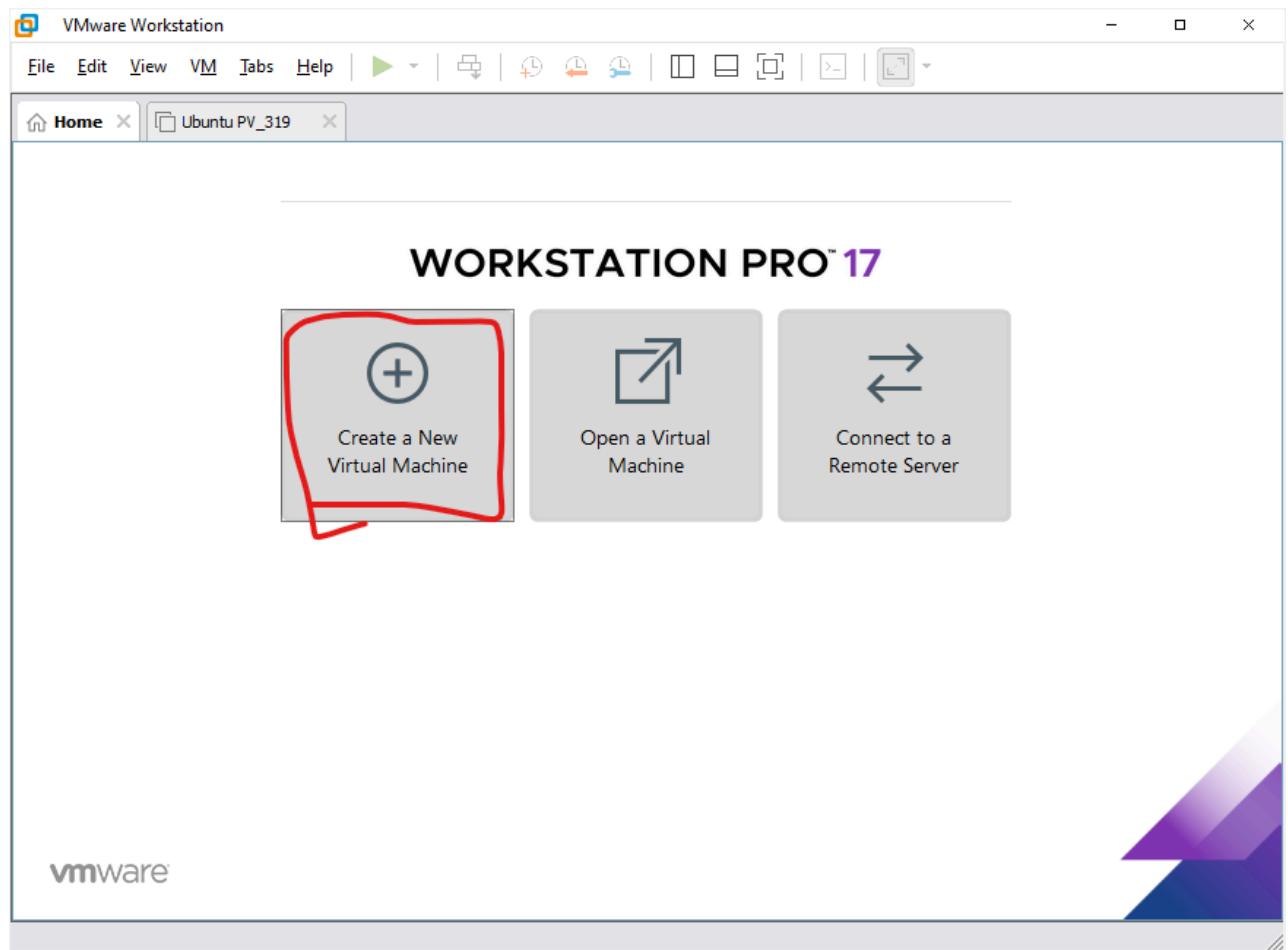
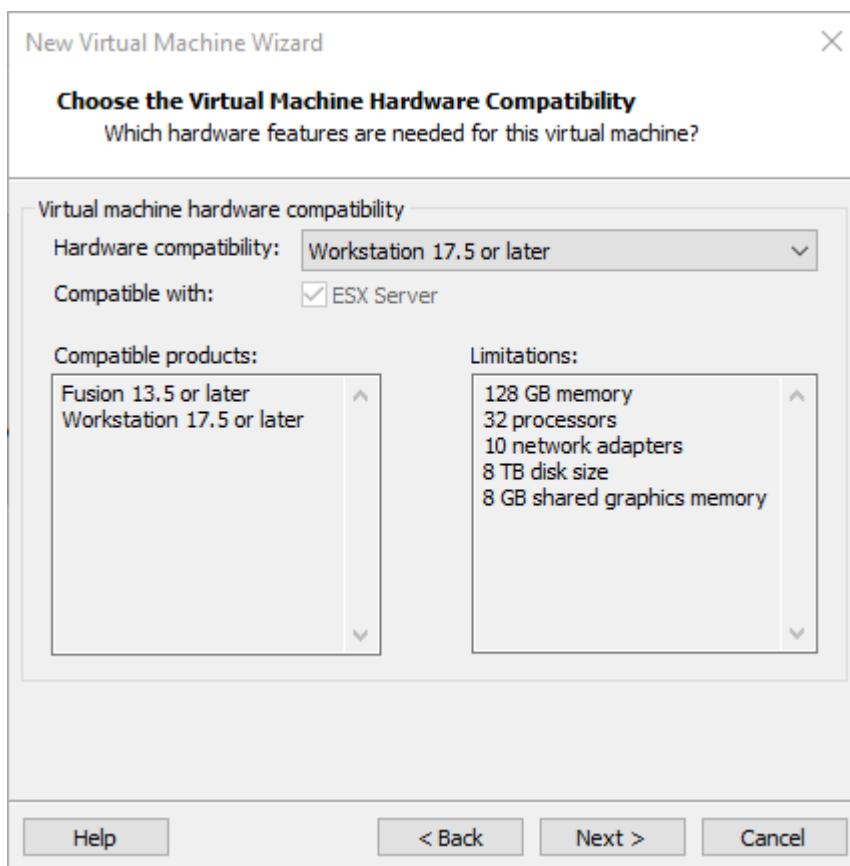
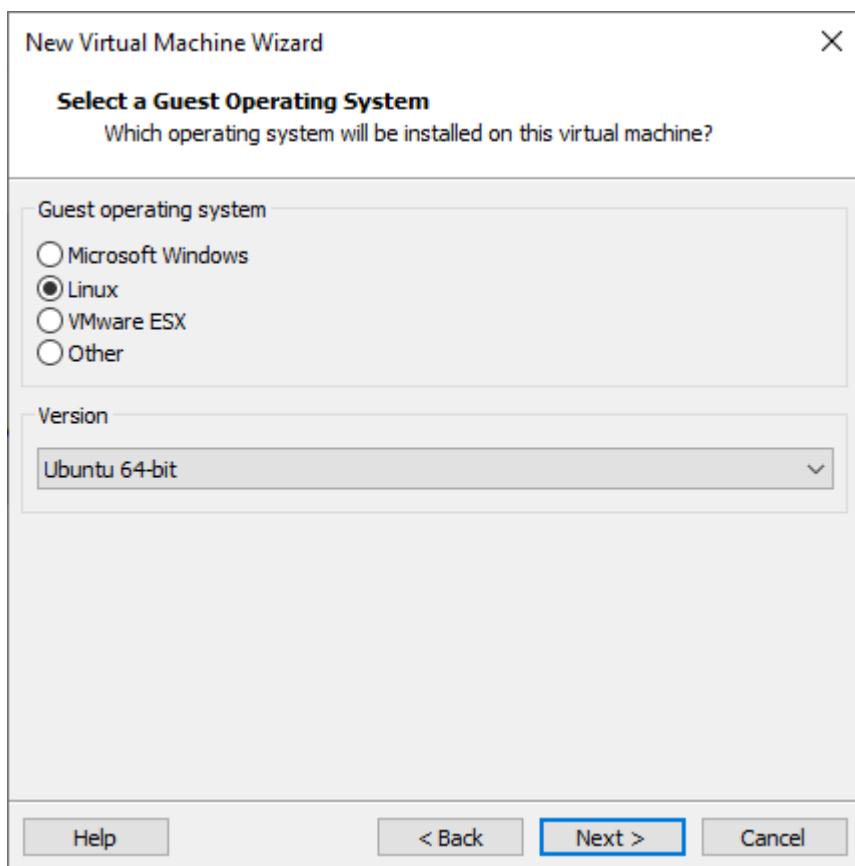
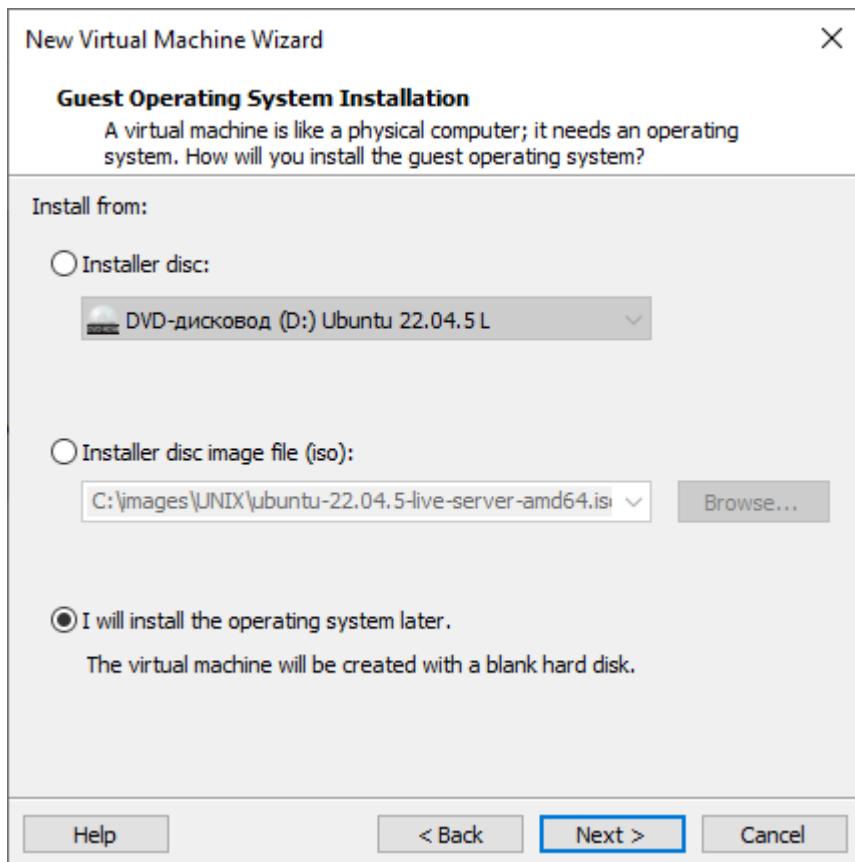
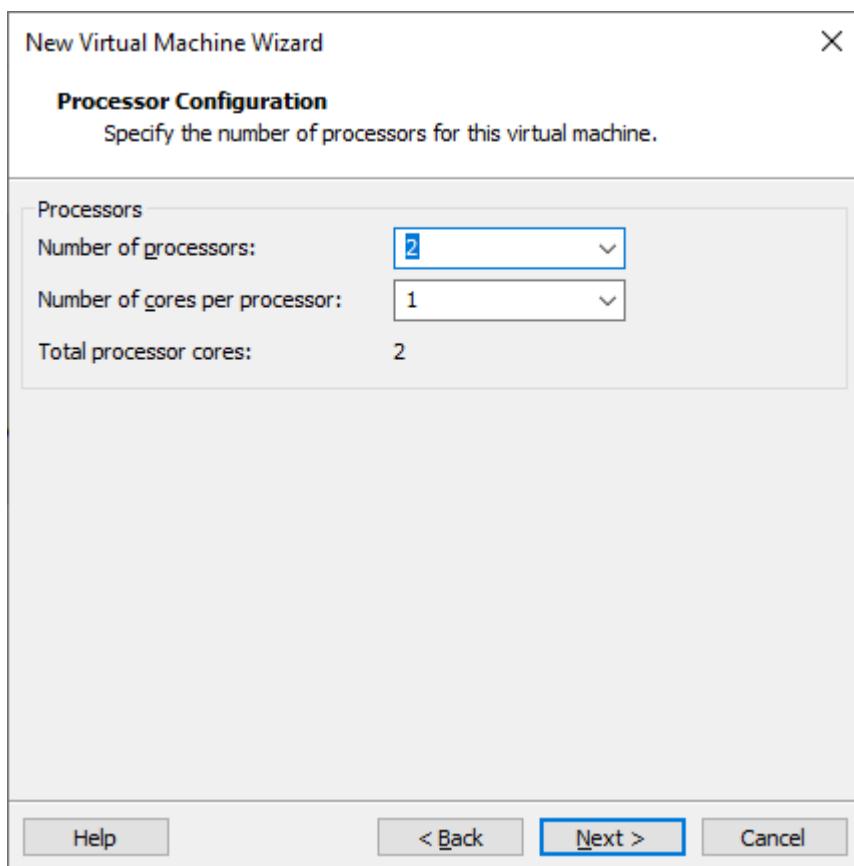
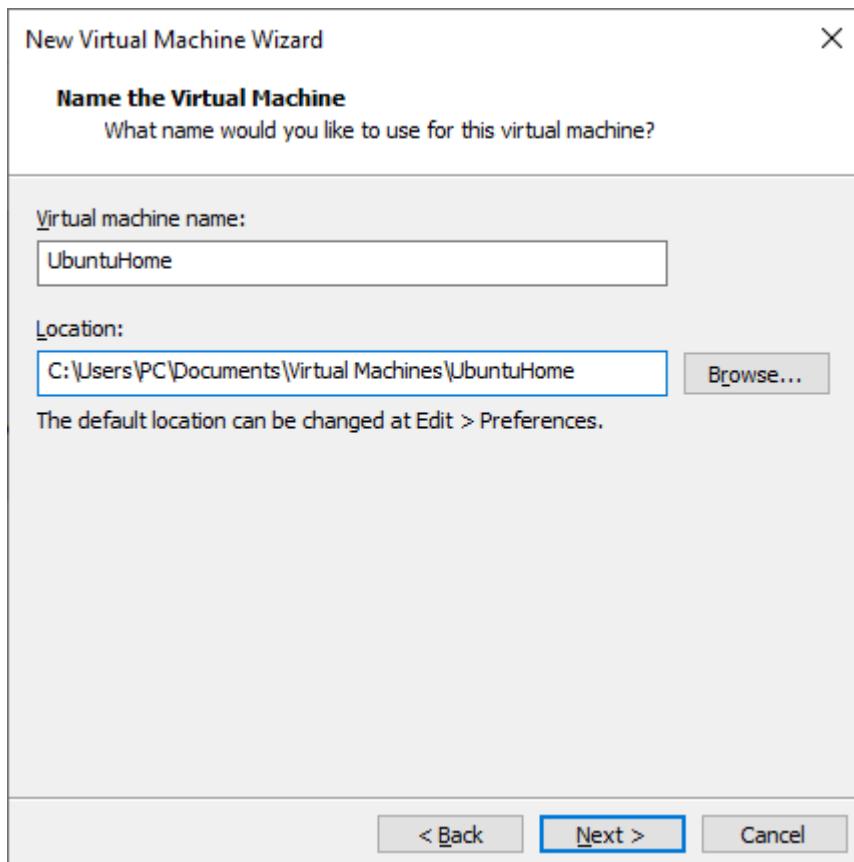


