|  |  |  |  |
| --- | --- | --- | --- |
| **Студент:** | Ханашевич Кирилл Андреевич | **Допуск:** |  |
| **Отчет по лабораторной работе №1**  «Установка гипервизора и создание виртуальных машин» | | | |
| Дата выполнения |  | Дата защиты |  |
| Оценка |  | Подпись |  |

Цель работы:

|  |
| --- |
| Получить навыки по работе с гипервизорами, развертыванию виртуальной инфраструктуре, удаленной работе с виртуальными машинами |

Задачи работы:

|  |
| --- |
|  |

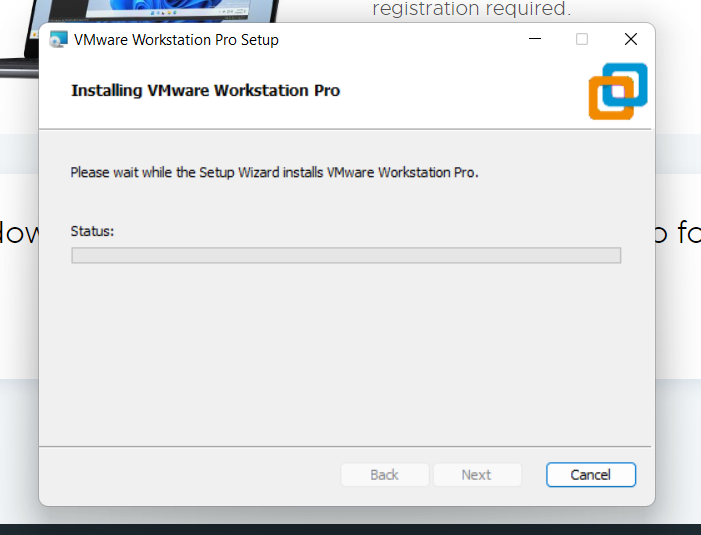
Краткий конспект теоретической части:

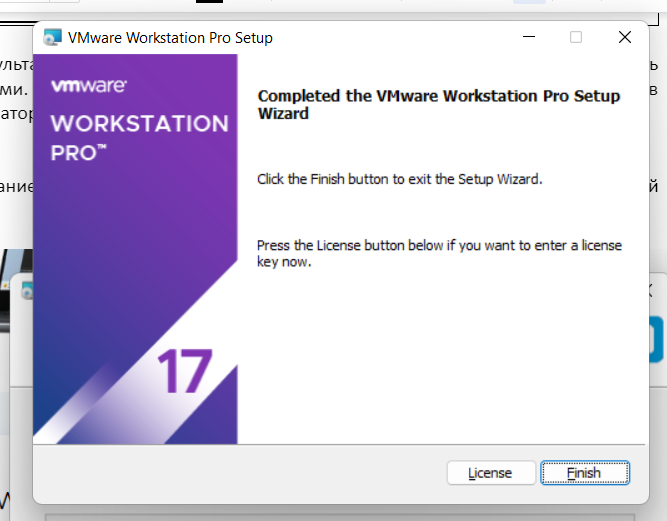
|  |
| --- |
| Гипервизор: программа на пк для создания “виртуальных машин”  Виртуальная машина:  Операционная ситема:  SSH:  Удаленный доступ:  Алгоритмы RSA и ed25519: |

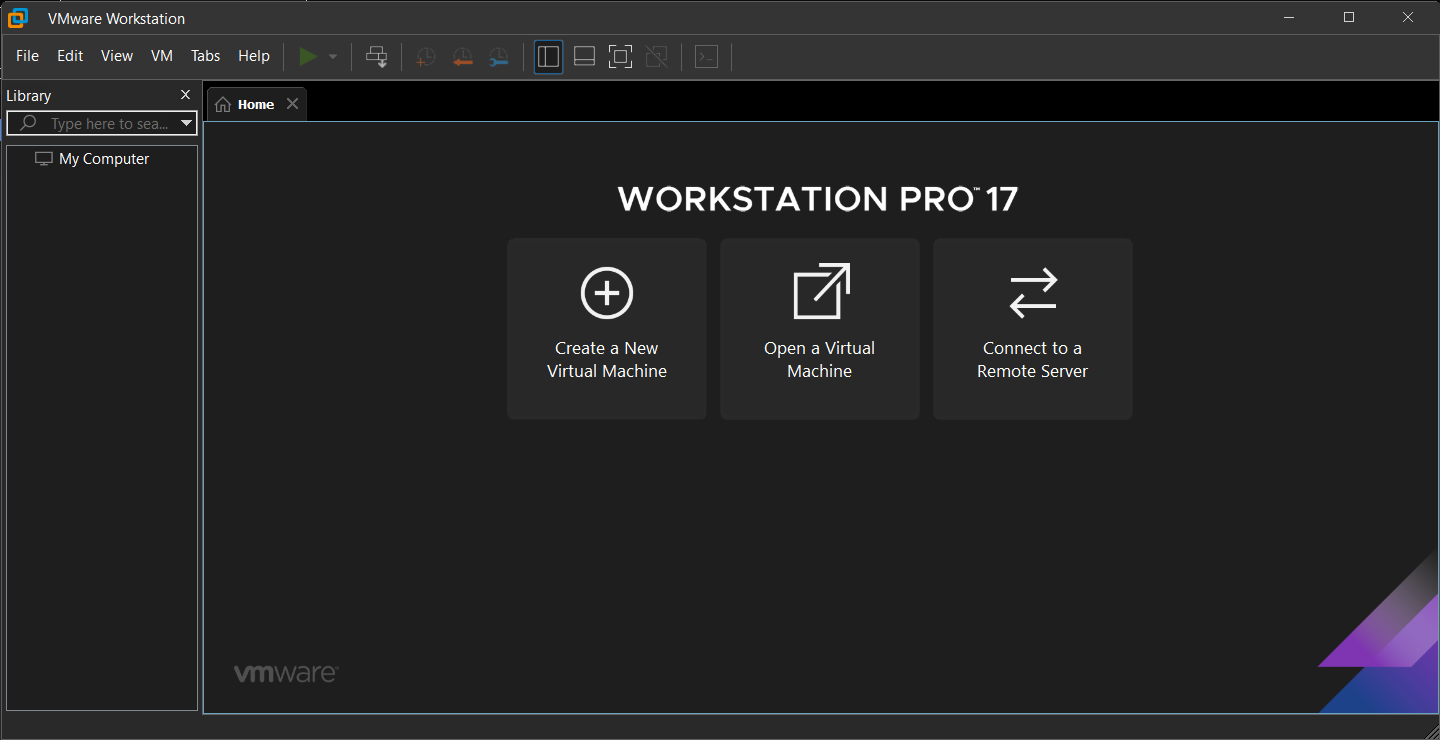
Результаты выполнения всех заданий следует наглядно подтверждать скриншотами. Все операции следует выполнять от учетной записи студента без прав администратора.

Задание 1. Установить гипервизор, показать процесс установки и стартовый экран.

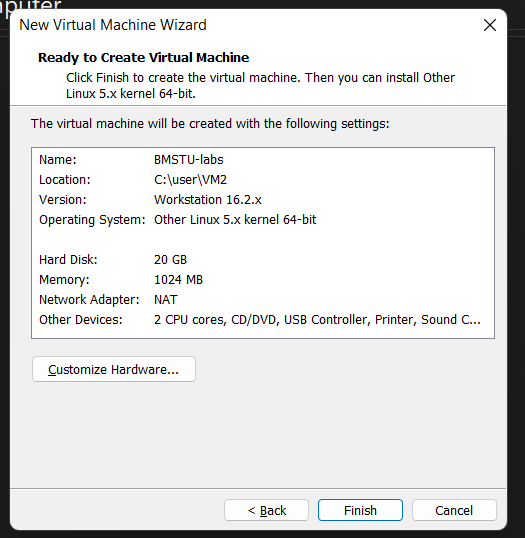
exe-файл worksation pro был скачан с официального сайта производителя, после чего был установлен, путем запуска, загруженного exe-файла

