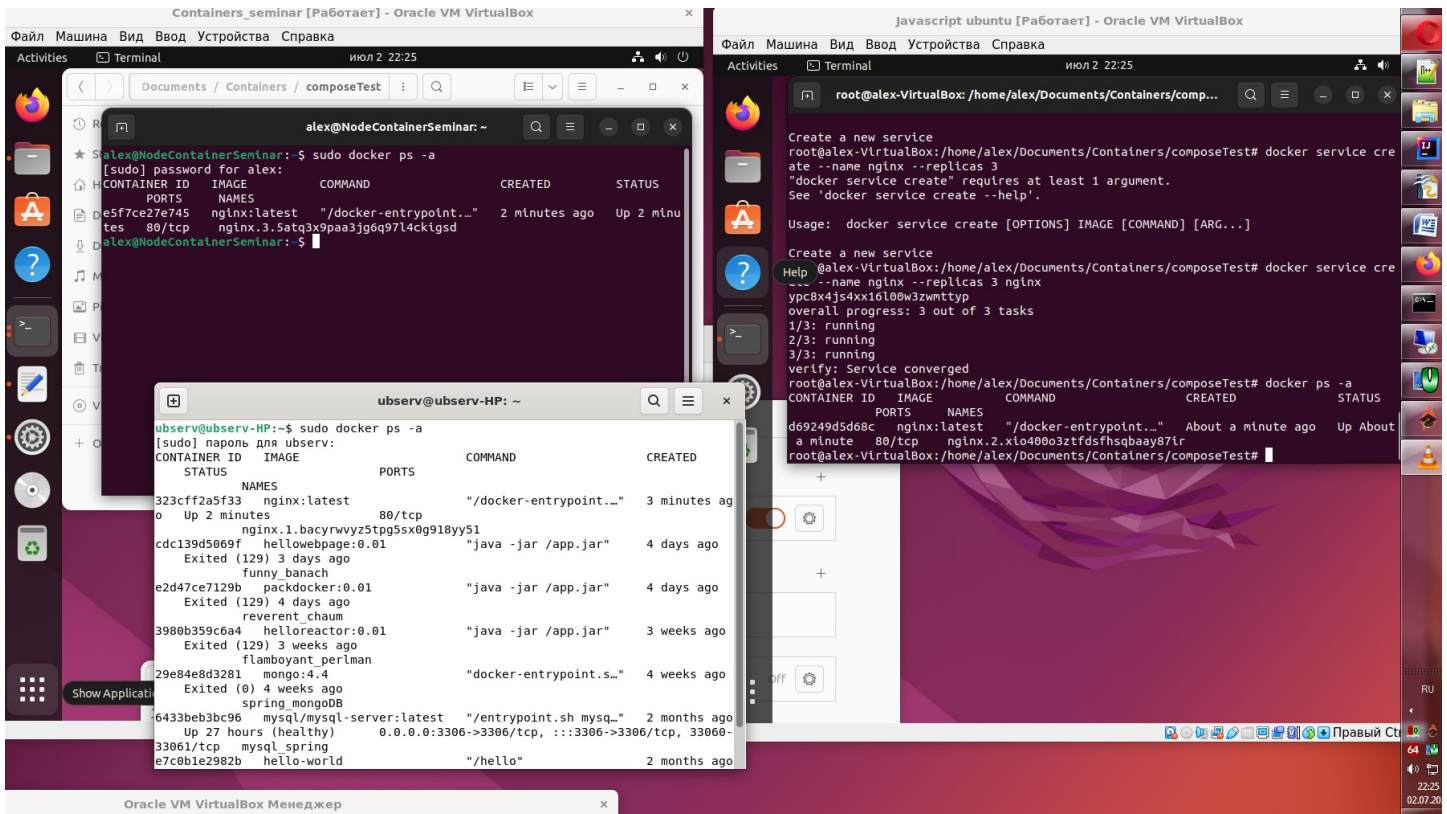
Протэгируем ноды



Распараллел сервис на 3 ноды

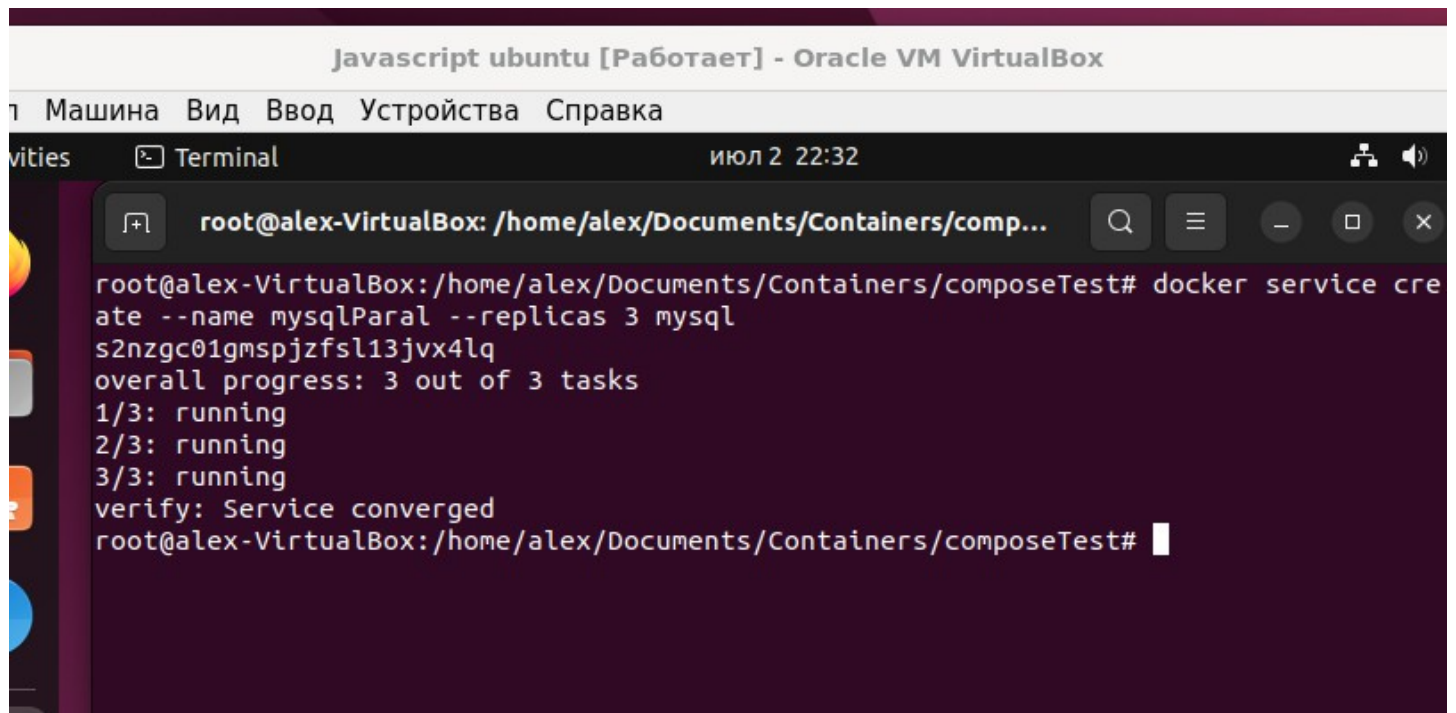


Проверка



Аналогично 2 сервиса

```
root@alex-VirtualBox:/home/alex/Documents/Containers/composeTest# docker service create --name mysqlParal --replicas 3 mysql
```



The screenshot shows a terminal window titled "Javascript ubuntu [Работает] - Oracle VM VirtualBox". The terminal interface includes a menu bar with "Машина", "Вид", "Ввод", "Устройства", and "Справка". Below the menu bar, the terminal title bar reads "Terminal" and the system clock shows "июл 2 22:32". The terminal content shows the command `docker service create --name mysqlParal --replicas 3 mysql` being executed. The output indicates that the service was created successfully, showing the service ID `s2nzgc01gmnpjzfs13jvx4lq`, the progress "overall progress: 3 out of 3 tasks", and the status of each replica: "1/3: running", "2/3: running", and "3/3: running". The terminal concludes with "verify: Service converged" and returns to the prompt.

```
root@alex-VirtualBox:/home/alex/Documents/Containers/composeTest# docker service create --name mysqlParal --replicas 3 mysql
s2nzgc01gmnpjzfs13jvx4lq
overall progress: 3 out of 3 tasks
1/3: running
2/3: running
3/3: running
verify: Service converged
root@alex-VirtualBox:/home/alex/Documents/Containers/composeTest#
```

```
root@alex-VirtualBox:/home/alex/Documents/Containers/composeTest# docker service create --name HWPParal --replicas 3 helloworld
```