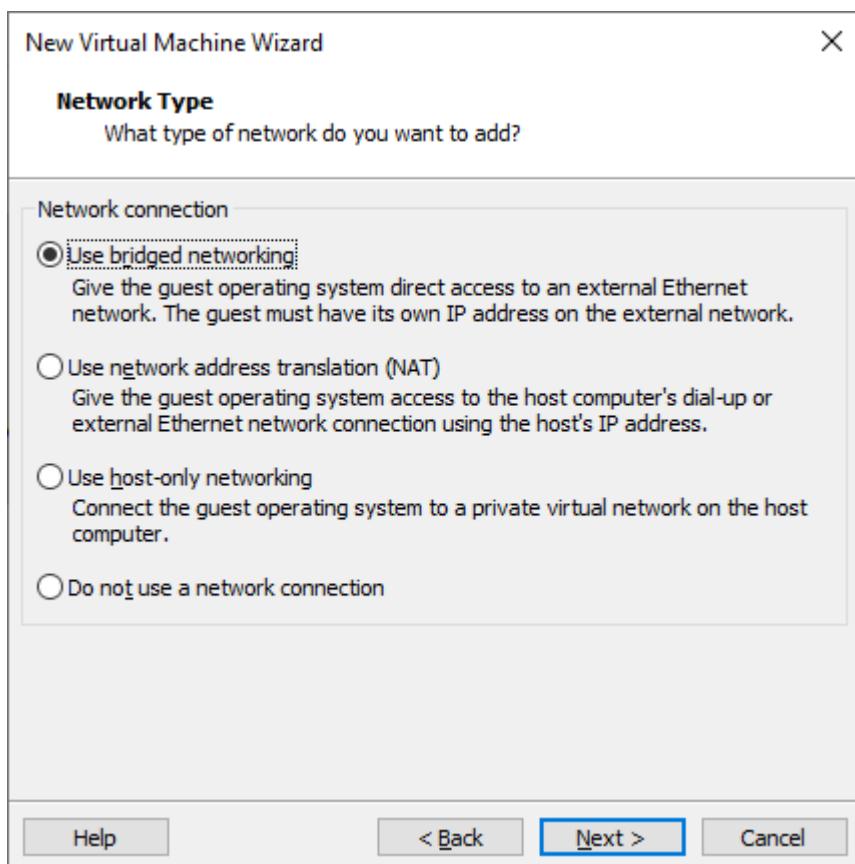
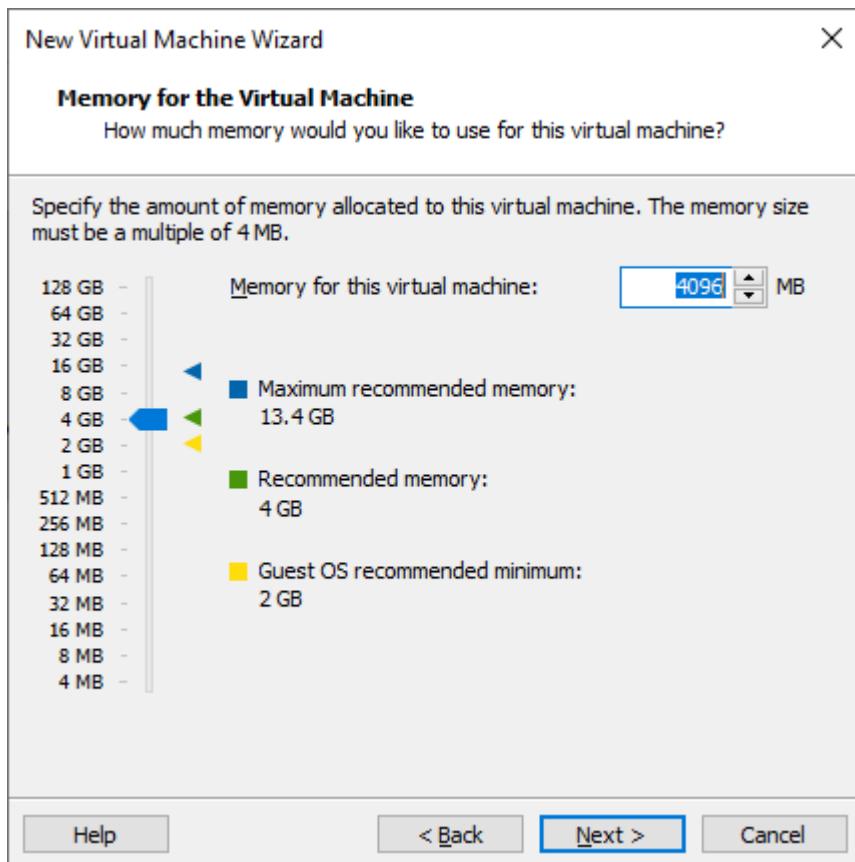
Создание виртуальной машины в VMware

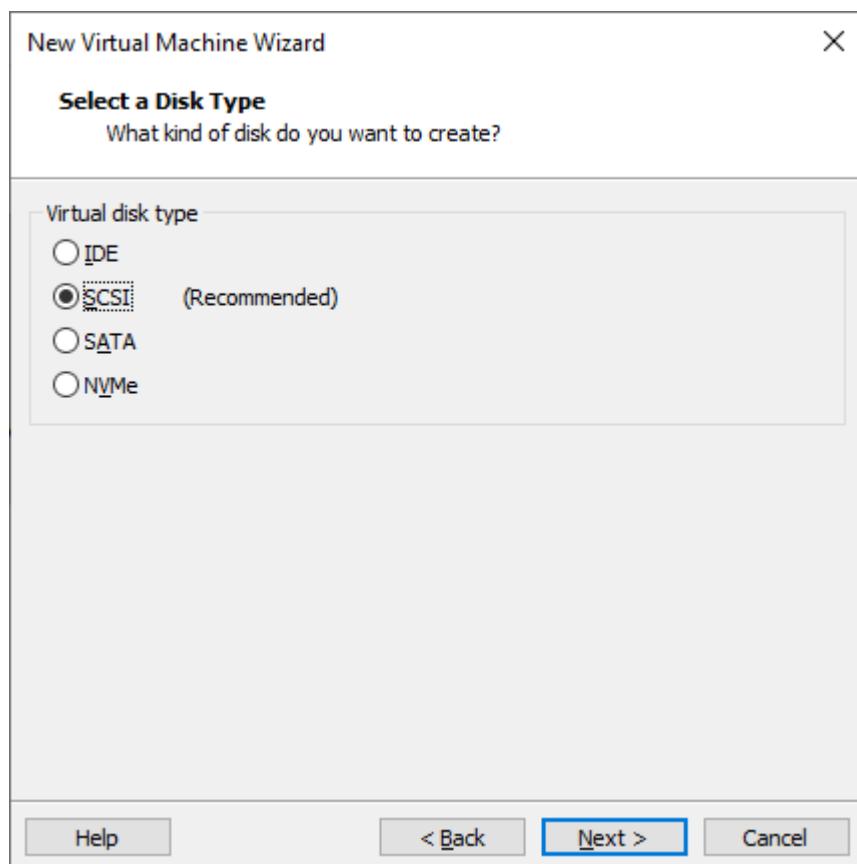
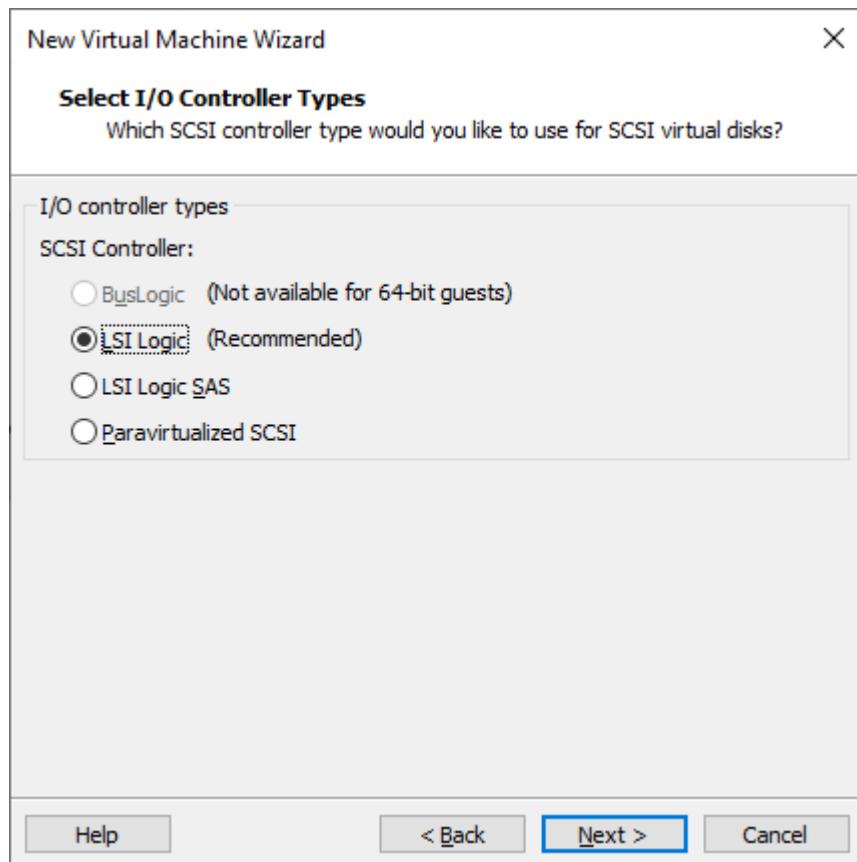


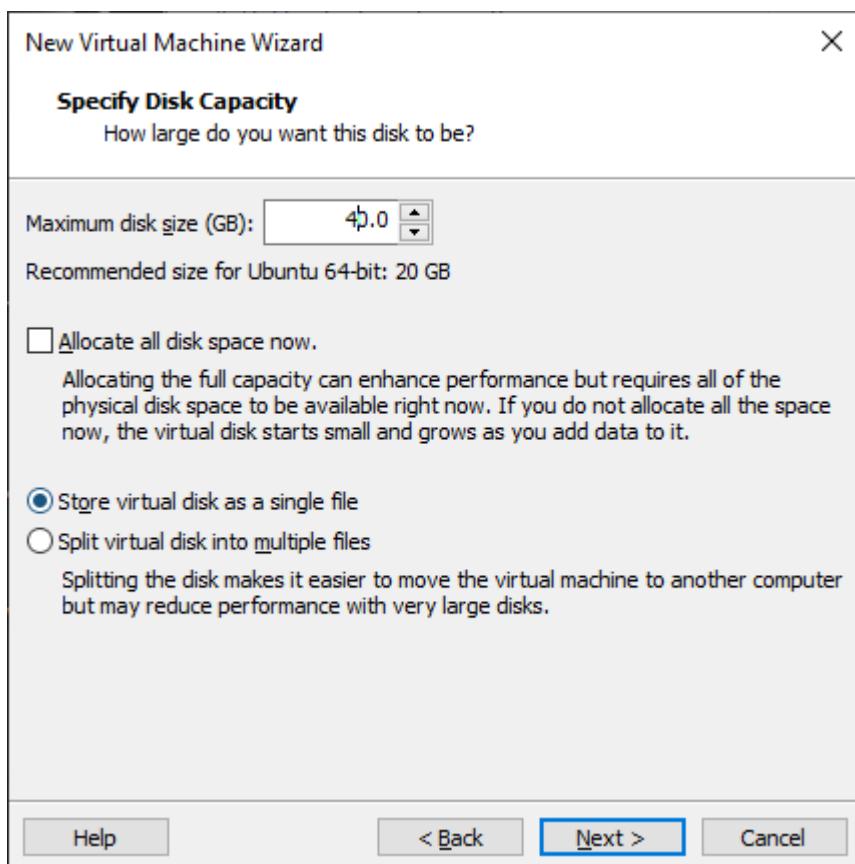
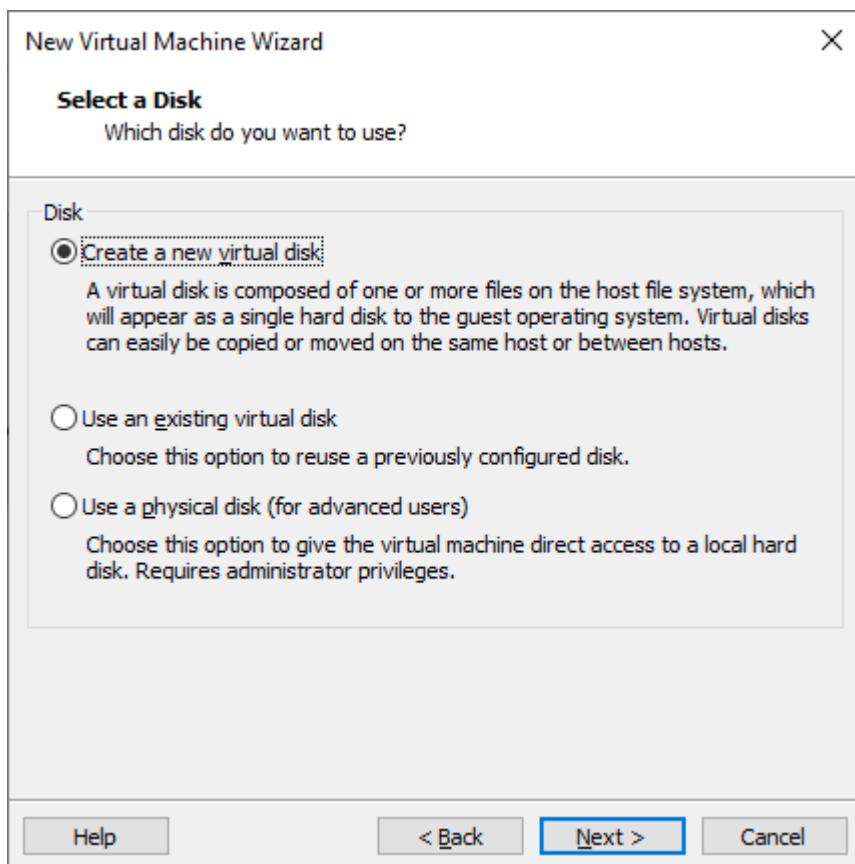


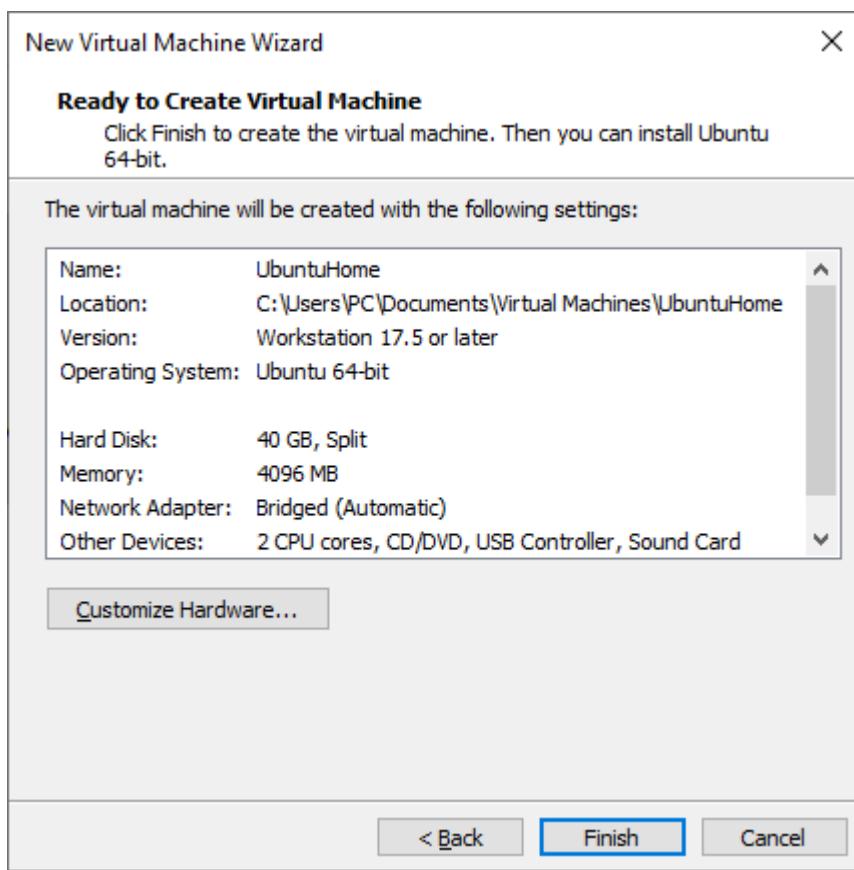
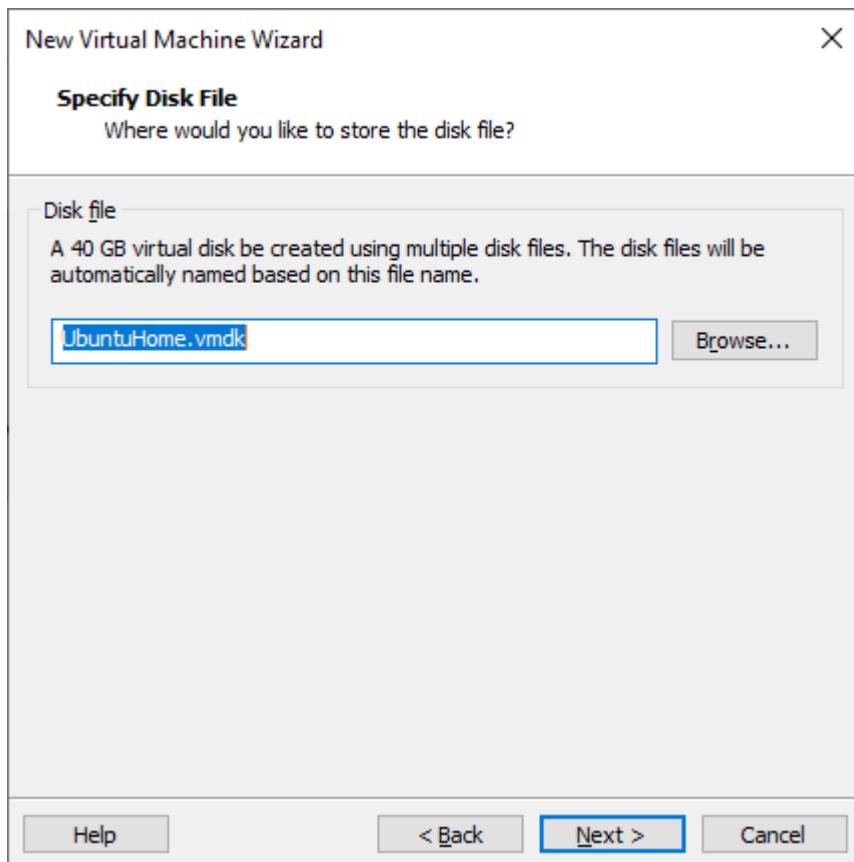