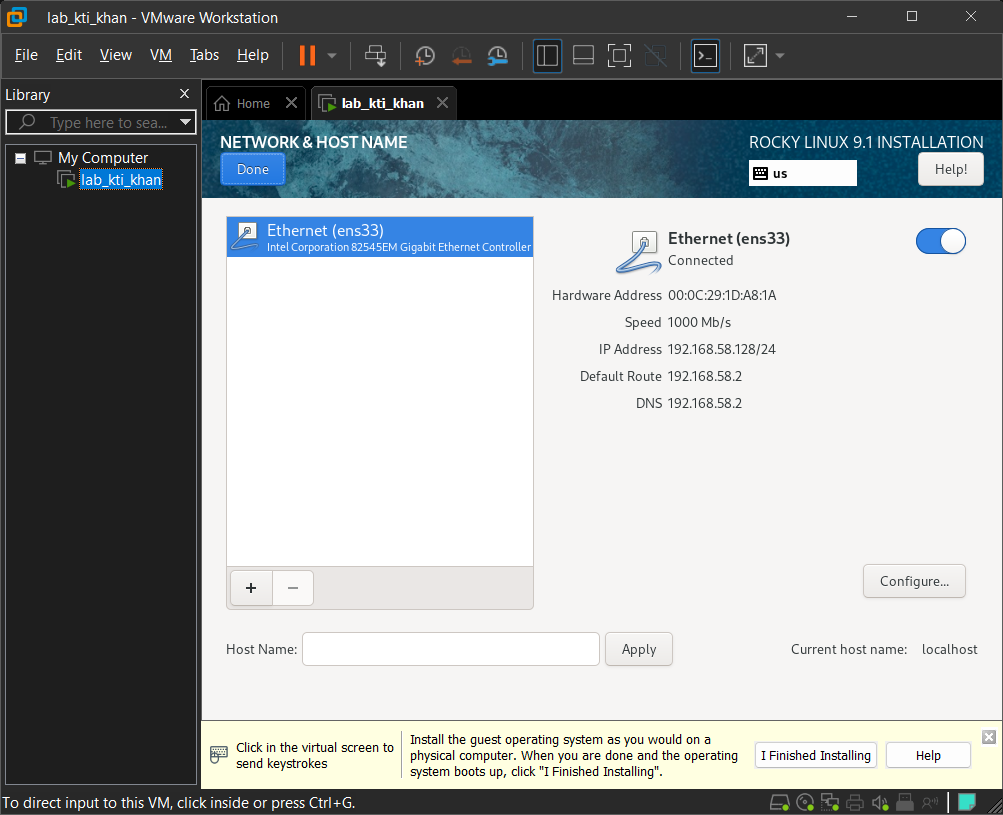


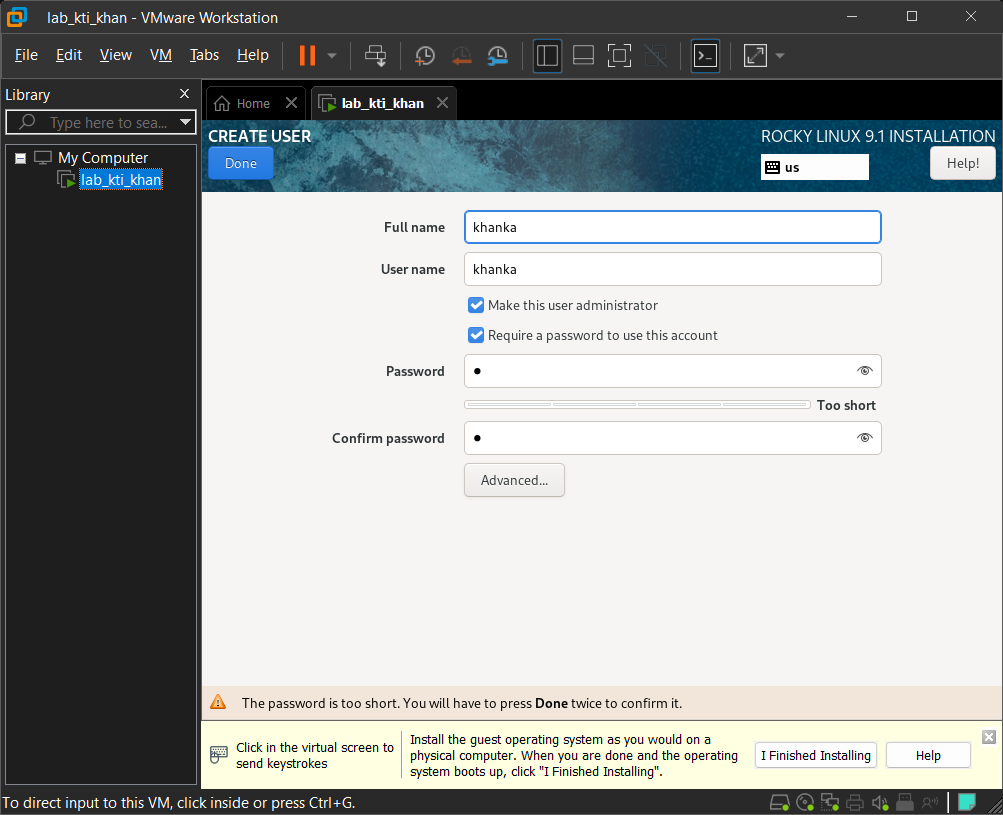


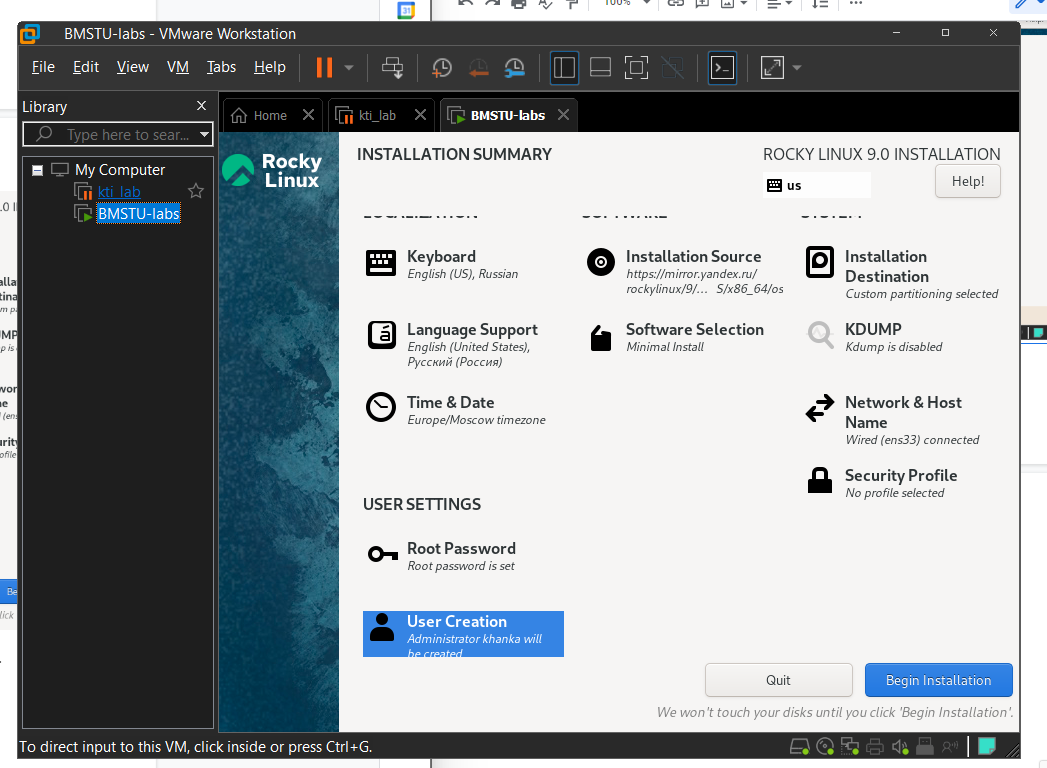


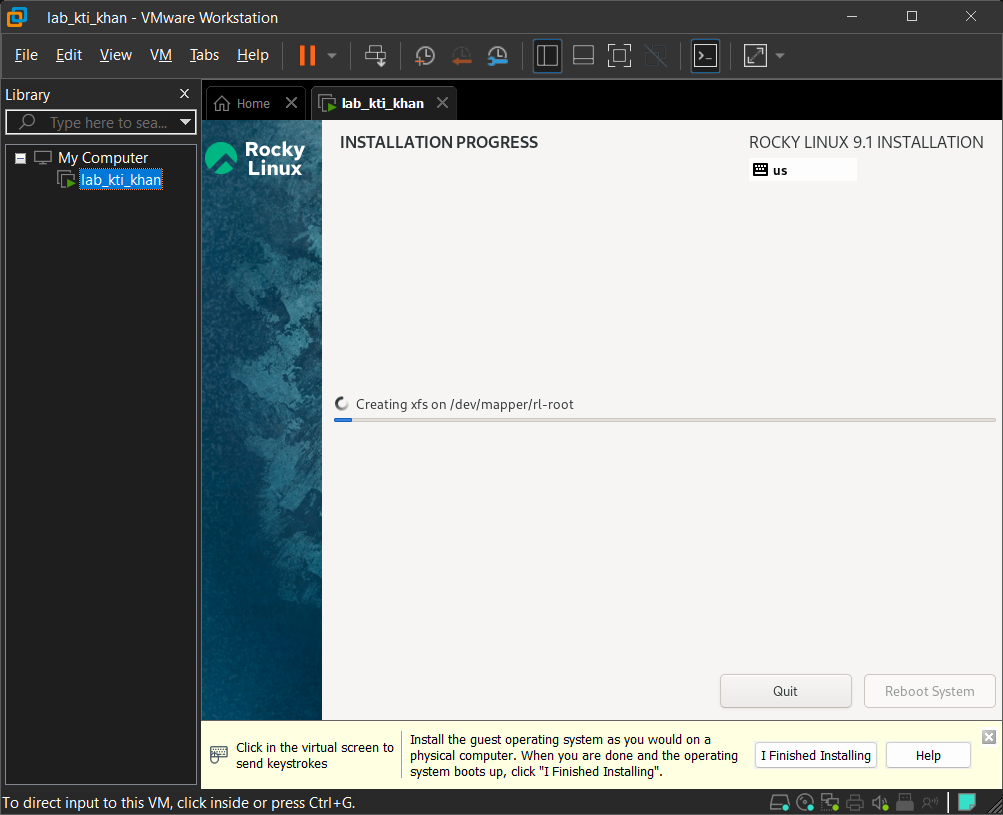
Задание 2. Создать виртуальную машину. Установить на нее операционную систему. Создать учетную запись студента (содержащую фамилию и инициалы). Показать настройки виртуальной машины. Отдельно показать сетевые настройки, IP-адрес.