создать 2 ДК-файла, в которых будут описываться сервисы

```
1 version: '3.9'
2
3 services:
4   pyth:
5     image: python:latest
6   nginx:
7     image: nginx:latest
```

```
version: '3.9'
services:
  pyth:
    image: python:latest
  alpine:
    image: alpine:latest
```

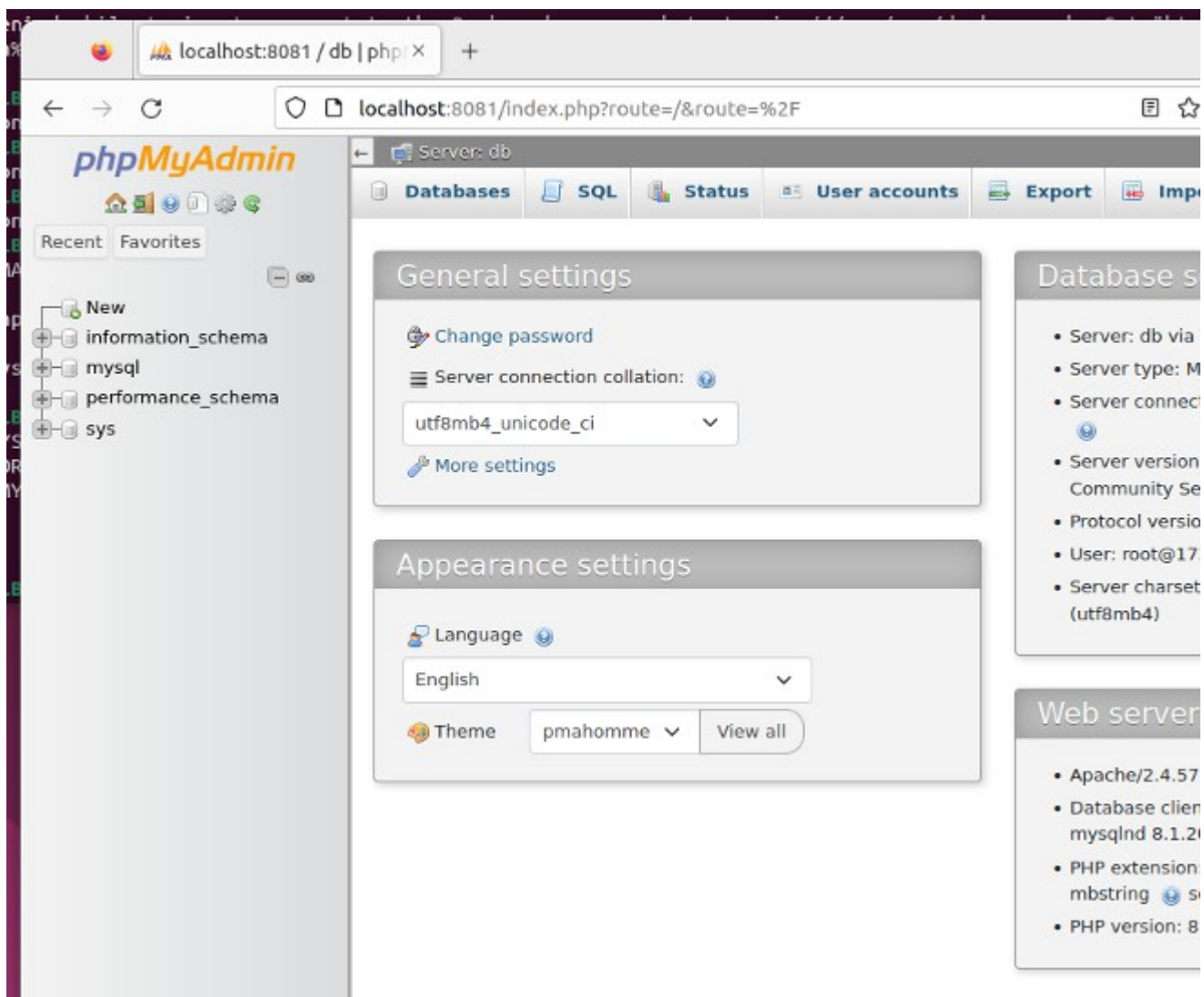
Повторить для окружения lab

```
alex@NodeContainerSeminar:~/Documents/Containers/composeTest$ touch docker-compose.yml
```

```
services:
  db:
    image: mysql:latest
    networks:
      - db-net
    environment:
      - MYSQL_ROOT_PASSWORD=12345
  watcherDB:
    image: phpmyadmin:latest
    networks:
      - db-net
    ports:
      - 8081:80
networks:
  db-net:
```

```
sudo apt install docker-compose
```

```
$ sudo docker-compose up -d
```