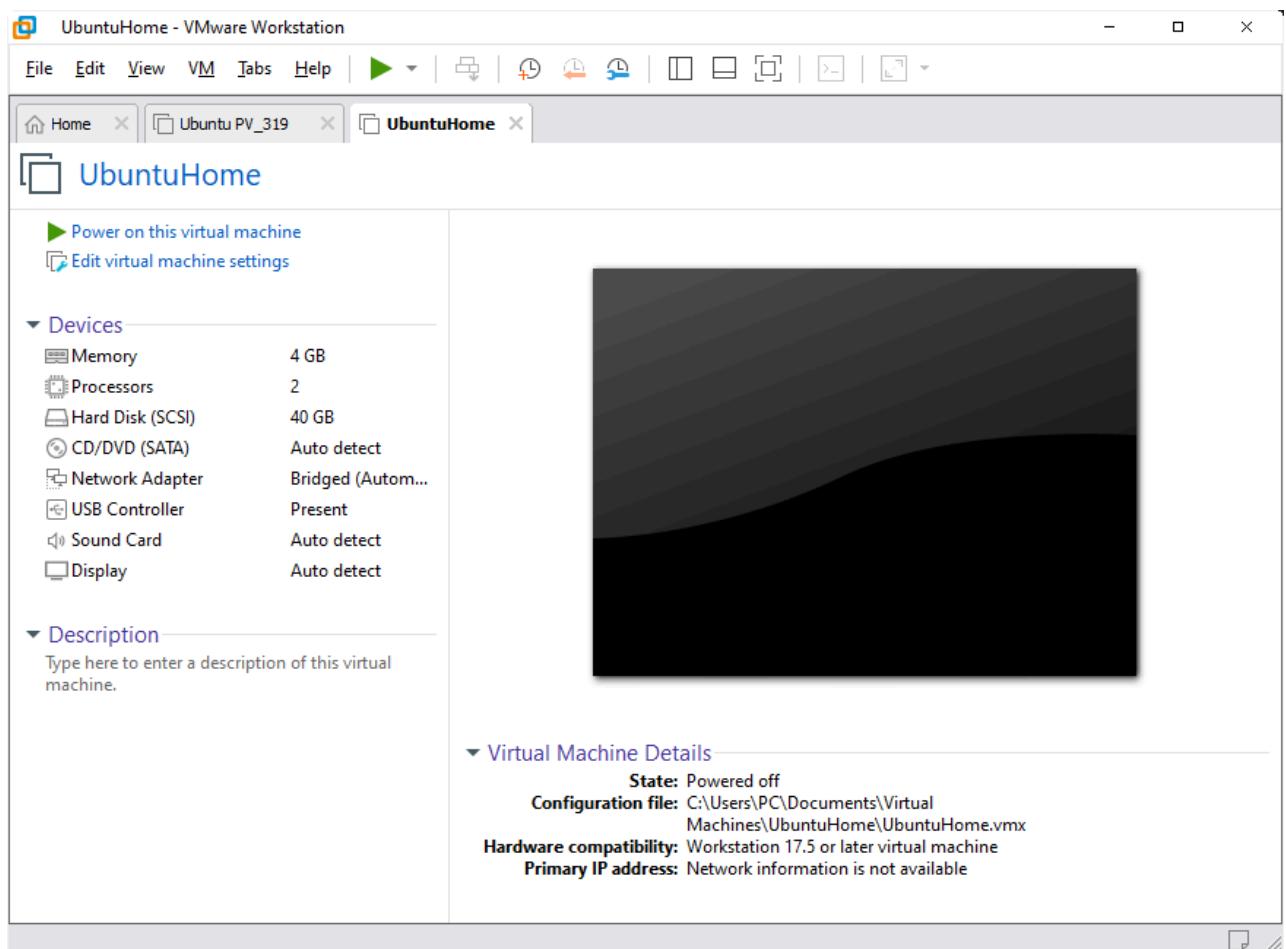


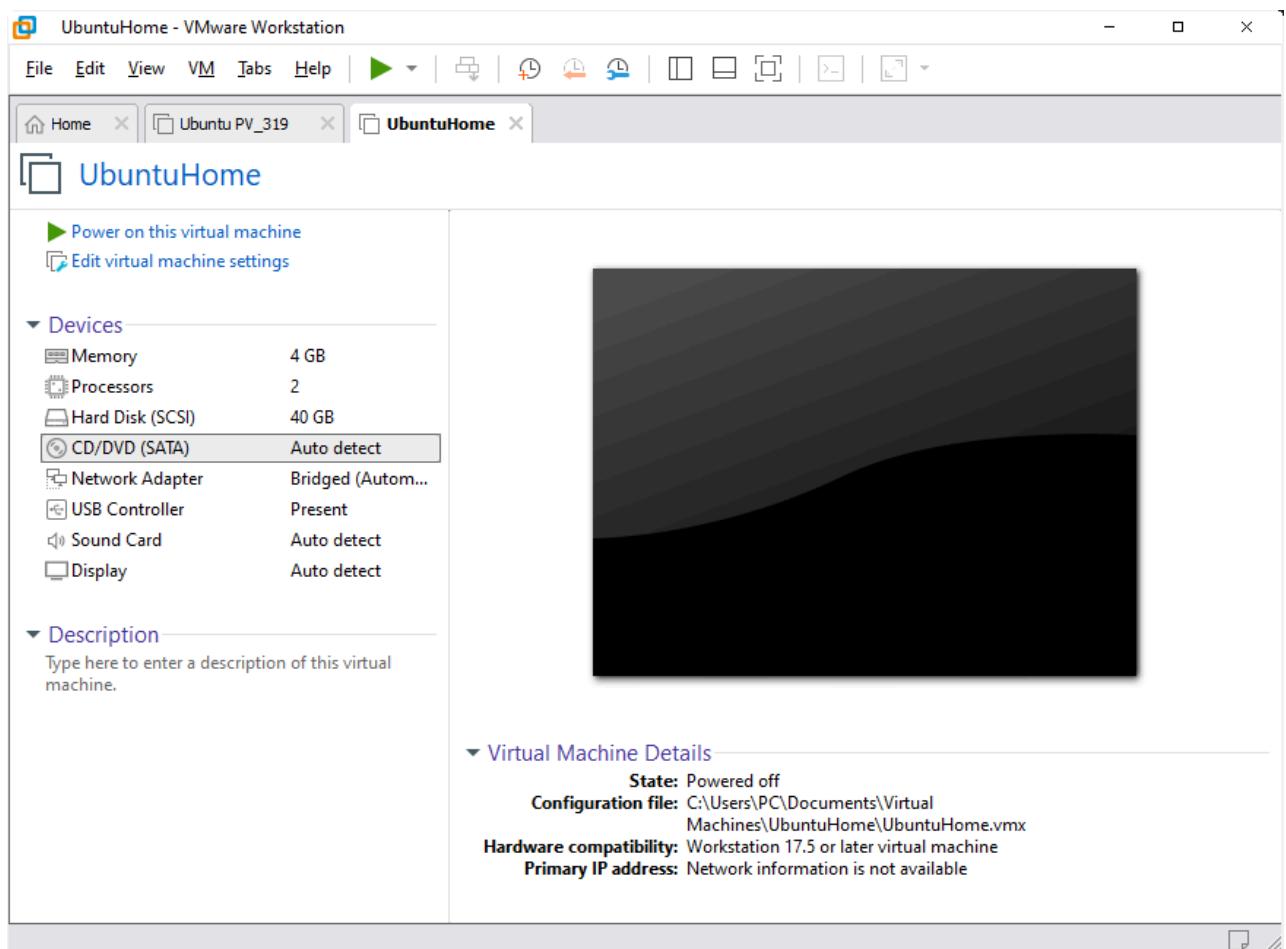












Virtual Machine Settings

Hardware Options

Device	Summary
Memory	4 GB
Processors	2
Hard Disk (SCSI)	40 GB
CD/DVD (SATA)	Auto detect
Network Adapter	Bridged (Automatic)
USB Controller	Present
Sound Card	Auto detect
Display	Auto detect

Device status

- Connected
 Connect at power on

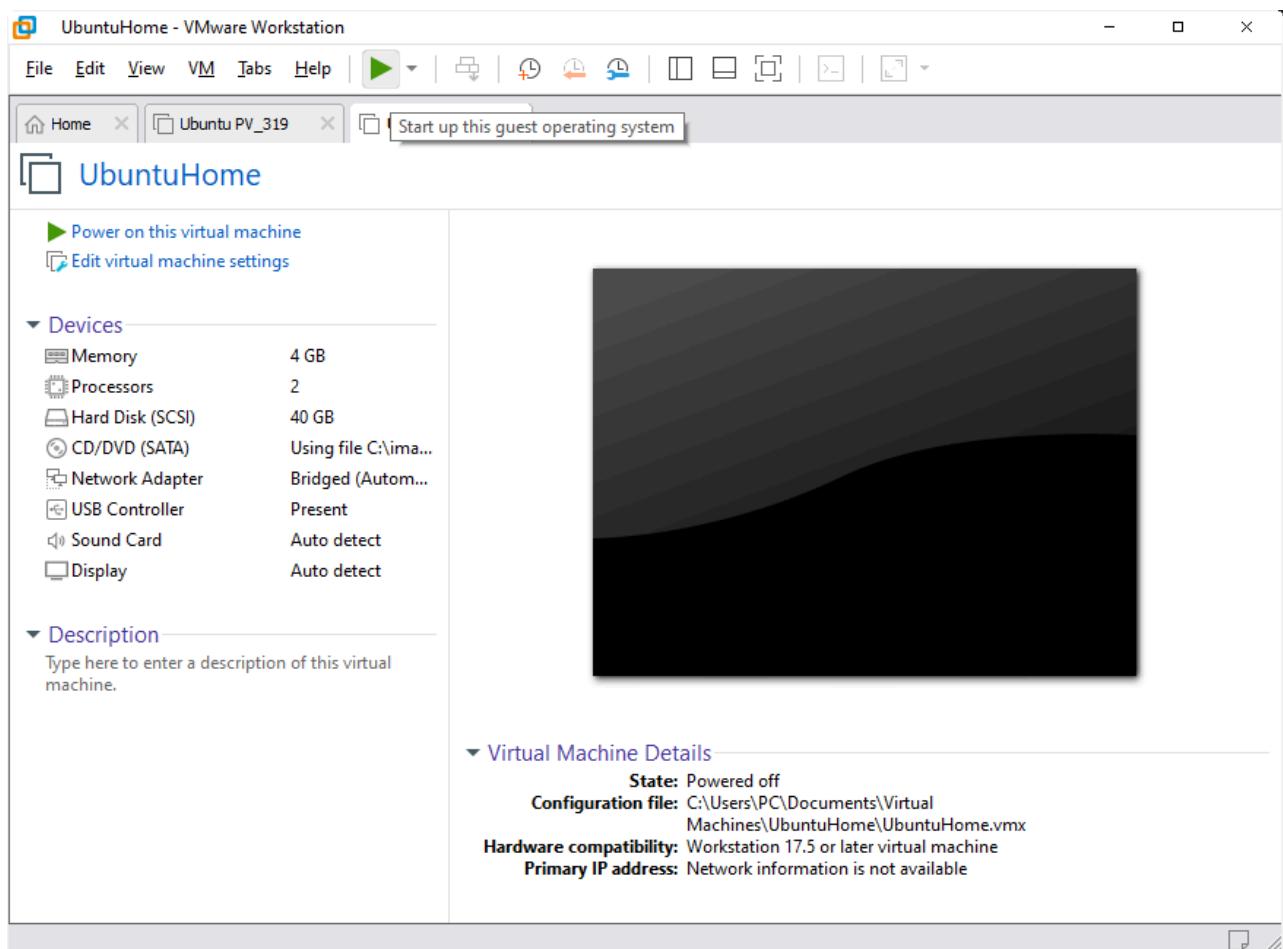
Connection

Use physical drive:

Auto detect

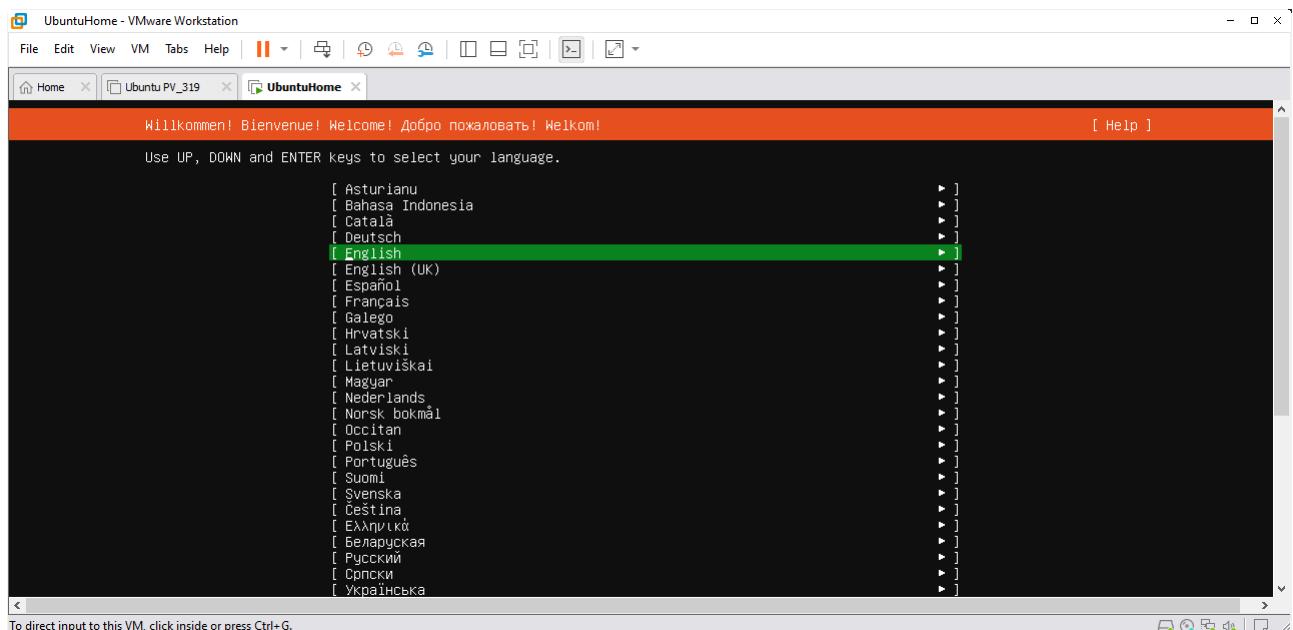
Use ISO image file:

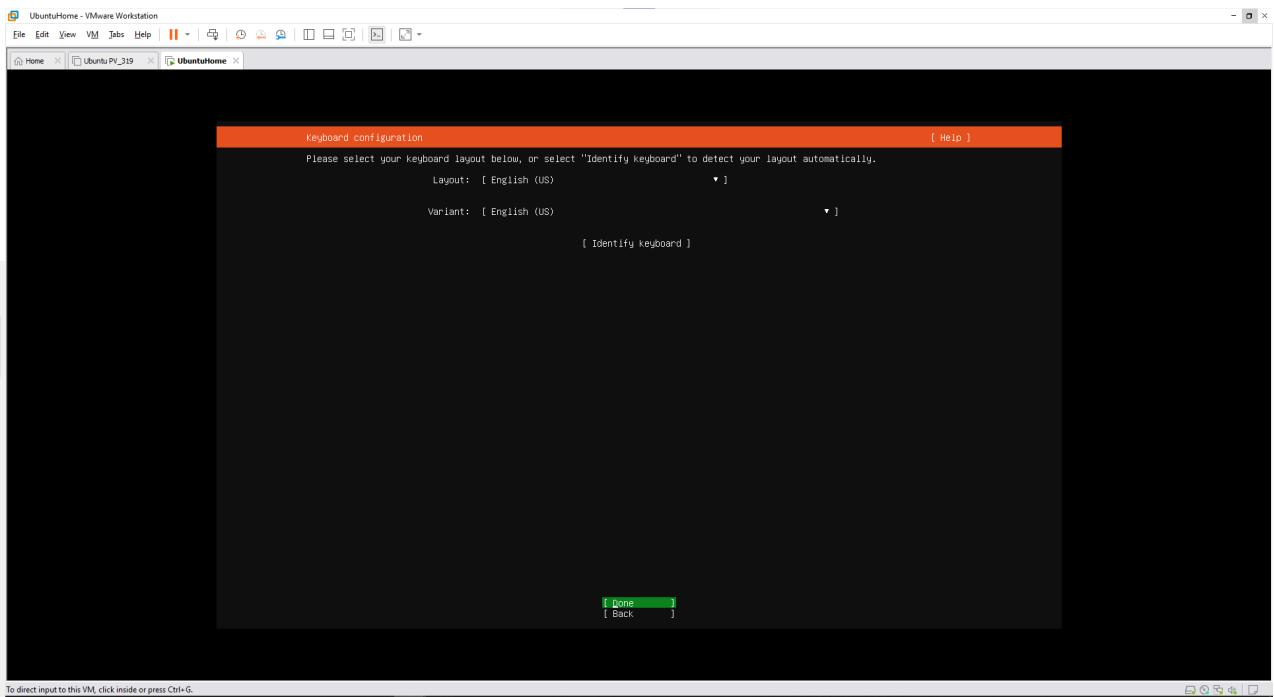
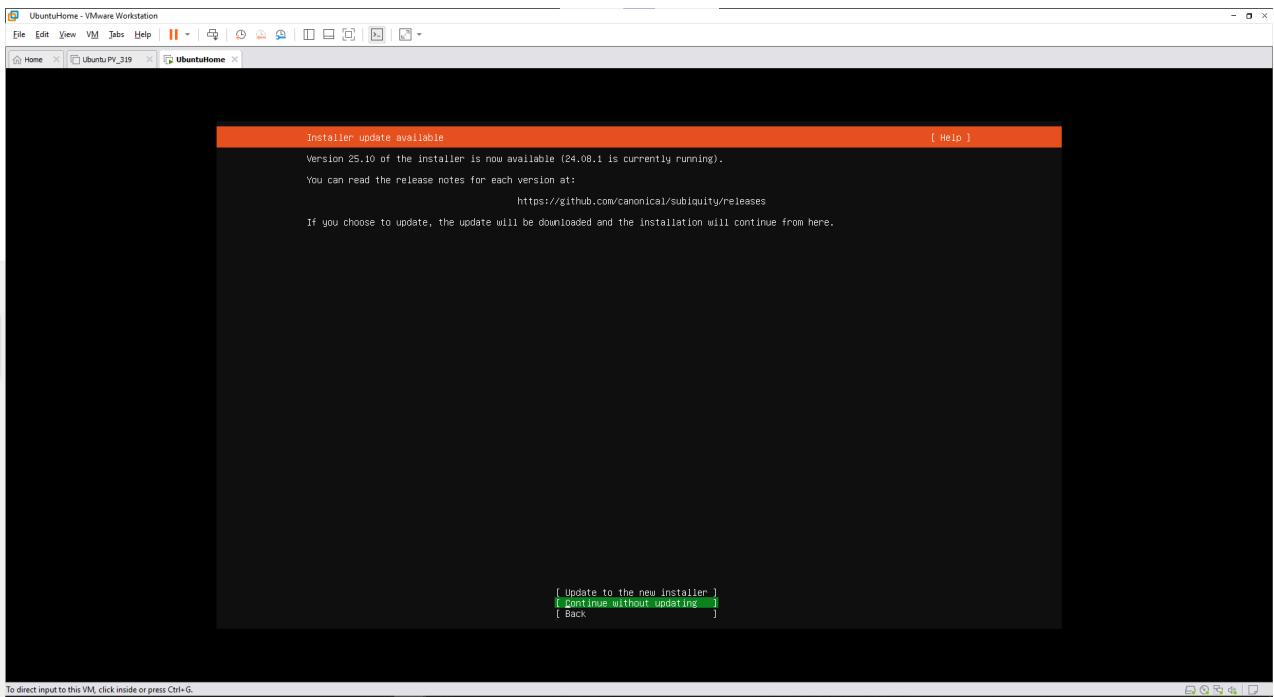
C:\images\UNIX\ubuntu-22.04.5-live-server-amd64

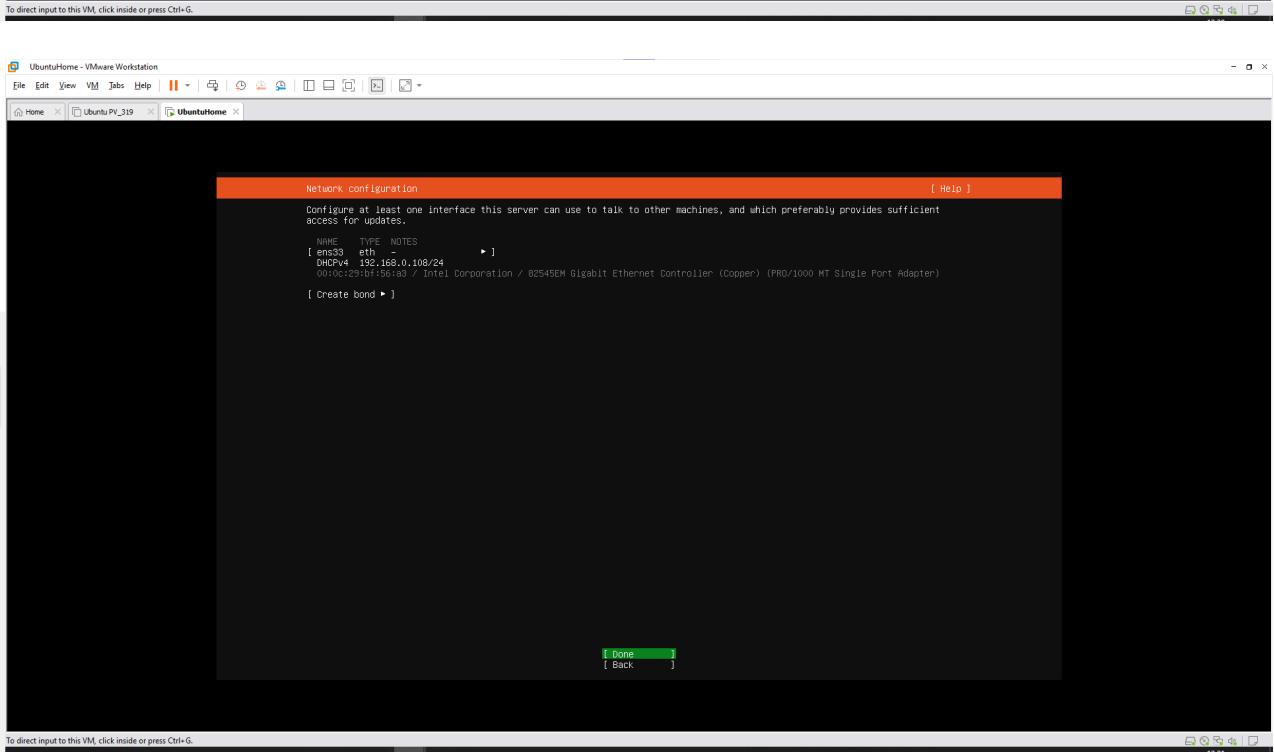
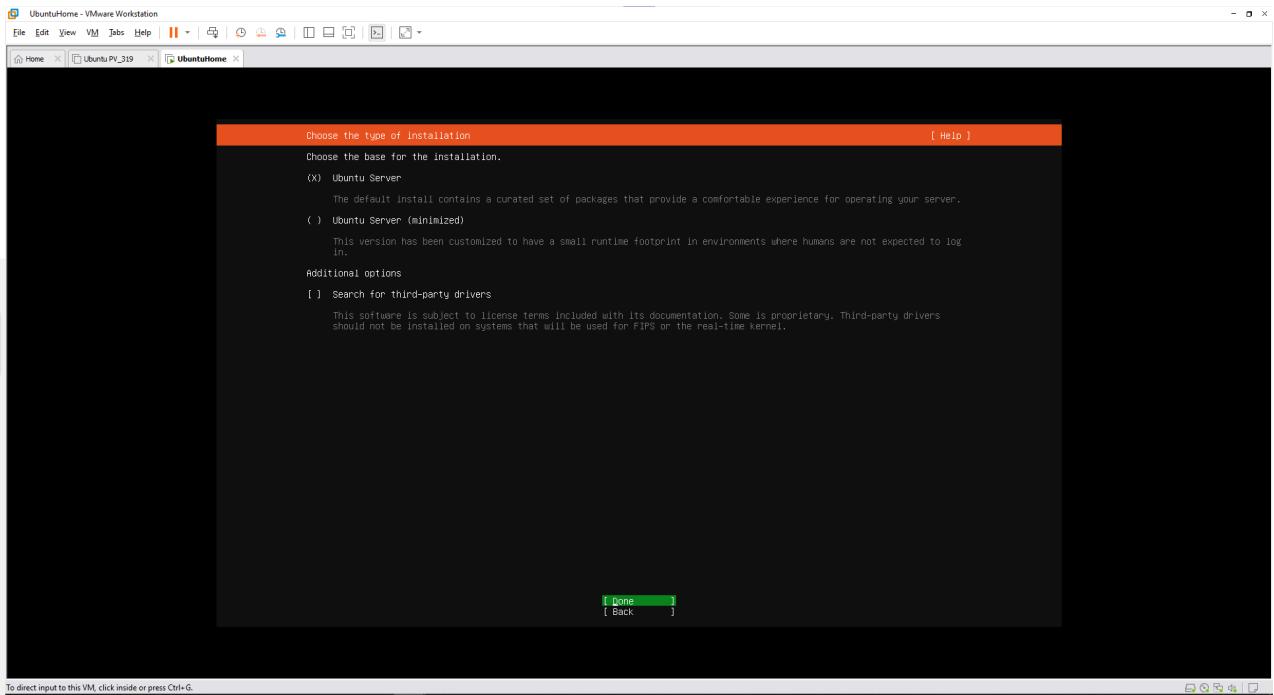


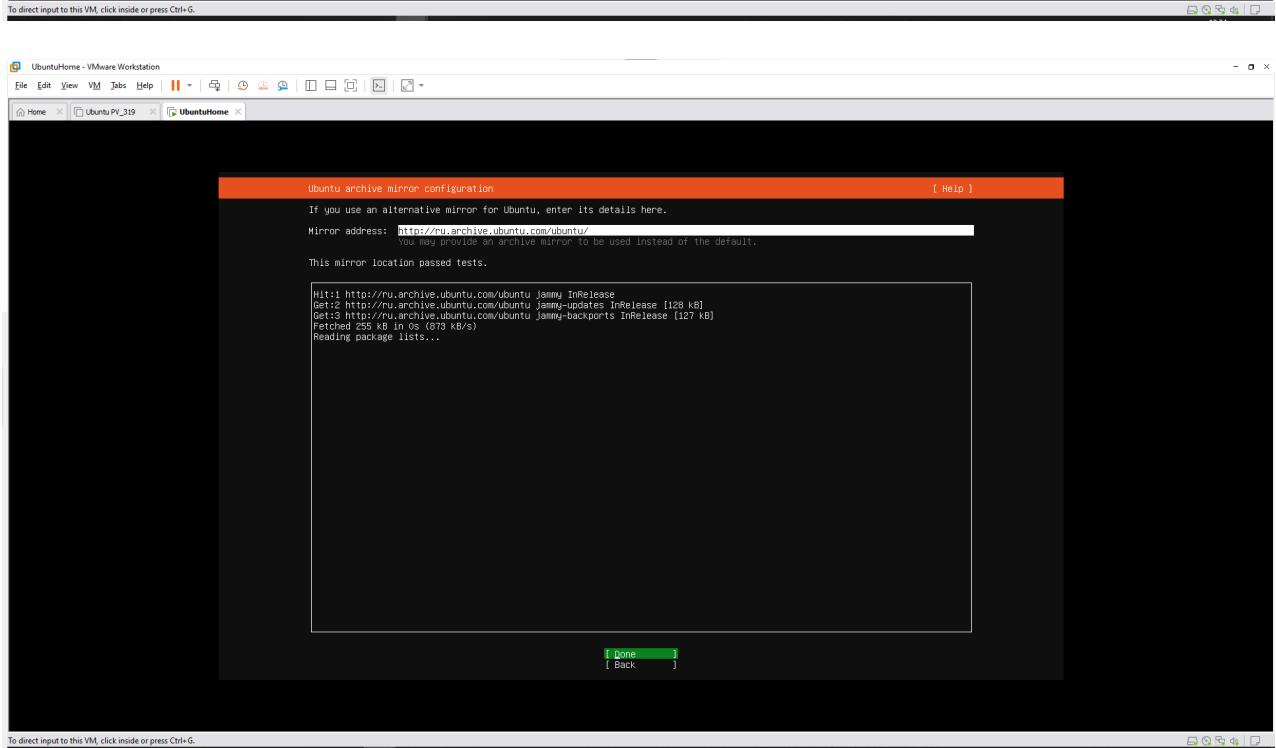
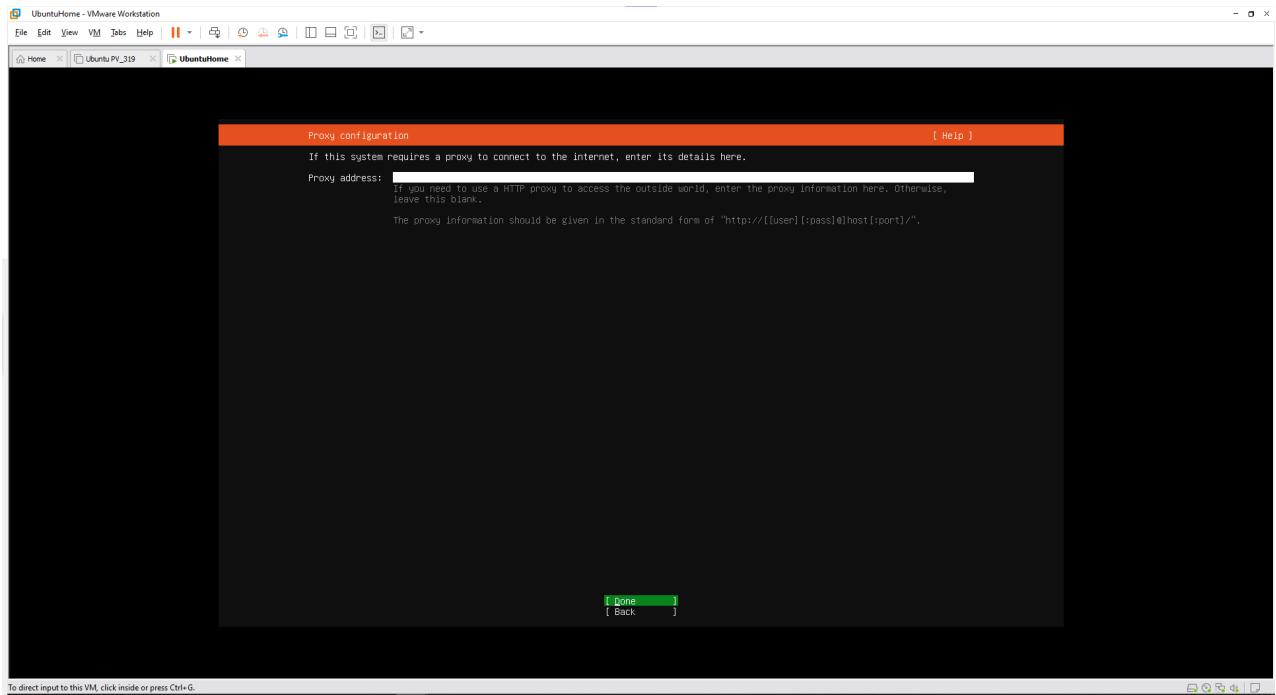
Ctrl+Alt+Enter – на весь экран