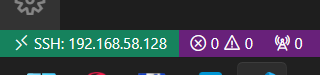
  
Перепроверим настройки виртуальной машины



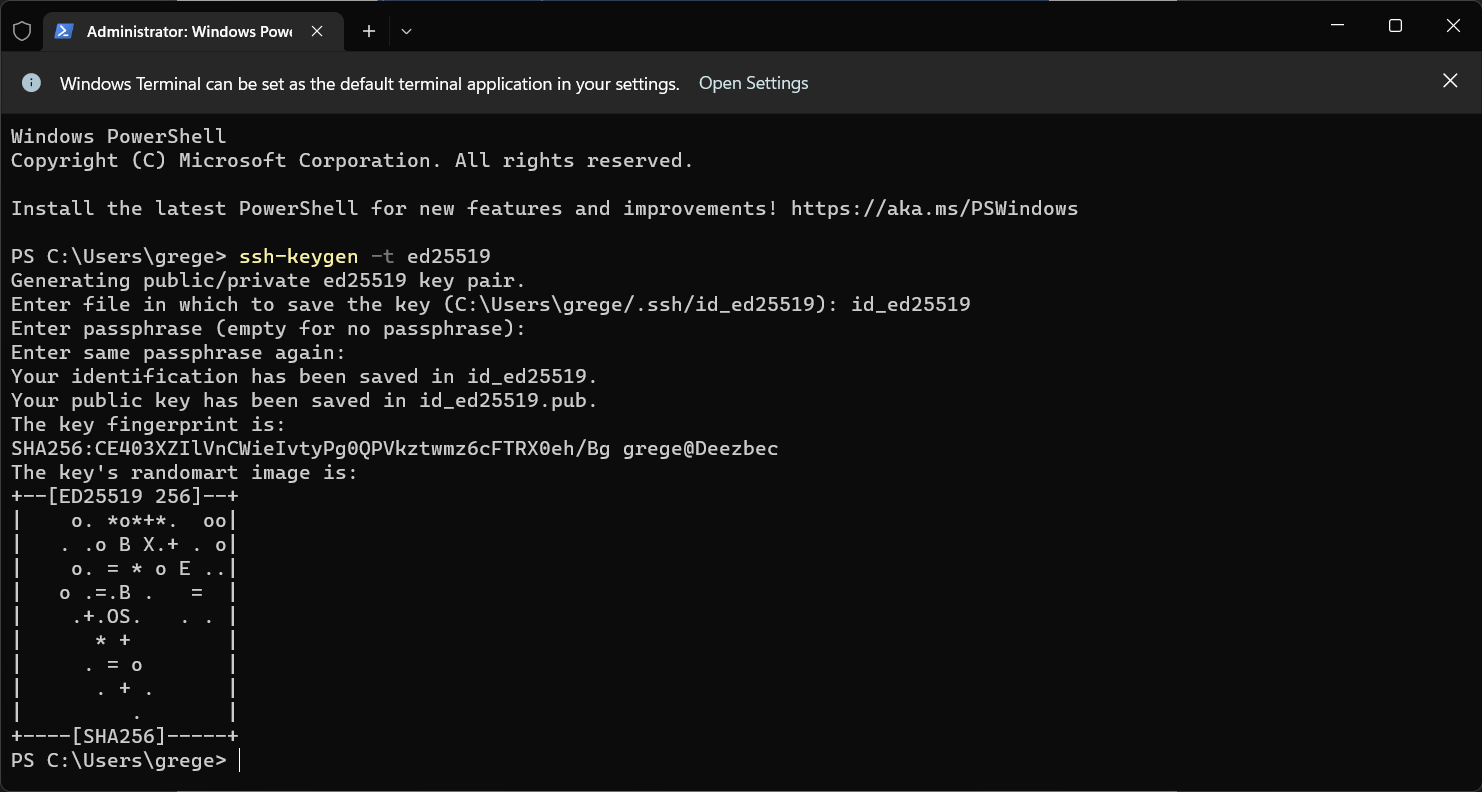


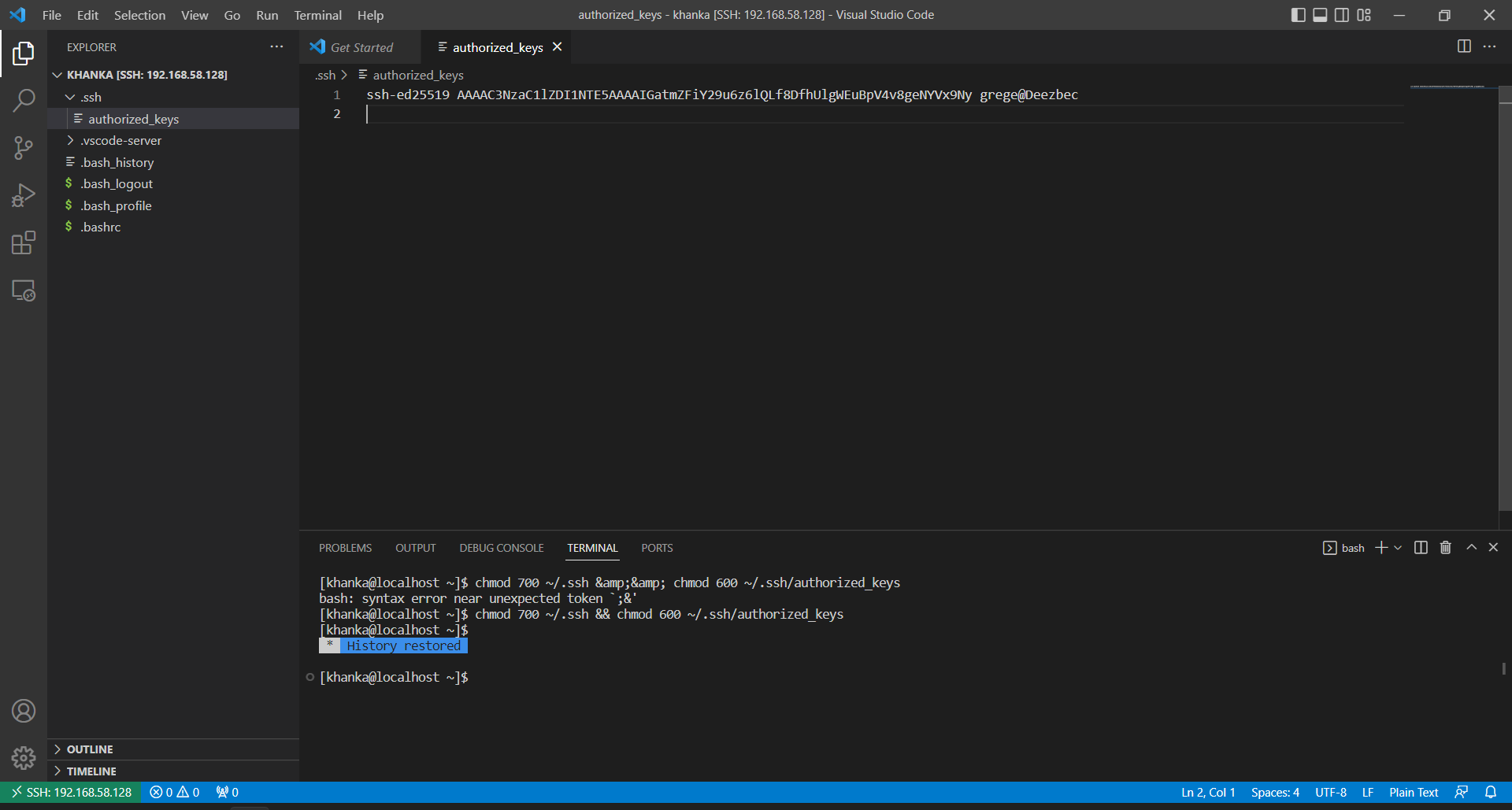


  
   
 sudo dnf install tar



Задание 3. Создать пару ключей. Продемонстрировать процесс доставки открытого ключа на виртуальную машину. Подключиться к ней по ssh, используя аутентификацию по ключам.





Контрольные вопросы:

1. Что такое гипервизор?

2. Что такое виртуальная машина?

3. Что такое операционная система?

4. Что такое асимметричное шифрование?