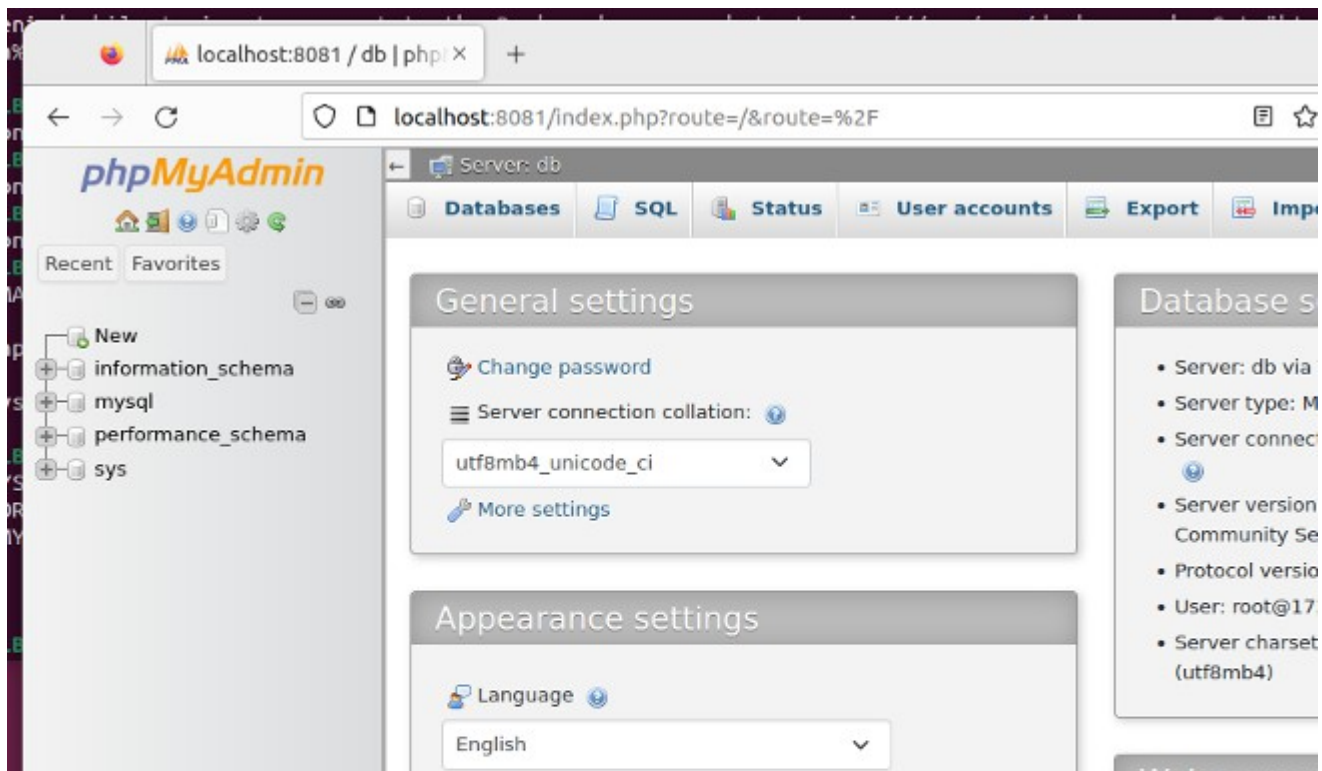
Повторить для окружения dev

```
ubserv@ubserv-HP:~$ touch docker-compose.yml
```

```
services:
  db:
    image: mysql:latest
    networks:
      - db-net
    environment:
      - MYSQL_ROOT_PASSWORD=12345
  watcherDB:
    image: phpmyadmin:latest
    networks:
      - db-net
    ports:
      - 8081:80
networks:
  db-net:
```

```
sudo apt install docker-compose
```

```
$ sudo docker-compose up -d
```



3 сервиса в каждом окружении
уже созданы o rly