1. Try or install Ubuntu Server







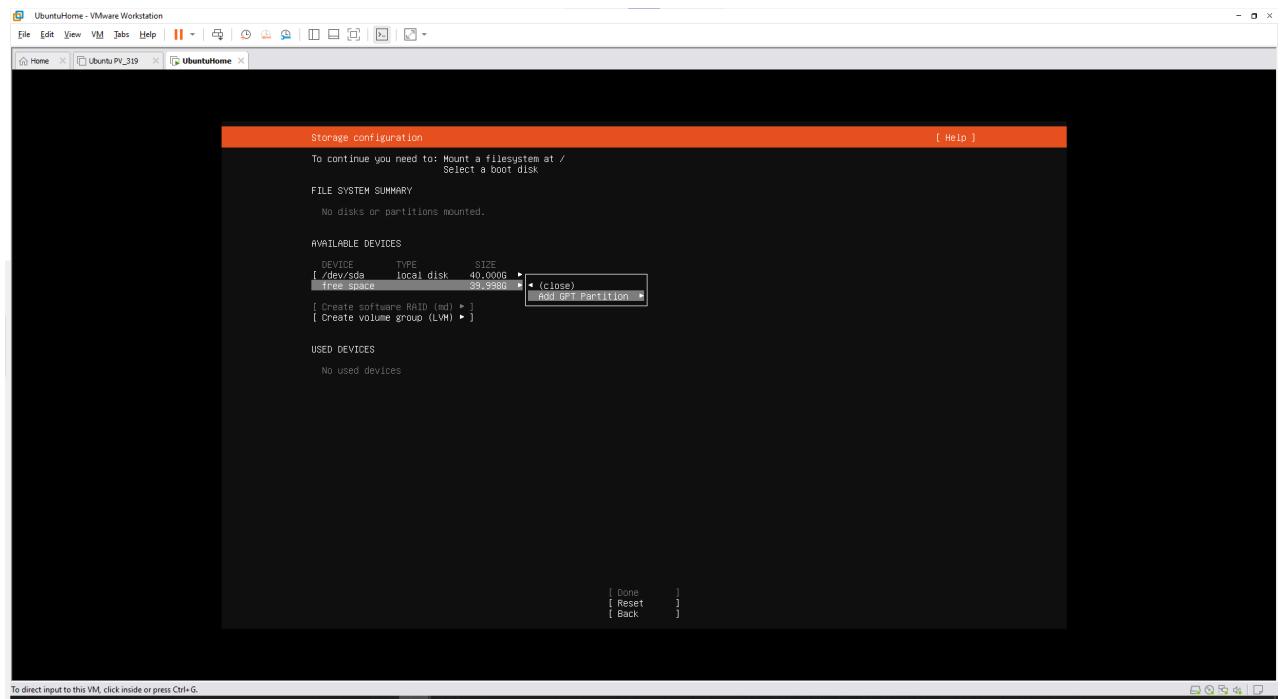
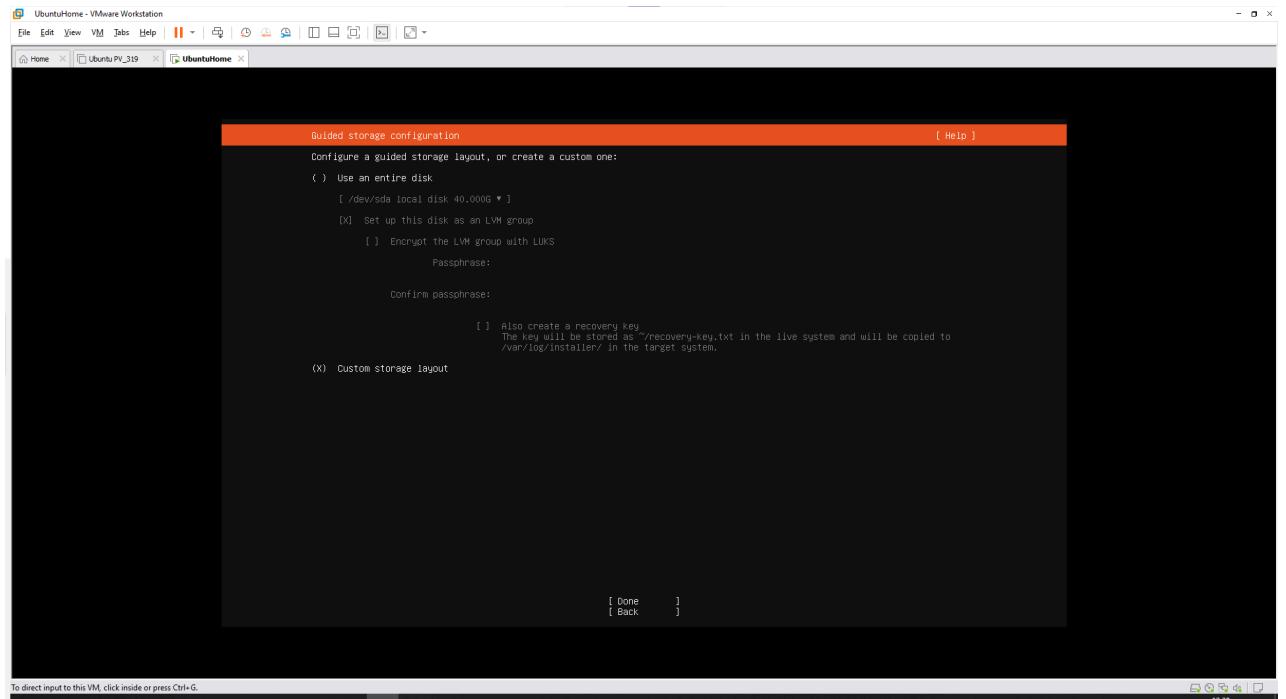


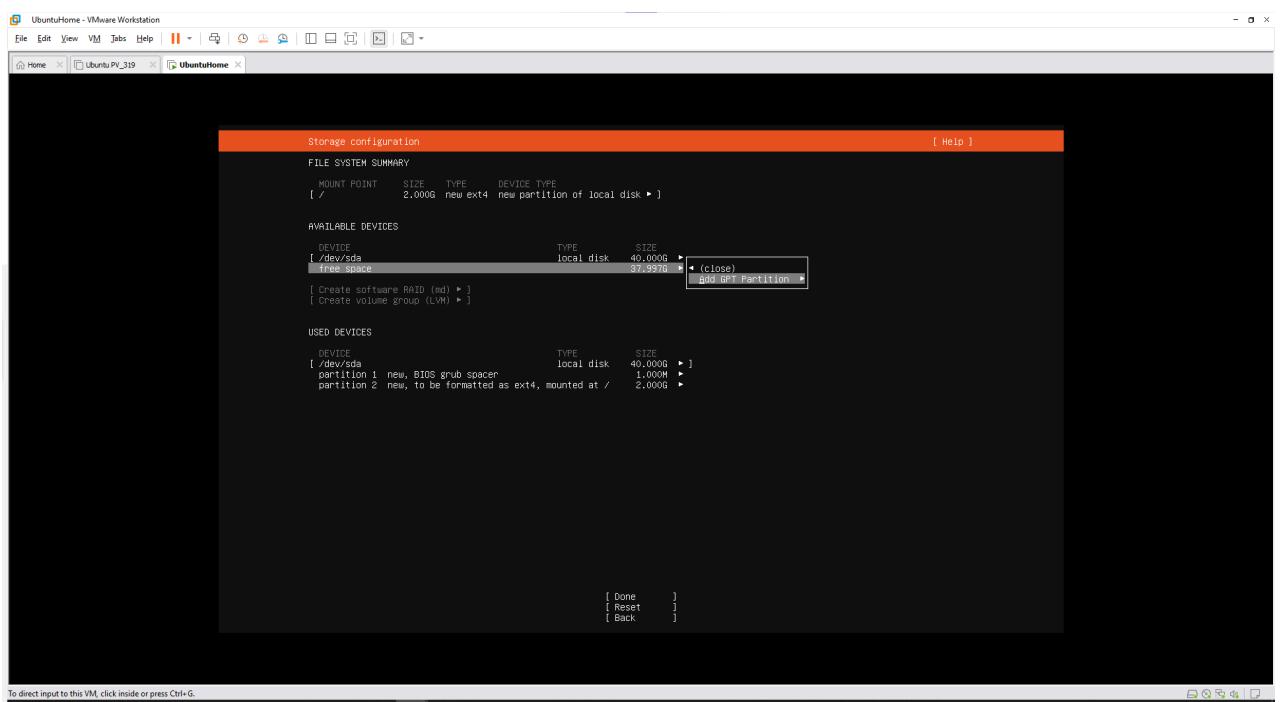
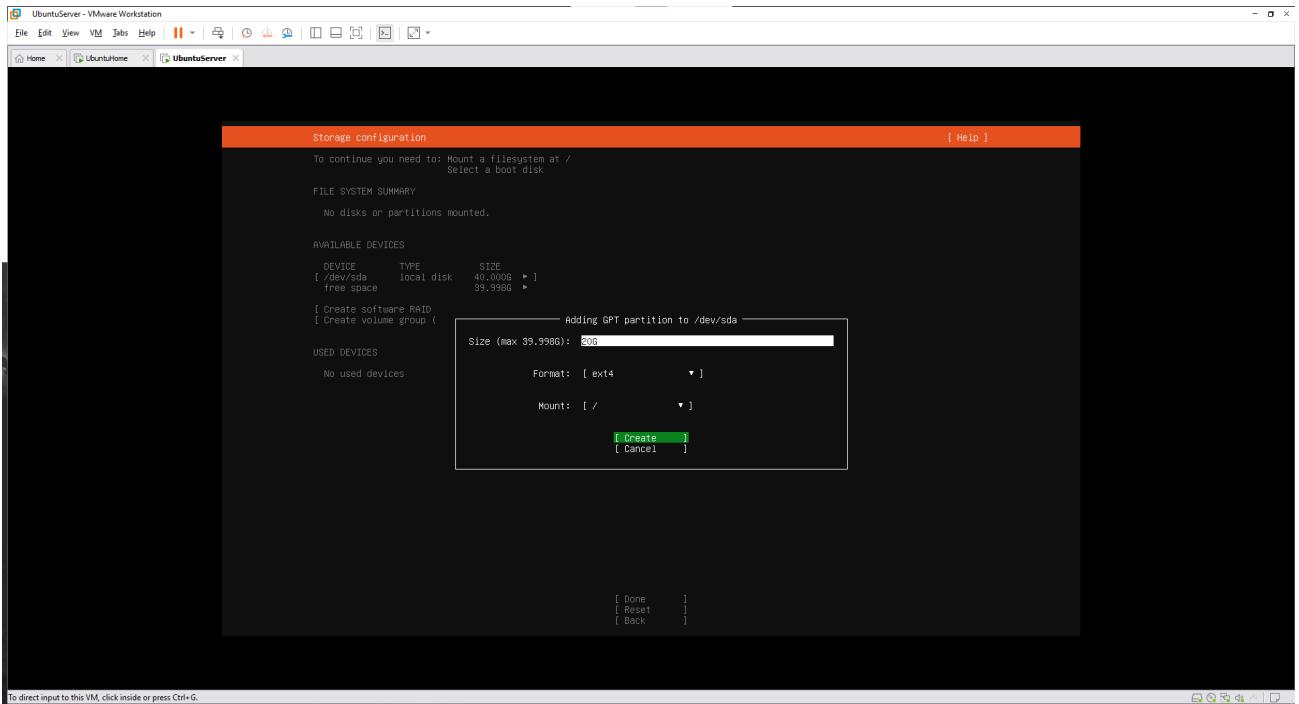
Tab – переход на след. Строку

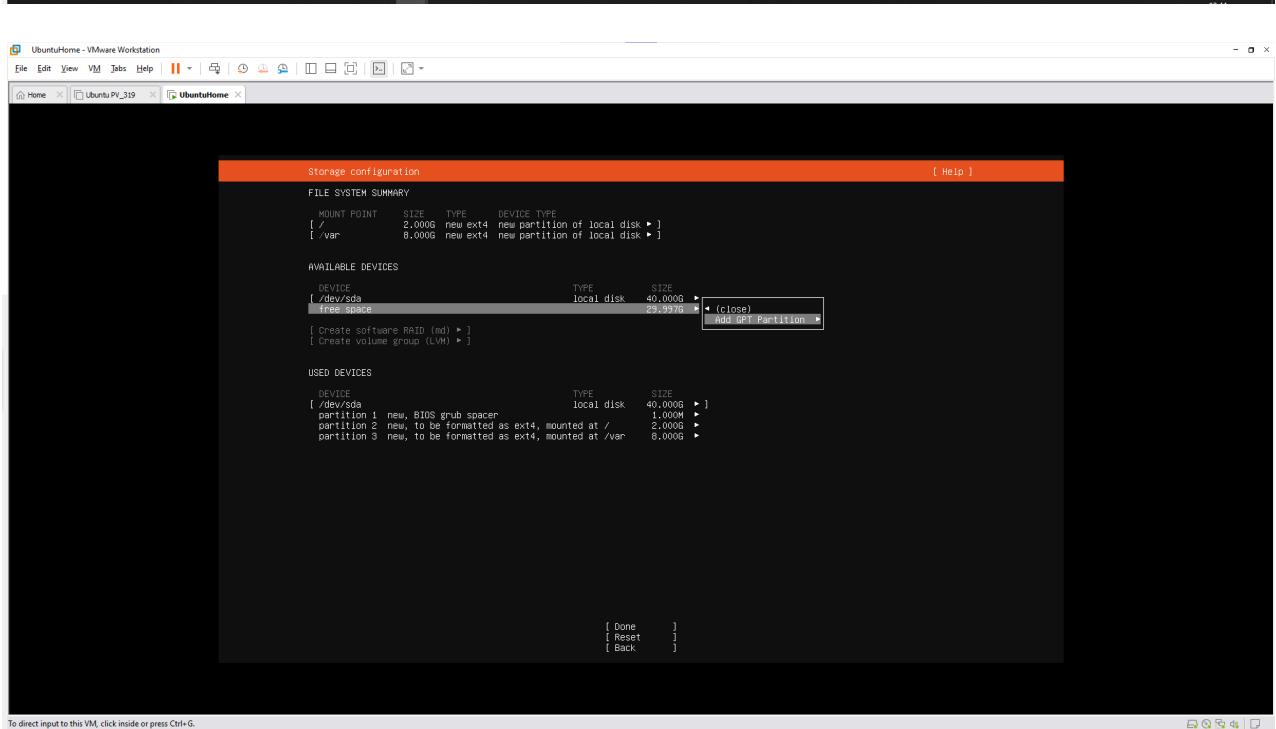
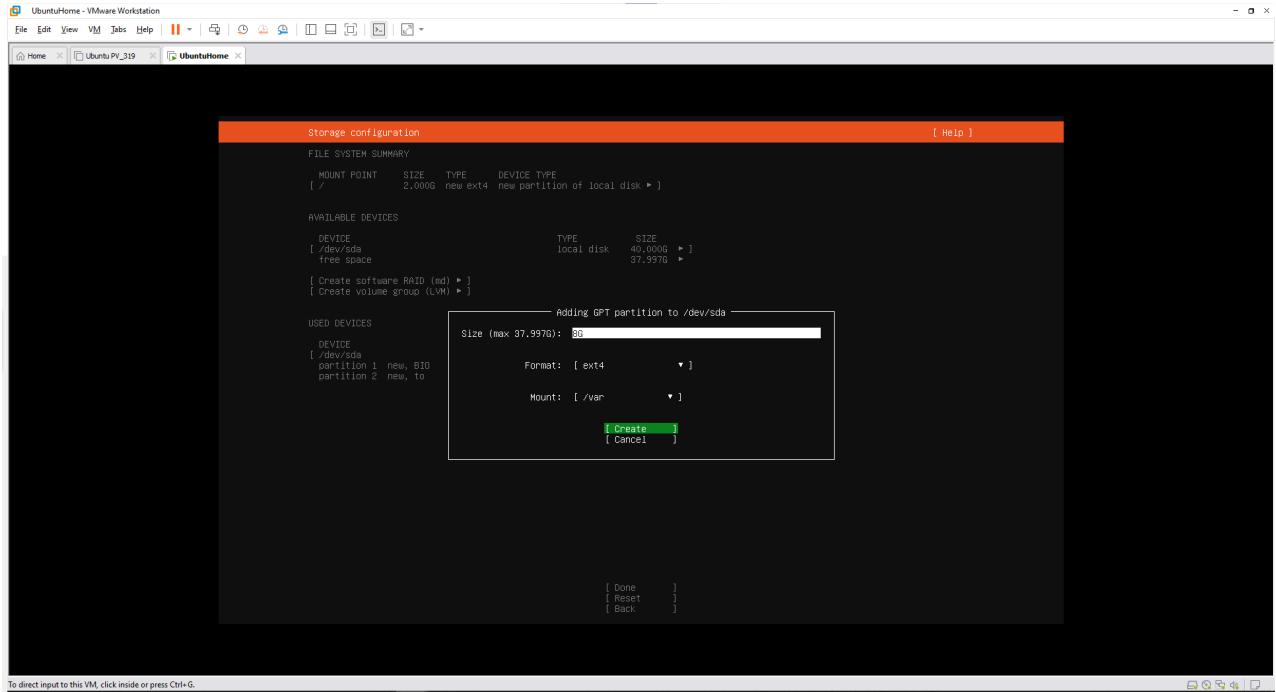
Shift+Tab – переход на предыдущую строку