5. Какие права необходимо настроить на папку .ssh и файл authorized\_keys?   
chmod 700 and chmod 600

|  |  |  |  |
| --- | --- | --- | --- |
| **Студент:** |  | **Допуск:** |  |
| **Отчет по лабораторной работе №2**  «Установка python и запуск веб-сервера flask» | | | |
| Дата выполнения |  | Дата защиты |  |
| Оценка |  | Подпись |  |

Цель работы:

|  |
| --- |
| Получить навыки по установке и сборке из исходного кода пакетов в Linux, запуску веб-приложений flask |

Задачи работы:

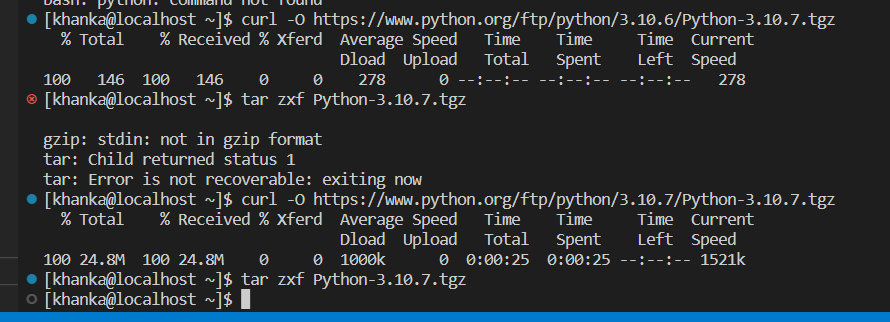
|  |
| --- |
|  |

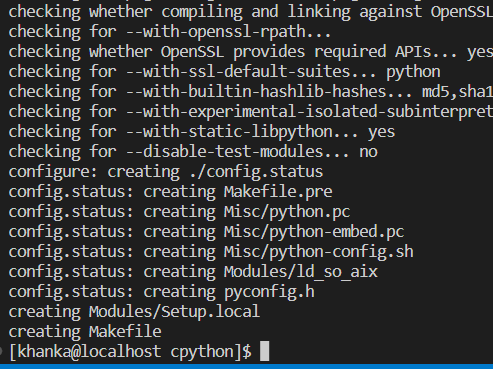
Краткий конспект теоретической части:

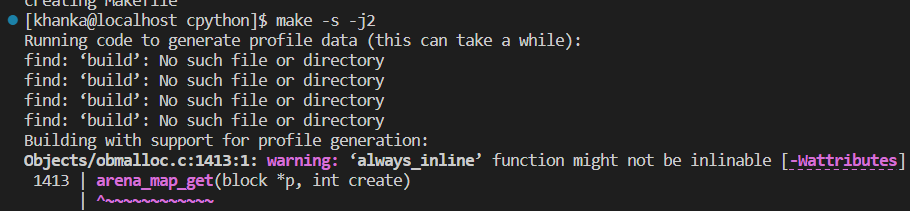
|  |
| --- |
| Python:  Виртуальное окружение (venv):  Установка и сборка пакетов в Linux:  Установка модулей Python:  Настройка фаерволла в Linux:  Запуск, остановка и диагностика сервисов в Linux: |

Результаты выполнения всех заданий следует наглядно подтверждать скриншотами. Все операции следует выполнять от учетной записи студента без прав администратора.

Задание 1. Загрузить исходный код последней стабильной версии Python. Продемонстрировать этапы сборки и установки. Показать версию установленного Python.

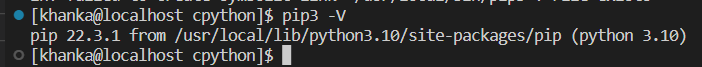
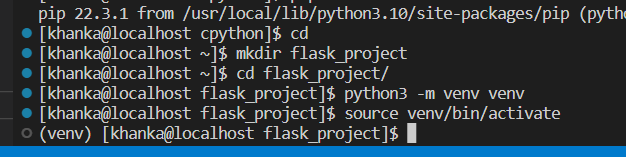


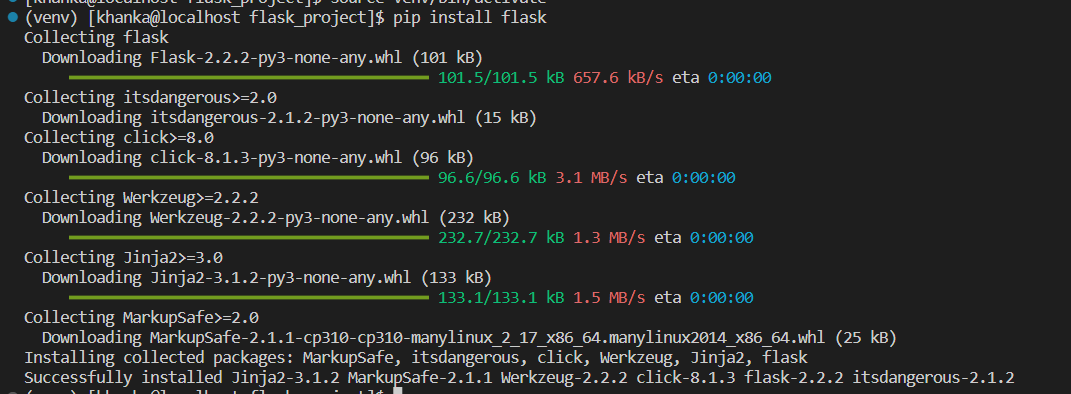


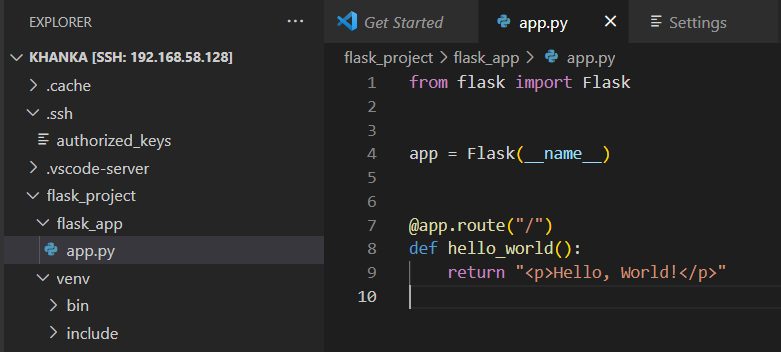
  
  
Для того, чтобы собранный python был доступен на системном уровне, необходимо его установить командой

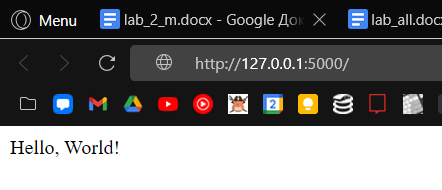
sudo make -s -j2 install  
  
  
sudo ln -s pip3.10 /usr/local/bin/pip3

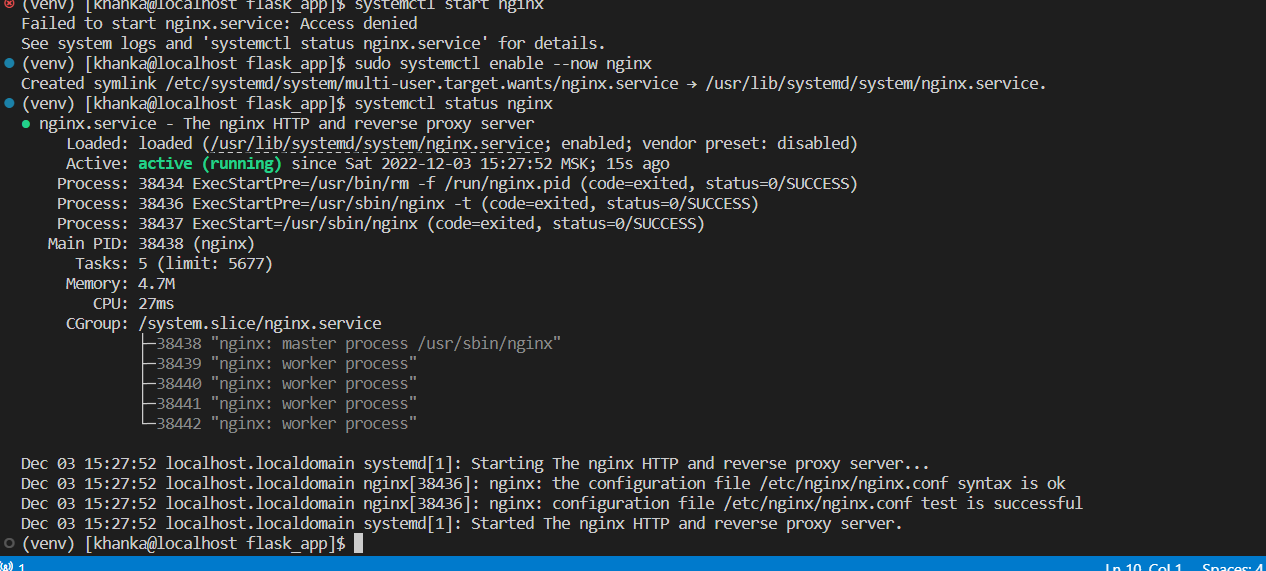
pip3 -V

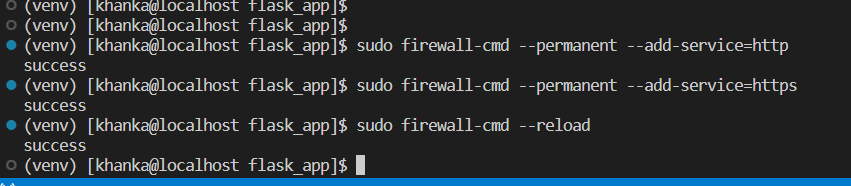
  
  


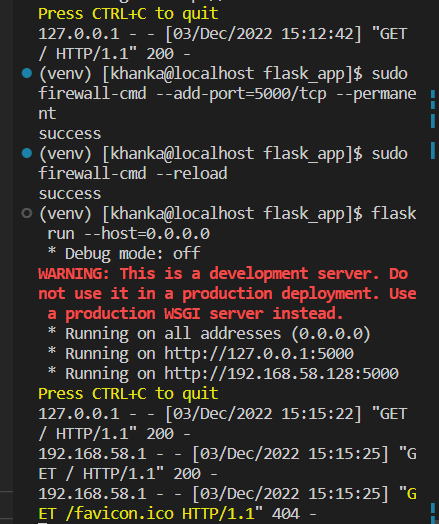
Задание 2. Установить необходимые модули и развернуть flask-приложение с функционалом авторизации. Для хранения пользователей использовать СУБД SQLite3.  
  