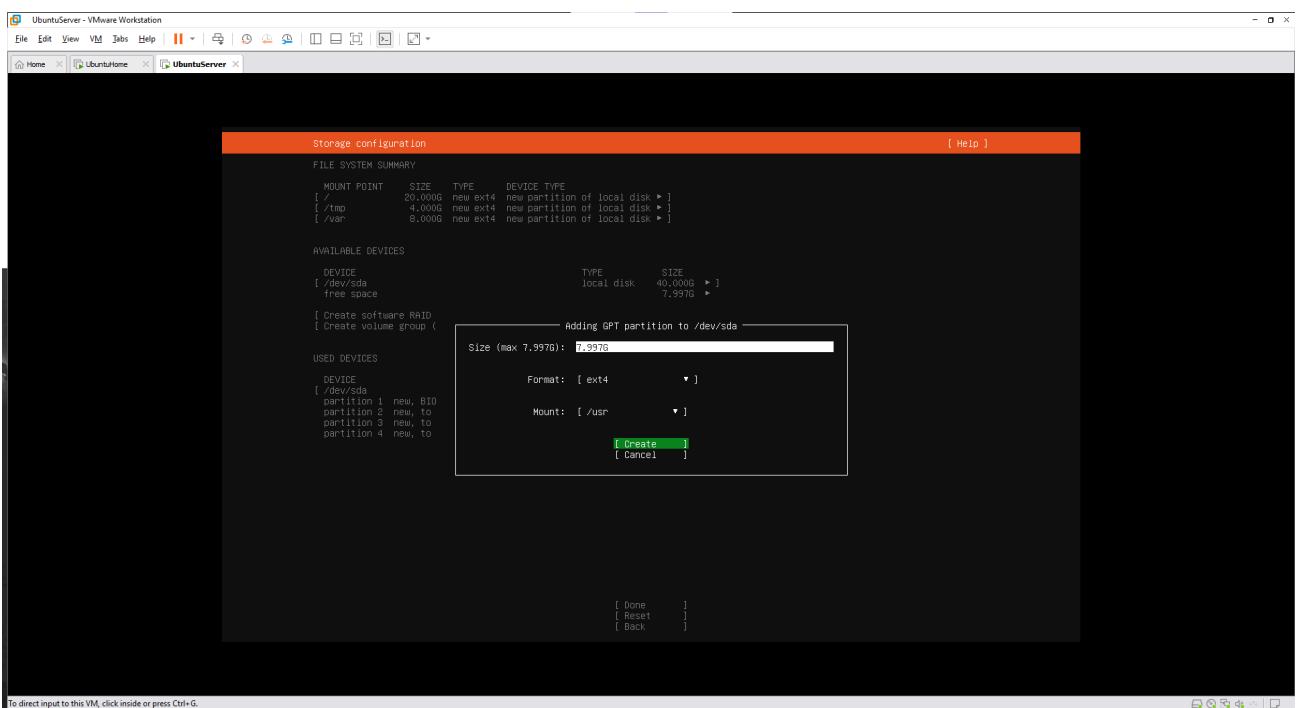
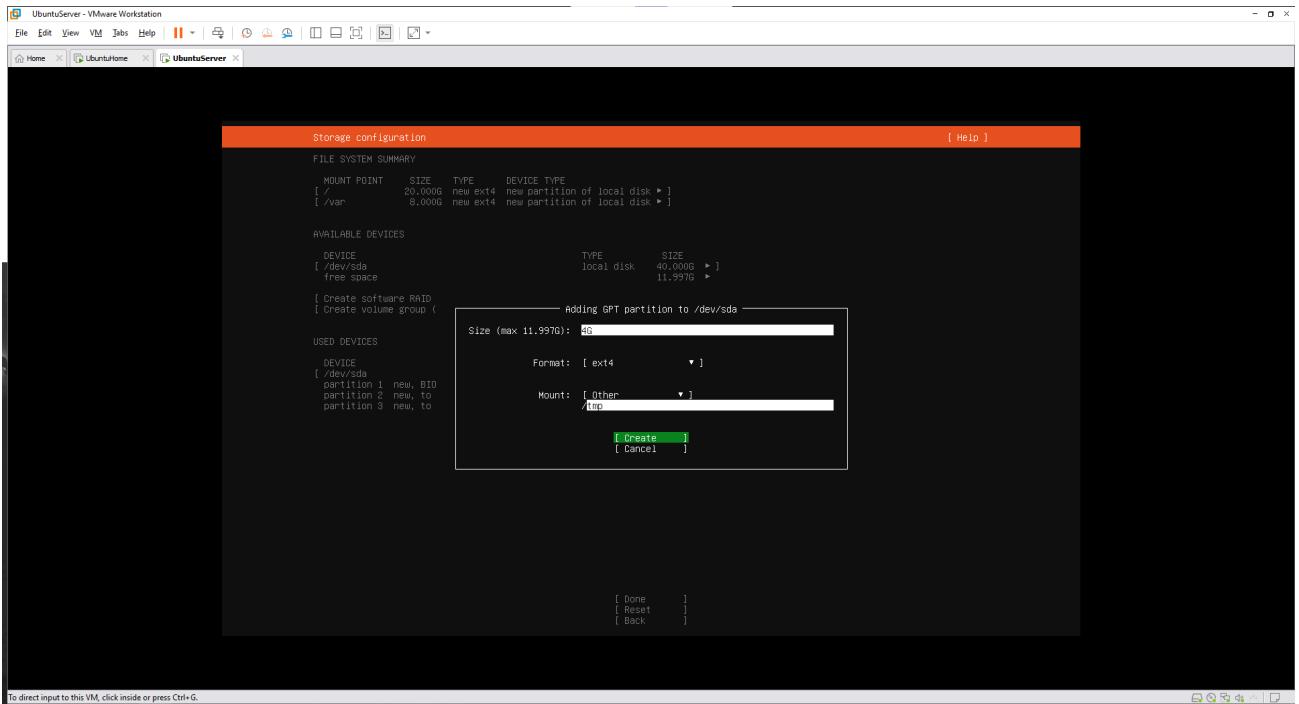
Перемещаться между строками можно также с помощью стрелочек

Ставим пробел чтобы появился крестик на custom storage layout:

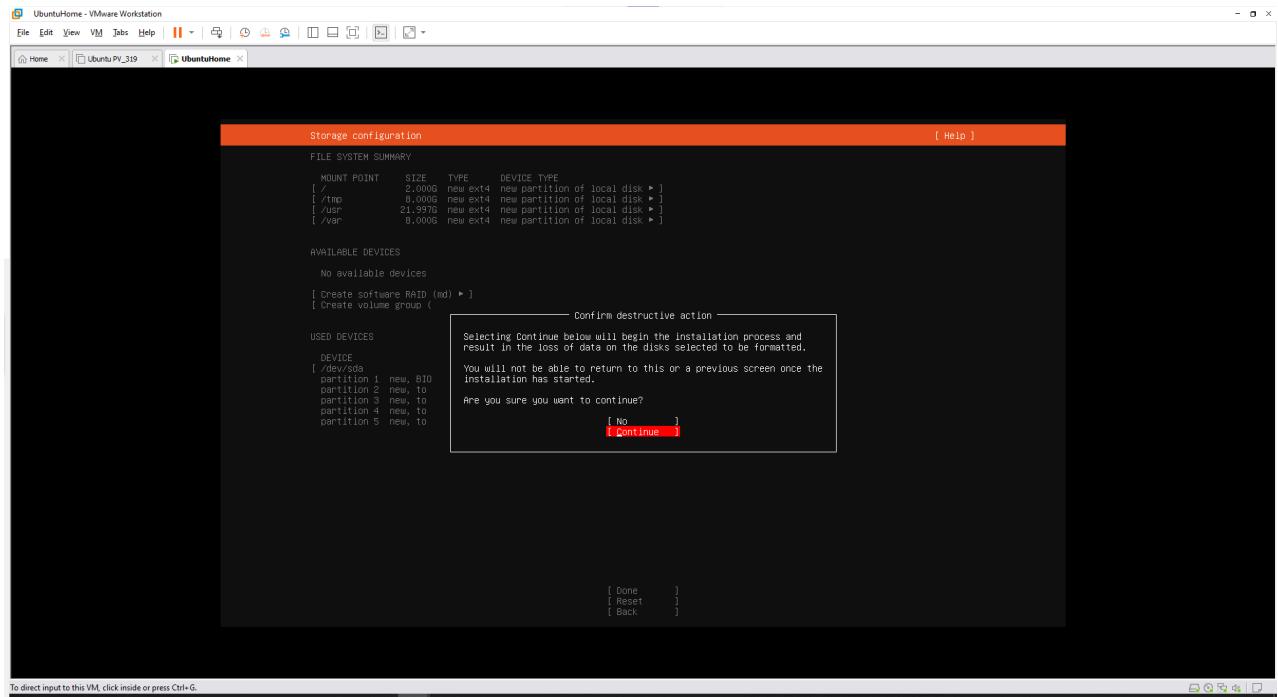




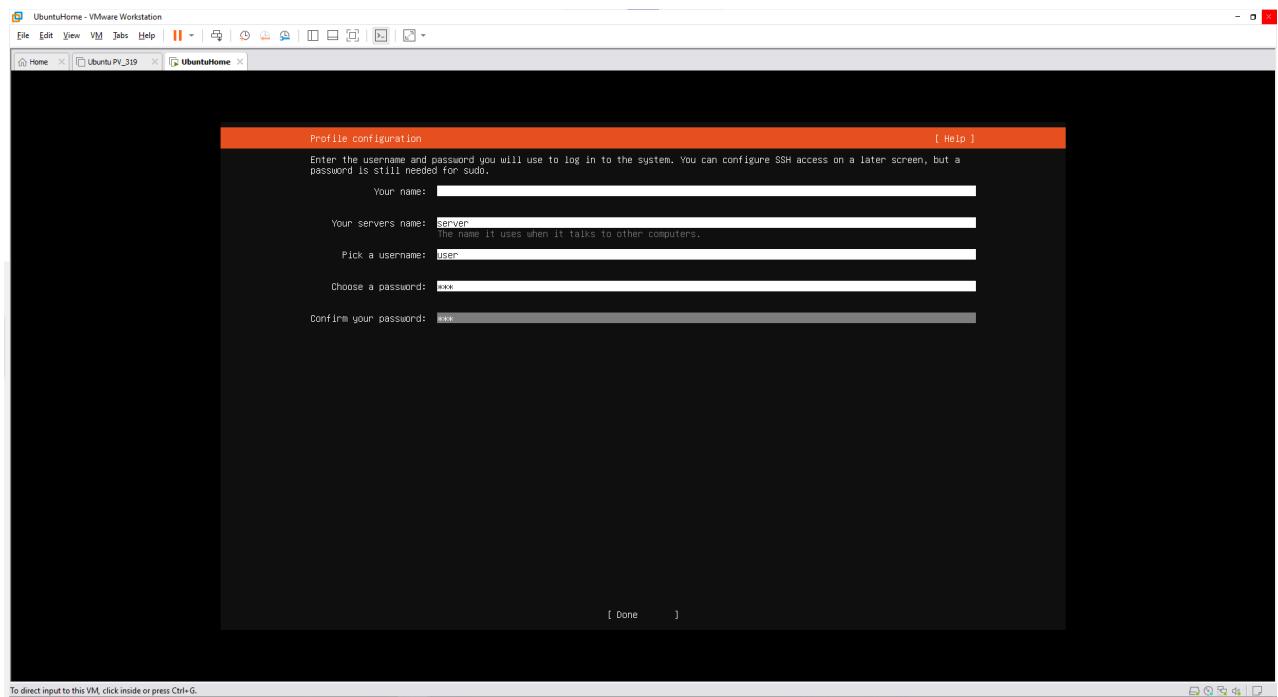




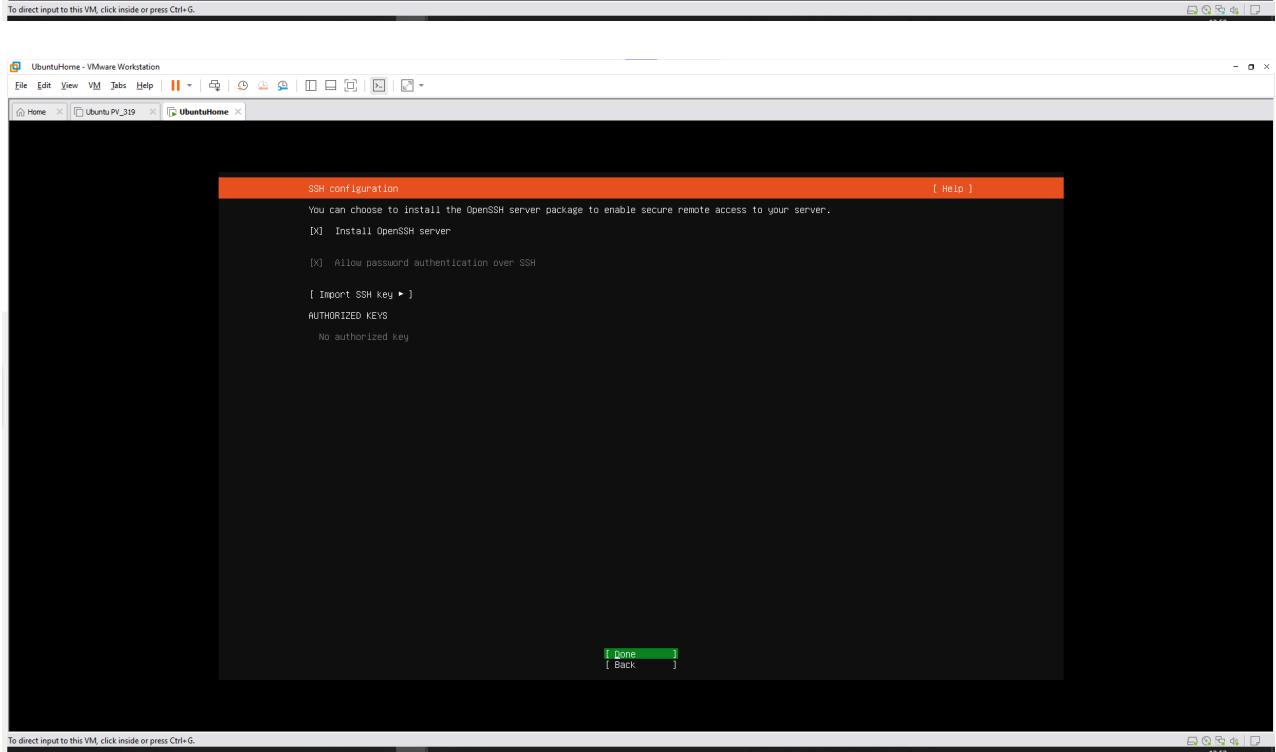
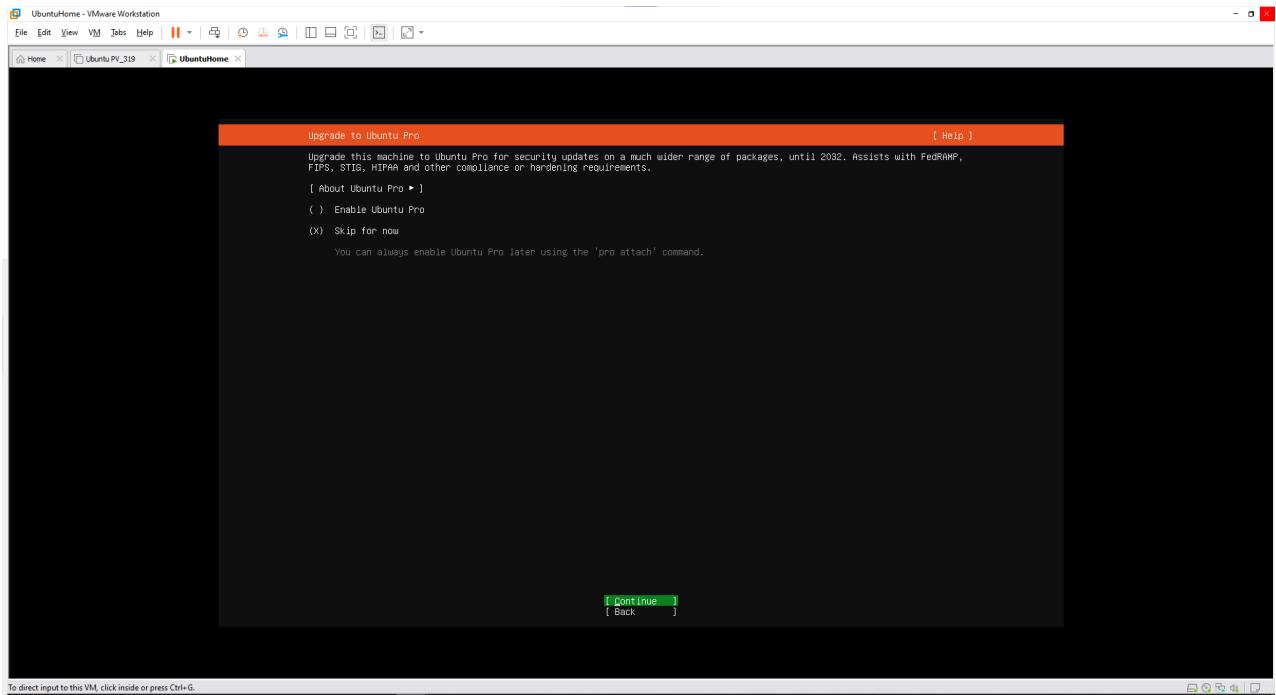
DONE

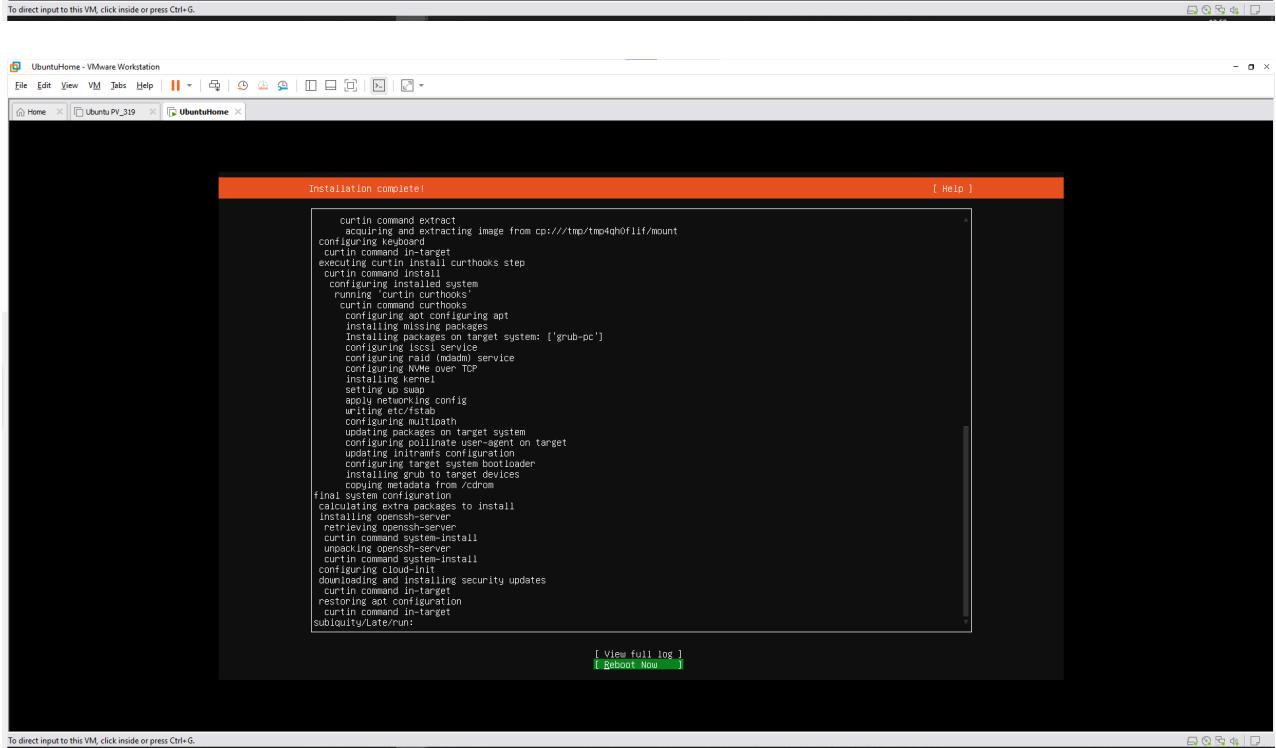
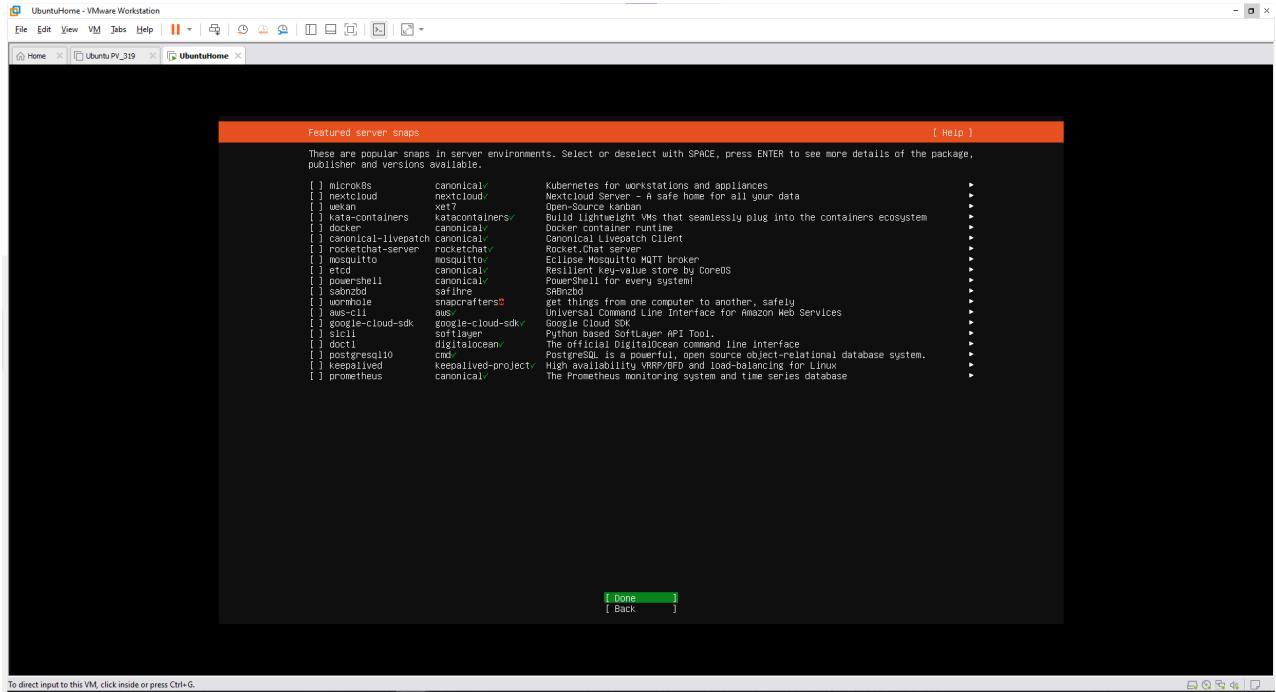


To direct input to this VM, click inside or press Ctrl+G.



To direct input to this VM, click inside or press Ctrl+G.





To direct input to this VM, click inside or press Ctrl+G.

Enter

Enter

```
user@server:~$
```

Основные команды:

pwd - print work directory

- Выводит на экран текущий каталог

ls – отображает содержимое указанного каталога. Если каталог не указать, ls показывает содержимое текущего каталога.

ls -a или **ls -all**

показывает скрытые папки и файлы. Для того чтобы сделать файл скрытым, нужно поставить точку перед его именем.

У каждого каталога есть 2 ссылки:

- точка – ссылка на текущий каталог
- точка точка – ссылка на родительский каталог.

`ls -d` – отображает свойства каталога без его содержимого. Этот ключ может быть полезен при просмотре каталогов с использованием масок или регулярных выражений.

`ls -l` – ключ `-l` означает The Long Format – вывод имен файлов в длинном формате. В длинном формате сначала выводятся тип файла и право доступа -> количество жестких ссылок (жесткая ссылка - символическое имя файла) -> владелец файла и группа файла -> размер файла -> дата последнего изменения файла -> имя файла
ключ `-h` означает The Human Readable. Этот ключ актуален только с ключом `-l`.

Ключи можно использовать как каждый со своим минусом, так и объединить все ключи под одним минусом. Но объединение ключей под одним минусом возможно только в том случае, если у объединяемых ключей нет параметров. Если же хотя бы у 1 объединяемого ключа есть параметры, его нужно поставить в конец объединения.

Сменить текущий каталог можно командой `cd`

`/` – корневой каталог, на который монтируется корневая файловая система. Это как диск C в Windows. Но в Unix-е концепция дисков полностью отсутствует. Остальные разделы монтируются на каталоги в корневом разделе, причем абсолютно любой раздел можно смонтировать на абсолютно любой каталог на любой глубине. То есть в любом Unix-е абсолютно все разделы объединяются на одно дерево. Внешние файловые системы, как правило, монтируются в каталоги `/mnt` и `/media`. `/` – показывает, что эти каталоги расположены в корневом каталоге.

Введенная без параметров команда `cd` переходит в домашний каталог

Все смонтированные файловые системы можно отобразить с помощью `mount`

Перемещаться по каталогам часто бывает удобно с помощью команд `pushd` и `popd`

`pushd` – сохраняет текущий каталог в стеке и переходит в указанный каталог. Без параметров команда `pushd` выводит на экран каталоговый стек

`popd` – переходит на вершину стека из текущего каталога на вершину стека.

Команда `ls` с ключом `I` не отображает действительные размеры каталогов, а лишь отображает размер текстового файла соответствующего каталога. Действительные размеры каталогов можно просчитать командой `du -hd 0`.

`-g` показывает фактический размер текущего или указанного каталога, включая все файлы и папки.

`du -hd 1 /etc | less`

- прокручивать консоль

`du` – disk usage (использование диска)

команда `mount` выводит очень много лишней информации, в которой теряется основная полезная информация. гораздо нагляднее информацию о разделах и дисках выводит команда `df -h` (`df` означает disk free)

Кроме того, `df` отображает размер диска или раздела, а также кол-во занятого и свободного места.

Просмотр файлов

В отличии от windows у Unix отсутствует реестр, а все программы настраиваются при помощи конфигурационных текстовых файлов. Для редактирования текстовых файлов используются текстовые редакторы. Основным текстовым редактором в консоли Unix является `vi`. Но часто возникает необходимость рассмотреть файл без редактирования. Для этого используются pager-ы. Pager – программа для просмотра текстовых файлов. В Unix есть 2 основных pages: `more` и `less`. `More` лучше подходит для просмотра маленьких файлов, страница которых помещается в 1 экран. А `less` гораздо удобнее для больших файлов – с поиском, прокруткой и тд. `More` сразу закрывается и возвращает управление консоли. Это удобно, когда файл помещается в 1 экран. А `less` блокирует консоль и в `less` гораздо удобнее выполнять поиск с конца файла чем в `more`. Но `less` ужасно неудобен, когда при просмотре маленького файла блокируется консоль. В остальном интерфейсы команды обеих программ одинаковые. Прокручивать файлы можно стрелками или `pageup/pagedown`. А выполнять поиск – слэшом либо вопросительным знаком. Но следует сказать, что `more` и `less` по-разному используют память.

`More` сразу же загружает файл в память, а `less` – только видимую его часть и смежные оформления. Открытие больших файлов pager-ом `more` может привести к зависанию системы.

Пользователи

Unix – это изначально многозадачная и многопользовательская ОС. В системе могут работать одновременно сколько угодно пользователей. У каждого пользователя есть домашний каталог, расположенный в каталоге `/home`. В нем находится профиль юзера его настройки и личные данные.

Главным пользователем в UNIX является `root`

`Root` – встроенный администратор, пользователь, которому можно абсолютно все. Переключаться между пользователями можно при помощи команды `su username` (`su-switch user`)

Введенная без параметров команда su переключает пользователя на root. В ubuntu команду su можно выполнить только от имени root-а, то есть при помощи команды sudo.

sudo - позволяет выполнить любую команду от имени root-а

Отобразить какие пользователи вошли в систему можно командами w или who.

Информация обо всех пользователях системы хранится в файле /etc/passwd.

Пользователь root может задать пароль любому пользователю при помощи команды passwd имя пользователя. Введенная без параметров эта команда задает пароль текущему пользователю. Ctrl+d завершает сеанс работы пользователя.

Процессы

Для того чтобы посмотреть список процессов используется команда ps. (процессы текущего пользователя)

ps ax – все процессы

Поиск процессов и других вещей в системе очень сильно облегчает фильтр grep.

grep PATTERNS [SOURCE]

где: PATTERNS – искомая строка

SOURCE – источник или множество данных, которым нужно найти эту строку. В качестве источника может выступать файл или вывод любой программы.

ps aux – также выводит имя пользователя, который породил процесс

ps acuh – вместо полного пути к исполняемому файлу выводит просто его имя

часто возникает необходимость удалить процесс как правило из-за того, что он завис. Это можно сделать командой kill или killall

kill – удаляет процесс по его PI (Process Identifier)

killall – удаляет процесс по имени

Список процессов в реальном времени можно отобразить командой top

Администрирование пользователей и групп.

Для администрирования пользователей в Linux используются команды useradd, usermod (изменяет учетную запись т пользователя), userdel, usershow, passwd (меняет пароль).

Для администрирования групп используются команды groupadd, groupmod, groupdel

useradd и usermod поддерживает практически одинаковый набор ключей. Все пользователи в системе хранятся в файле /etc/passwd и создавать пользователей

можно редактируя этот файл в текстовом редакторе, но делать это настоятельно не рекомендуются, поскольку в системе есть специальные команды (записаны выше). Информация о группах хранится в файле /etc/group

```
useradd uswer -G users -m
```

```
usermod uswer -s /bin/bash
```

```
less /etc/passwd
```

```
ls -alh
```

Права доступа

Абсолютно у каждого файла в UNIX есть владелец файла, группа файла и права доступа. Все эти свойства отображаются в длинном формате командой ls. Первым столбиком отображаются тип файла и права доступа в формате t rwx rwx rwx.

t – type (тип файла). Возможные значения: d – directory, - – regular file, l – symbolic link, p – named pipe

rwx – owner (права владельца). Возможные значения:

rwx – group (права группы), назначенной файлу

rwx – other (права всех остальных пользователей).

r – read (чтение)

w – write (запись)

x – execite (выполнение)

В памяти права доступа хранятся в виде трехразрядного восьмеричного числа, где старший разряд хранит права владельца, средний разряд хранит права группы и младший разряд – права всех остальных пользователей. Каждое действие занимает 1 бит. Права на файл можно назначить как комбинациями rwx так и трехразрядным восьмеричным числом.