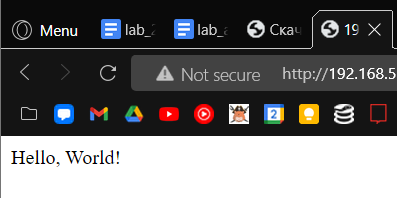


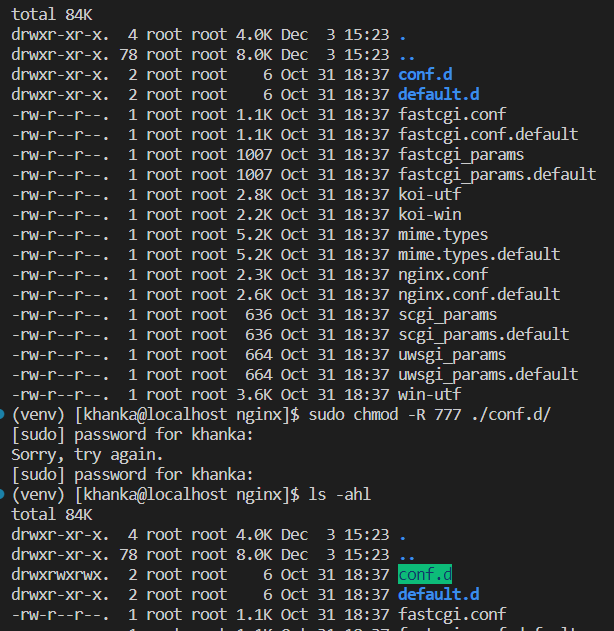


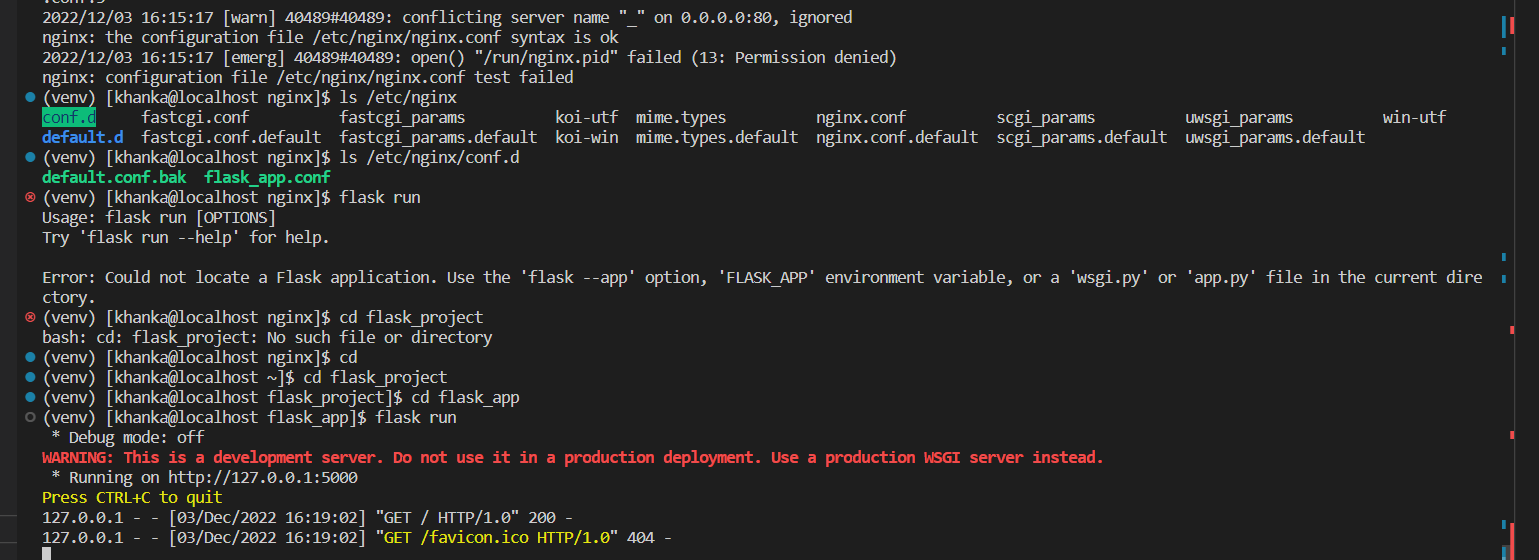


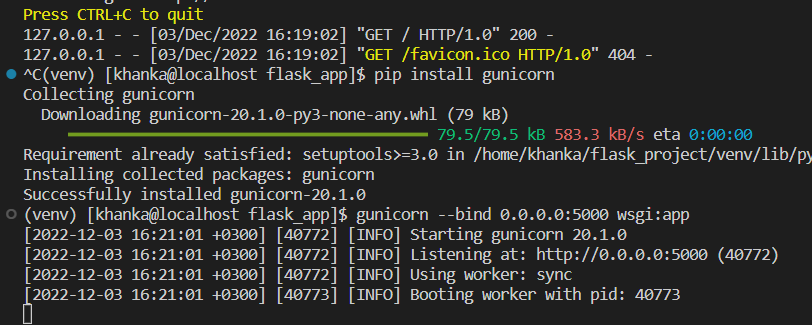






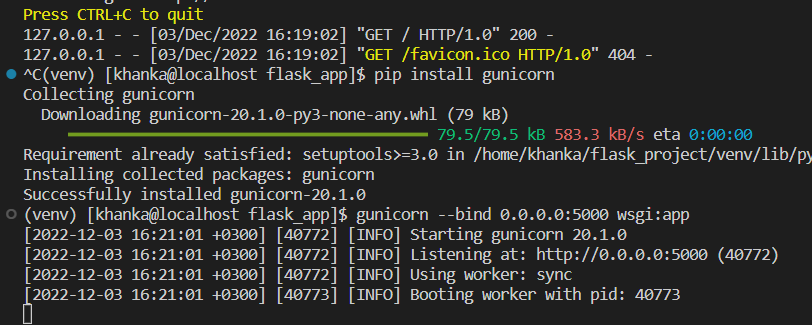


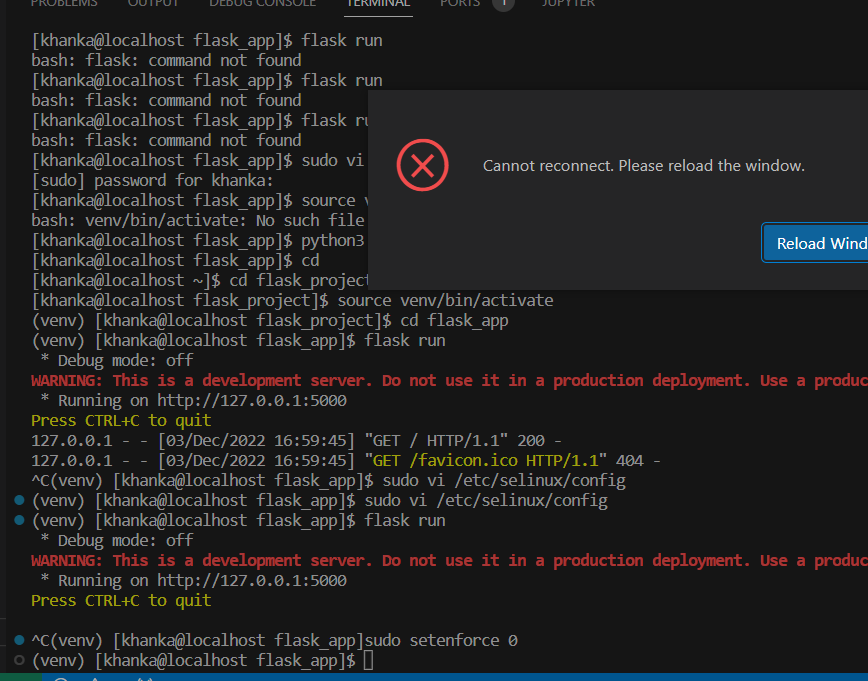
  


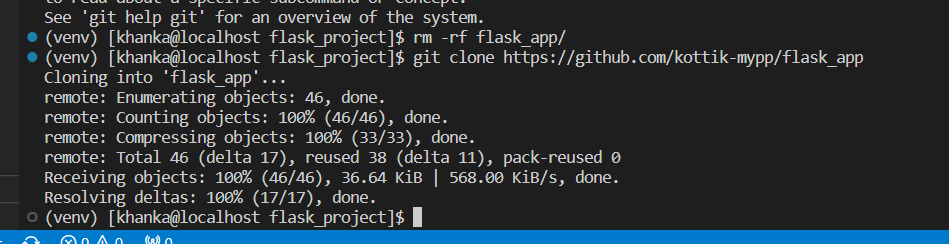


Задание 3. Установить необходимые модули для запуска flask-приложения при помощи gunicorn.

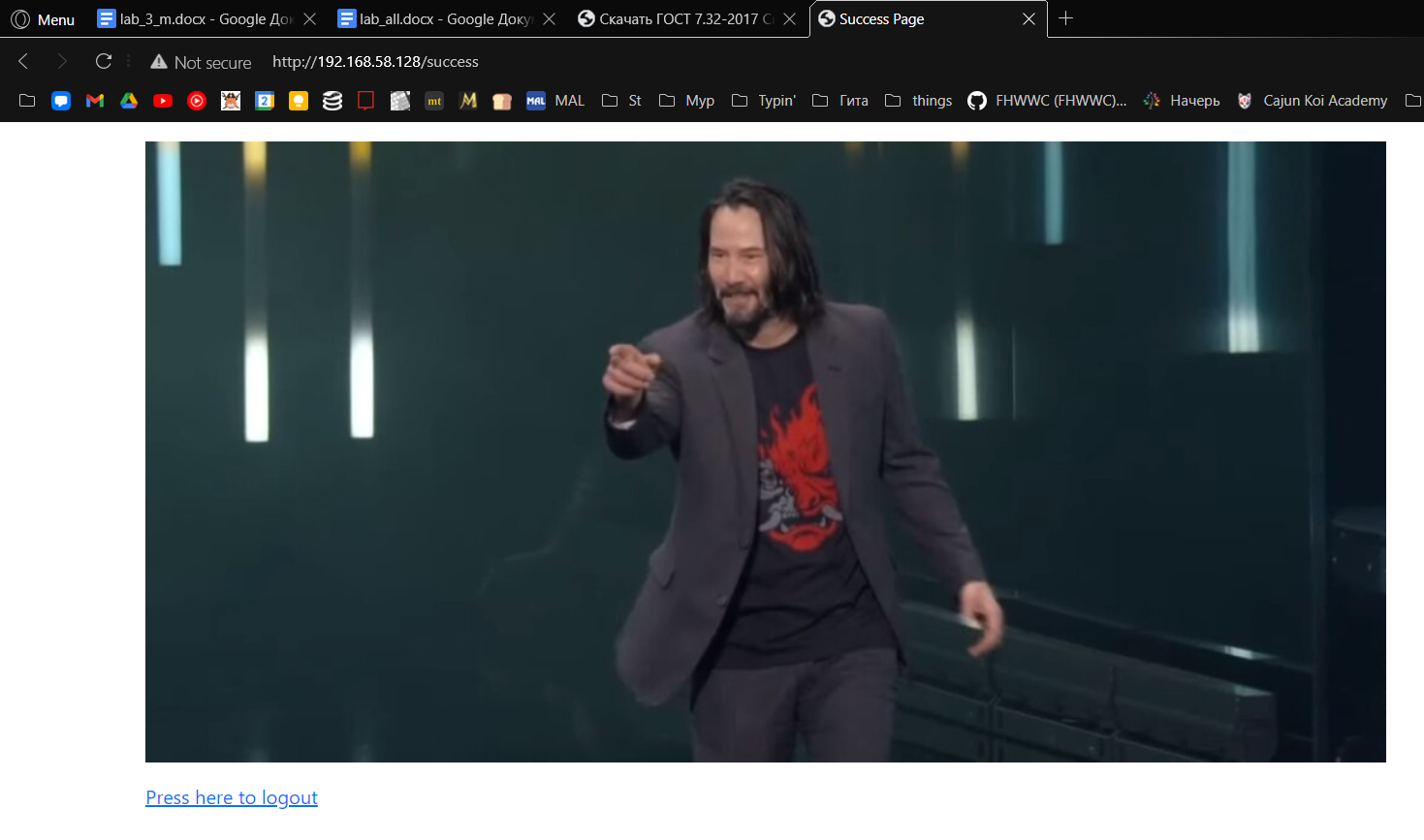
Задание 4. Установить nginx из репозитория разработчиков. Сконфигурировать его так, чтобы он проксировал http-запросы с порта 80 запущенному приложению.







Задание 5. Настроить системные сервис и сокет для автоматического запуска gunicorn и nginx.



Контрольные вопросы:

1. Что такое пакетный менеджер?

2. Почему возможен вызов программы по ее названию без указания полного пути к исполняемому файлу?

3. Какие порты называются открытыми и прослушиваемыми?

4. Что такое демон и сервис в контексте Linux?

5. Чем различаются файловый и сетевой сокеты?

|  |  |  |  |
| --- | --- | --- | --- |
| **Студент:** |  | **Допуск:** |  |
| **Отчет по лабораторной работе №3**  «Выпуск TLS-сертификатов и работа с git-репозиториями» | | | |
| Дата выполнения |  | Дата защиты |  |
| Оценка |  | Подпись |  |

Цель работы:

|  |
| --- |
| Получить навыки по выпуску сертификатов, работе с git-репозиториями |

Задачи работы:

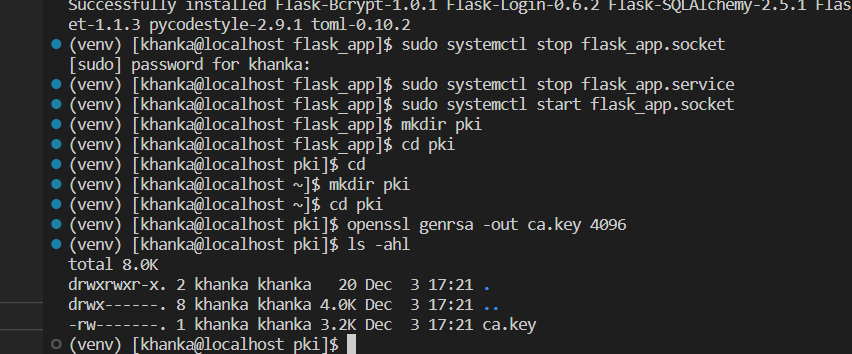
|  |
| --- |
|  |

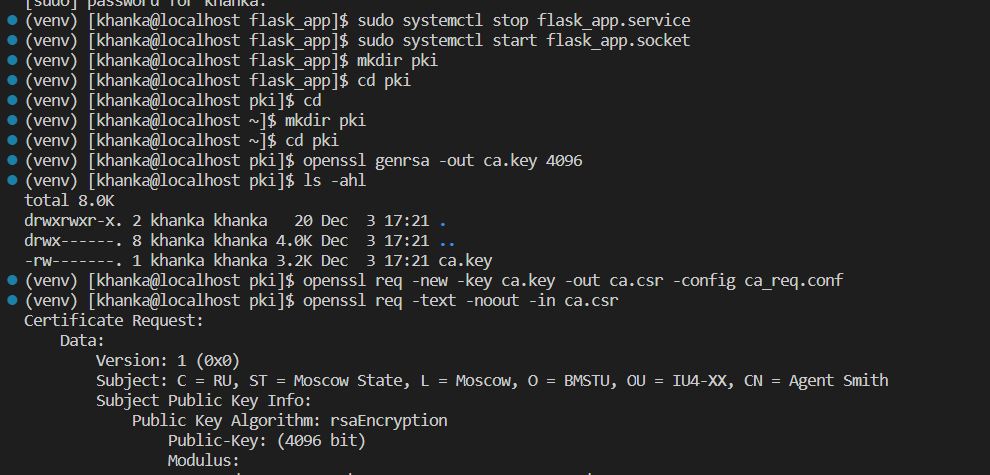
Краткий конспект теоретической части:

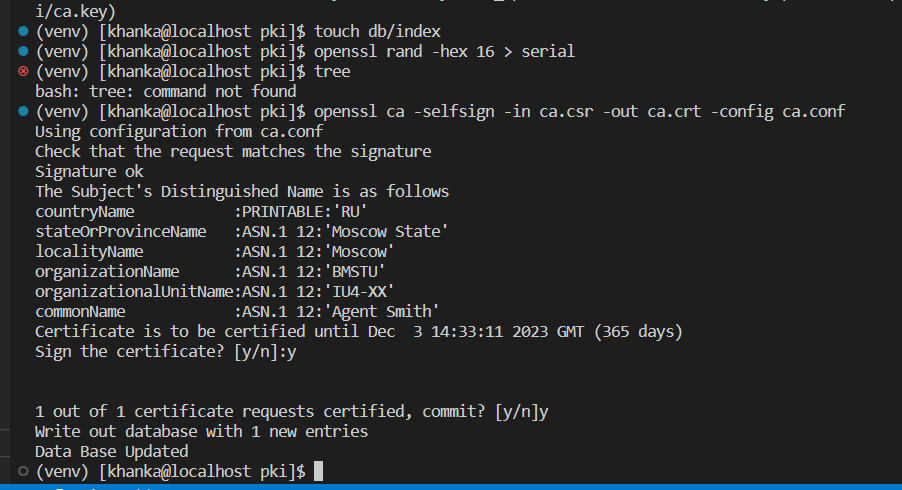
|  |
| --- |
| HTTPS и TLS:  Инфраструктура PKI:  Алгоритм Диффи-Хеллмана:  Репозиторий:  Git:  GitHub: |

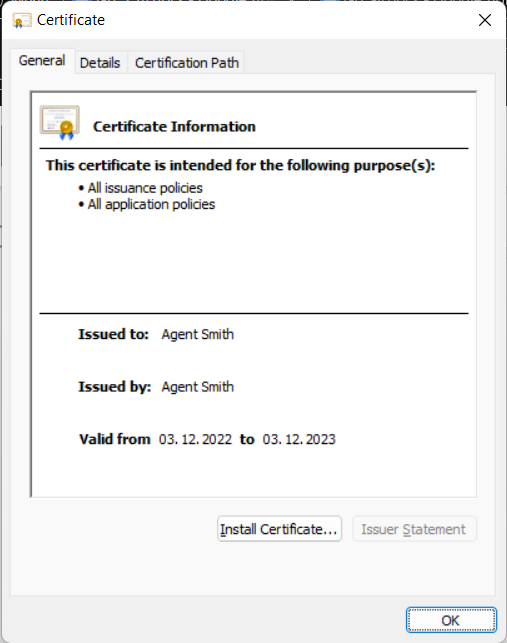
Результаты выполнения всех заданий следует наглядно подтверждать скриншотами. Все операции следует выполнять от учетной записи студента без прав администратора.

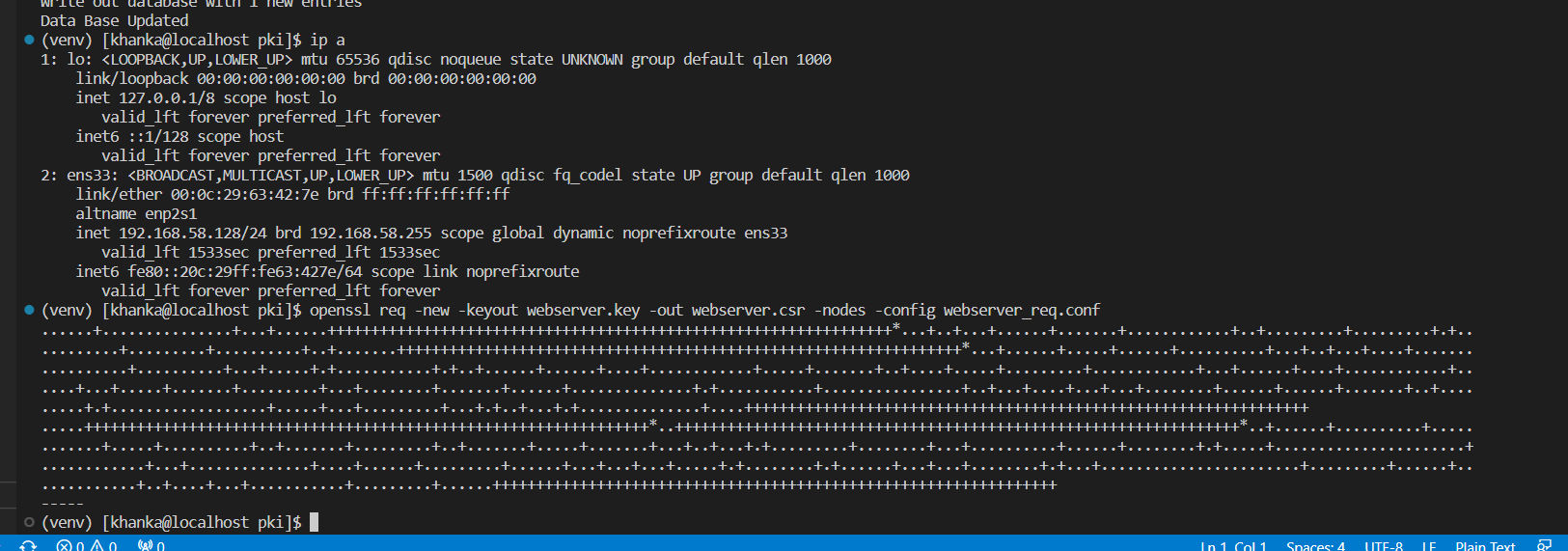
Задание 1. Создать базовую инфраструктуру удостоверяющего центра. В CN указать фамилию и имя студента. Сертификат созданного УЦ загрузить на хост и добавить в доверенные корневые.

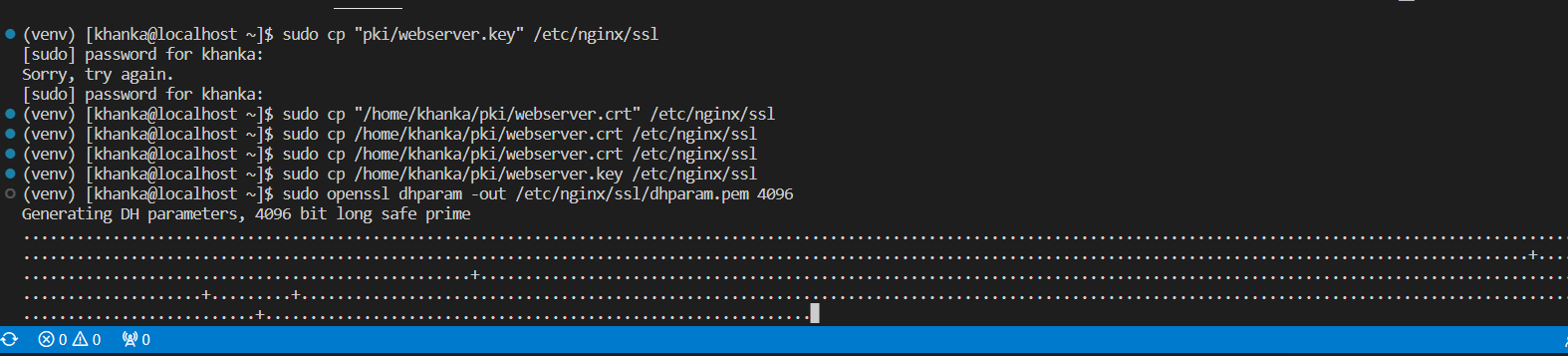


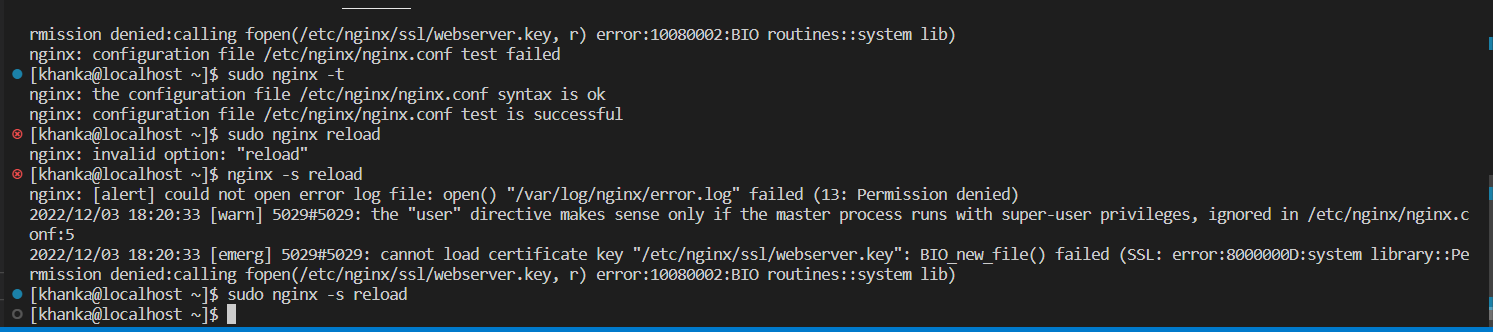


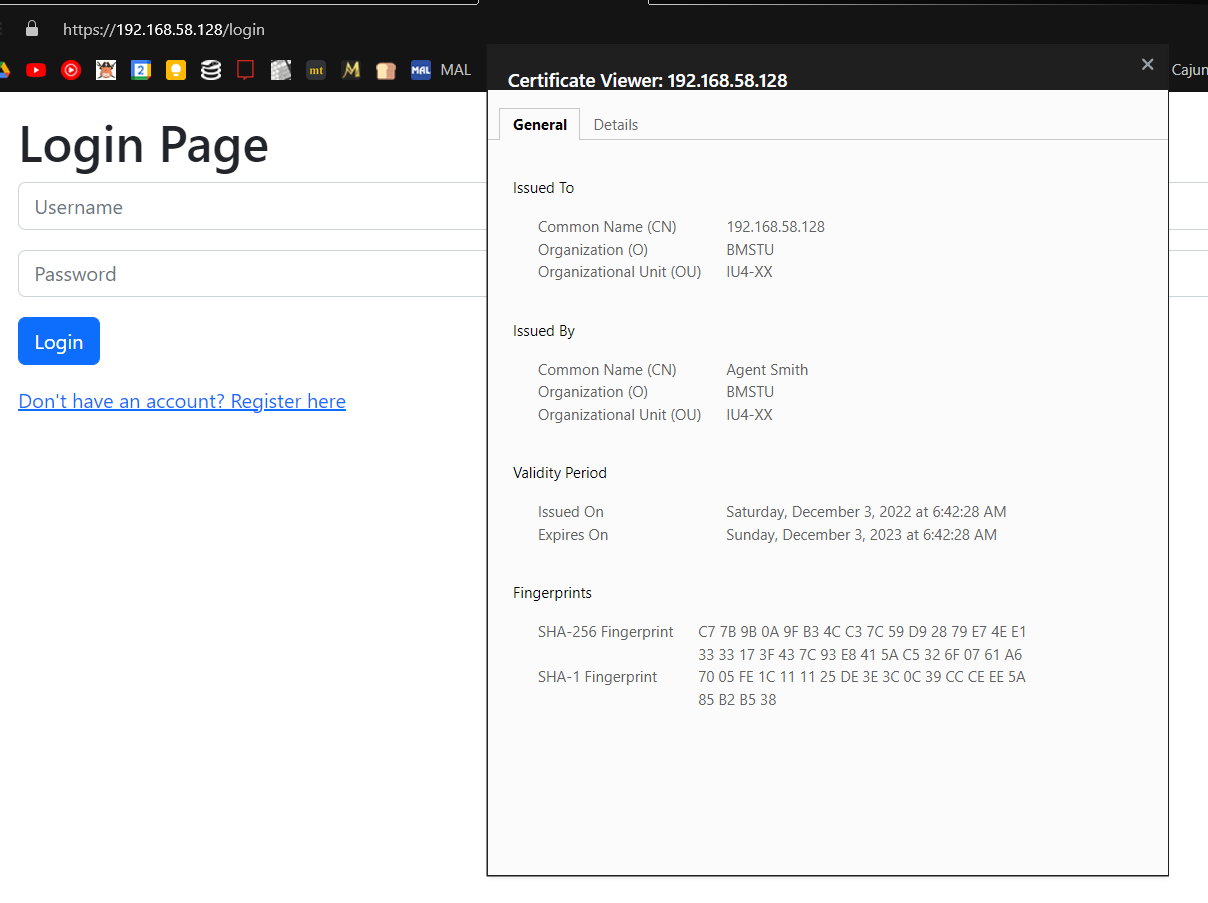








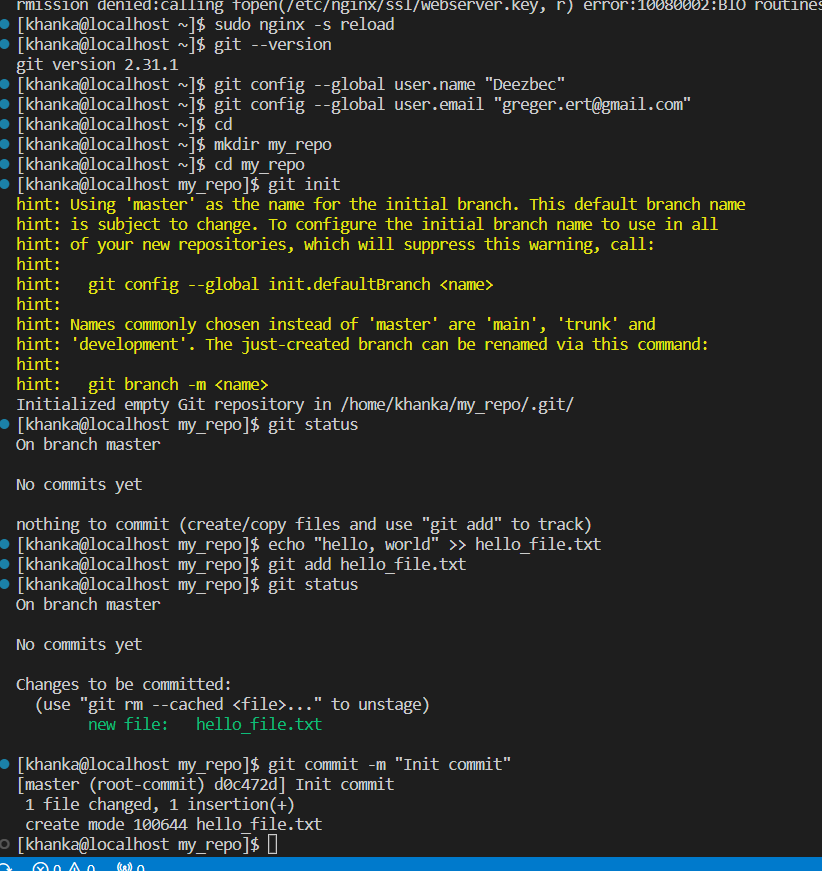
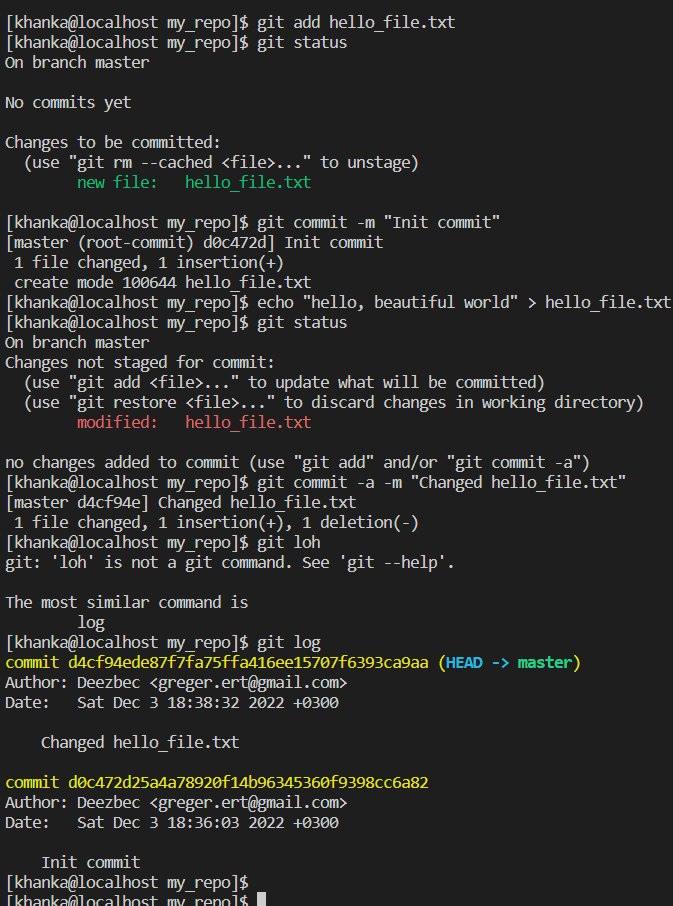




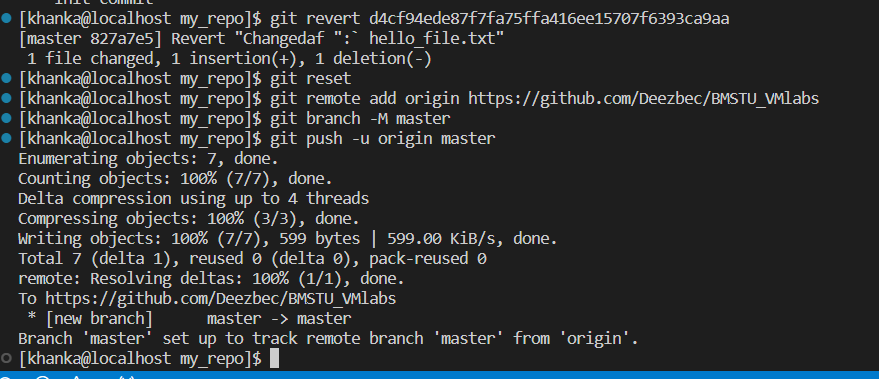
Задание 2. Выпустить сертификат для веб-сервера. Настроить nginx на прослушивание порта 443. Настроить перенаправление с 80 порта на 443. Продемонстрировать его работоспособность.

Задание 3. Создать репозиторий с проектом с авторизацией. Настроить .gitignore так, чтобы файл базы данных не версионировался.

Задание 4. Внести изменение в файлы в репозитории, создать новый коммит, откатить проект к предыдущей версии.

Задание 5. Синхронизировать локальный репозиторий с удаленным на GitLab. Приложить ссылку на репозиторий.



Контрольные вопросы:

1. Что такое PKI?

2. В чем суть принципа forward secrecy?

3. Что такое репозиторий?

4. Каковы базовые команды работы с git-репозиторием?

|  |  |  |  |
| --- | --- | --- | --- |
| **Студент:** |  | **Допуск:** |  |
| **Отчет по лабораторной работе №4**  «Инфраструктура в контейнерах» | | | |
| Дата выполнения |  | Дата защиты |  |
| Оценка |  | Подпись |  |

Цель работы:

|  |
| --- |
| Получить навыки по развертыванию приложений в контейнерах, |

Задачи работы:

|  |
| --- |
|  |

Краткий конспект теоретической части:

|  |
| --- |
| Контейнерная инфраструктура:  Docker:  Открытие портов в контейнерах:  Монтирование томов в контейнерах:  Docker Compose:  Взаимодействие сервисов в docker compose: |

Результаты выполнения всех заданий следует наглядно подтверждать скриншотами. Все операции следует выполнять от учетной записи студента без прав администратора.

Задание 1. Развернуть вторую виртуальную машину. На ней установить docker, docker compose. Запустить контейнер с nginx с кастомной веб-страничкой с текстом «Hello from <student\_name>»

Задание 2. Написать Dockerfile для сборки образа c системными пакетами (gcc g++ musl-dev), необходимыми для установки модулей проекта с авторизацией. Собрать кастомный образ по этому Dockerfile.

Задание 3. Написать файл docker-compose.yaml, в котором описать сервисы nginx и python. В контейнер сервиса python необходимо смонтировать директорию с приложением. Показать работу приложения по 80 порту, логи docker compose с отображением запросов.

Задание 4. Добавить в docker-compose.yaml сервис postgresql. Смонтировать контейнеру сервиса postgresql инициализационный скрипт, в котором создается пользователь, база и таблички. Продемонстрировать работоспособность и логи работы всех трех сервисов.

Задание 5. Настроить перенаправление запросов с первой виртуальной машины на вторую в контейнер с приложением. Показать работы приложения по https, логи с запросами на второй виртуальной машине.

Задание 6. Написать скрипт для автоматического развертывания всей инфраструктуры в контейнерах. Продемонстрировать работу скрипта. Приложить ссылки на репозитории с кодом инфраструктуры и кодом приложения.

Задание 7. Составить структурную схему развернутой на двух виртуальных машинах инфраструктуры.

Контрольные вопросы:

1. Что такое контейнер?

2. Что такое образ контейнера?

3. Как можно смонтировать файлы в контейнер?

4. Каковы преимущества применения контейнерной инфраструктуры?