Изменить владельца файла можно командой chown USER FILE

Изменить группу файла можно командой chgrp GROUP FILE

Изменить права доступа можно командой chmod MOD FILE

Работа с файлами

В ОС Unix файлы можно перемещать, копировать и тд.

Можно создать жесткие и символьные ссылки на файлы. Простые текстовые файлы можно создавать тестовым редактором или командой перенаправления в поток или же `get`.

`cat` – создает файл

`cat > File.txt`

Копировать файл можно командой `cp`

`cp file1 file2 fileN destination/`

`file1, file2, fileN` – копируемые файлы

`destination` – каталог назначения, в который нужно скопировать указанные файлы

Как правило копируют 1 файл, но 1 командой можно скопировать сколько угодно файлов.

Команда `cp` может копировать каталоги только с ключом `-r`:

`cp -r default/ ~`

Перемещать файлы можно с помощью команды `mv`:

`mv default/ /home/user/bkp/`

В UNIX нет команды переименования файлов. Для этих целей используется команда перемещения:

`mv bkp BackUp`

- Переименовывает файл `bkp` в `BackUp`

Полезные ключи при работе с файлами

Практически любая команда для работы с файлами поддерживает ключ `-r`, который означает Recursive. Этот ключ применяется к каталогам и применяет действия к самому каталогу так и к его содержимому (например, скопировать, удалить, сменить владельца и т.д.)

Ключ `-v` Verbose – выводит на экран подробную информацию

Ключ `-I` Interactive – По каждому файлу задает пользователю вопрос: действительно ли он хочет выполнить операцию над файлом.

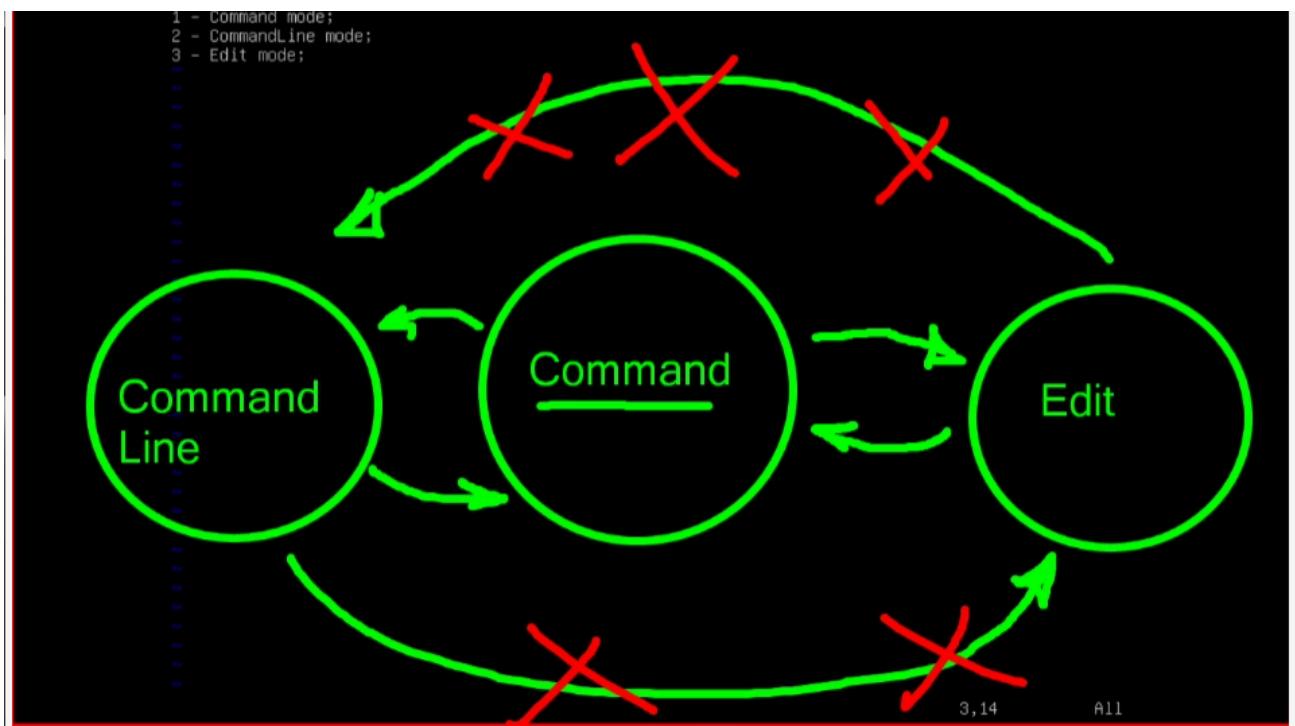
И практически любая команда UNIX поддерживает ключ `-f` Force, который принудительно выполняет действие.

`rm` – Удаляет файл

rm File.txt

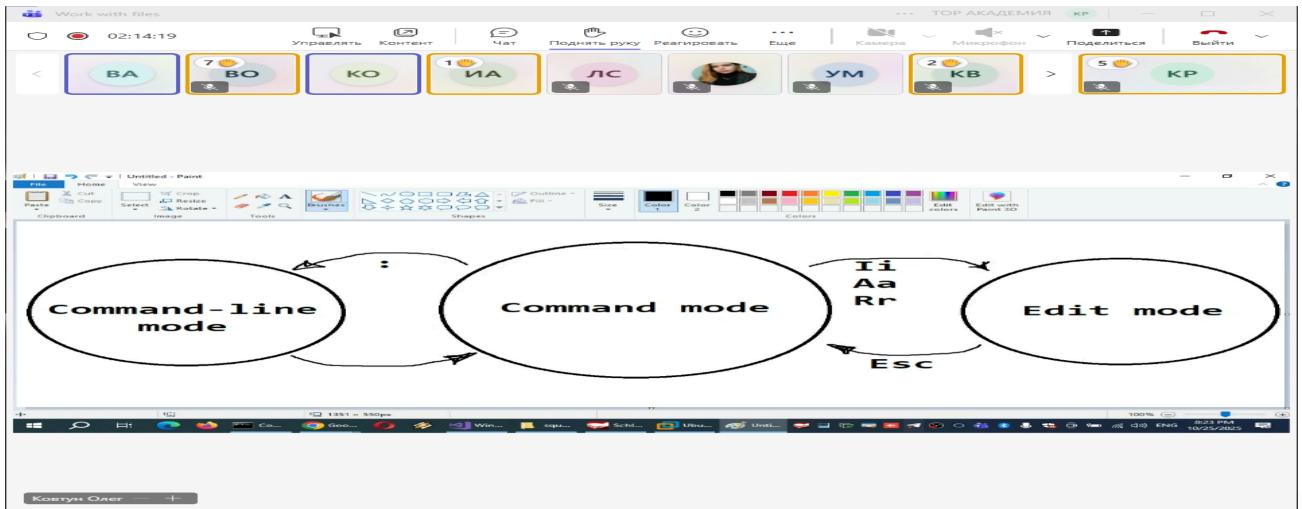
Текстовый редактор VI

VI – консольный режимный текстовый редактор, позволяющий создавать и редактировать текстовые файлы. Vi поддерживает 3 режима работы: 1 – командный режим, 2 – режим командной строки, 3 – режим редактирования. В командном режиме редактор запускается по умолчанию. В этом режиме выполняется перемещение по тексту, поиск по тексту. Это основной режим работы. Из него можно выйти как в режим командной строки, так и в режим редактирования.



В режиме командной строки выполняются операции над файлами – сохранение, открытие, выход из редактора. Из командной строки в режим редактирования можно перейти при помощи клавиши `i(I)`, `a(A)`, `r(R)` – пока используем только `i`. Выйти из режима редактирования в командный режим – клавиша `Esc`. Из командного режима перейти в

режим командной строки - клавиша двоеточие (:)



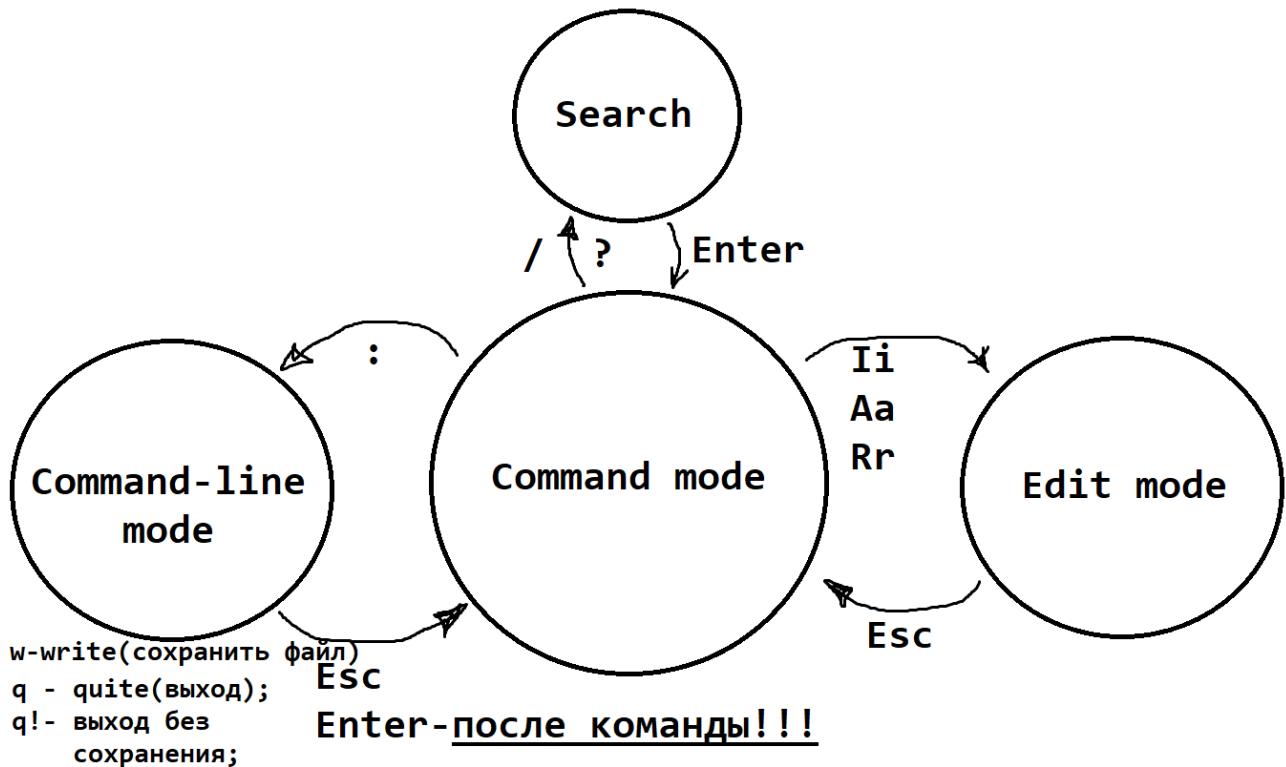
Выйти из командной строки в командный режим **esc** или **enter** после команды.

Команды режима командной строки:

w – write (сохранить файл)

q – quite (выход)

q! – выход без сохранения



w – на 1 слово вперед

b – на 1 слово назад

{, } – на 1 абзац вперед/назад

[,] – двойной клик – в начало или в конец файла

x - удалить

u - вернуть

Ctrl + z – свернуть панель

fg - развернуть

d 2 раза – удаляет строку

d }

d]]

Любая команда удаления вырезает удаляемый объект в буфер.

p – вставить из буфера

y y – копирует строку

Настройка сети в Unix

! ниже приведена настройка для временной работы сети. В следующем подразделе описывается постоянная настройка.

Просмотреть сетевую конфигурацию сервера можно при помощи команды ifconfig.

apt install net-tools

ping 192.168.0.1

ping 8.8.8.8

netstat -rn4 – таблица маршрутизаций

route -A inet del 0.0.0.0 – удаляет маршрут

Важная команда: route -A inet add -net default gw 192.168.0.1 – добавить шлюз по умолчанию

Настройка DNS сервера

more /etc/resolv.conf

<https://putty.org/index.html>

Сетевые настройки, сделанные таким образом сохраняются только до перезагрузки системы. Для того, чтобы сделать постоянные сетевые настройки, параметры нужно внести в конфигурационный текстовый файл.

netplan apply

`traceroute -nw 1 8.8.8.8` – трасировка маршрута

Сетевые конфигурационные файлы находятся в каталоге /etc/netplan/ и называются по принципу 50-cloud-init.yaml. Эти файлы считаются сетевым конфигуратором в порядке числового префикса и применяются настройки с наибольшим числовым префиксом. То есть если есть файлы 50-cloud-init.yaml и 99-cloud-init.yaml, то настройки применяются из файла 99-cloud-init.yaml. Файл с префиксом 50 перезаписывается после каждой перезагрузки системы. Для применения сетевых настроек после изменения конфигурационного файла используется команда netplan apply.

Туториал по настройки сети на Ubuntu

<https://documentation.ubuntu.com/server/explanation/networking/configuring-networks/>

Облегчение дальнейшего ввода (дописывает команду основываясь из истории)

- ## 1. Копируем файл inputrc:

```
cd /etc  
cp inputrc inputrc.default
```

2. Редактируем файл:

```
vi /etc/inputrc
```

меняем в 41 и 42 строках

должно получиться так:

```
12 # set convert-meta off  
13  
14 # try to enable the application keypad when it is called. Some systems  
15 # need this to enable the arrow keys.  
16 # set enable-keypad on  
17  
18 # see /usr/share/doc/bash/inputrc.arrows for other codes of arrow keys  
19  
20 # do not bell on tab-completion  
21 # set bell-style none  
22 # set bell-style visible  
23  
24 # some defaults / modifications for the emacs mode  
25 $if mode=emacs  
26  
27 # allow the use of the Home/End keys  
28 "\e[1~":" beginning-of-line  
29 "\e[4~":" end-of-line  
30  
31 # allow the use of the Delete/Insert keys  
32 "\e[3~":" delete-char  
33 "\e[2~":" quoted-insert  
34  
35 # mappings for "page up" and "page down" to step to the beginning/end  
36 # of the history  
37 # "\e[5~":" beginning-of-history  
38 # "\e[6~":" end-of-history  
39  
40 # alternate mappings for "page up" and "page down" to search the history  
41 "\e[A": history-search-backward  
42 "\e[B": history-search-forward  
43  
44 # mappings for Ctrl-left-arrow and Ctrl-right-arrow for word moving  
45 "\e[1;5C": forward-word  
46 "\e[1;5D": backward-word  
47 "\e[5C": forward-word  
:set number
```

1. Устанавливаем net-tools:

```
apt install net-tools
```

2. Устанавливает ip-адрес для ens33 с помощью команды ifconfig ens33 inet 192.168.0.22

```
root@server:/home/user# ifconfig
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet 192.168.0.108 netmask 255.255.255.0 broadcast 192.168.0.255
                inet6 fe80::20c:29ff:febf:56a3 prefixlen 64 scopeid 0x20<link>
                    ether 00:0c:29:bf:56:a3 txqueuelen 1000 (Ethernet)
                    RX packets 840 bytes 621825 (621.8 KB)
                    RX errors 0 dropped 1 overruns 0 frame 0
                    TX packets 286 bytes 30066 (30.0 KB)
                    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
        inet 127.0.0.1 netmask 255.0.0.0
                inet6 ::1 prefixlen 128 scopeid 0x10<host>
                    loop txqueuelen 1000 (Local Loopback)
                    RX packets 110 bytes 10193 (10.1 KB)
                    RX errors 0 dropped 0 overruns 0 frame 0
                    TX packets 110 bytes 10193 (10.1 KB)
                    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@server:/home/user# ifconfig ens33 inet 192.168.0.22
root@server:/home/user# _
```

3. Проверяем сеть

4. Добавляем шлюз по умолчанию с помощью команды:

```
route -A inet add -net default gw 192.168.0.1
```

Проверяем сеть

```
root@server:/home/user# route -A inet add -net default gw 192.168.0.1
root@server:/home/user# ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=108 time=27.3 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=108 time=28.4 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=108 time=28.3 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=108 time=27.9 ms
64 bytes from 8.8.8.8: icmp_seq=5 ttl=108 time=28.2 ms
64 bytes from 8.8.8.8: icmp_seq=6 ttl=108 time=28.2 ms
^C
--- 8.8.8.8 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5004ms
rtt min/avg/max/mdev = 27.329/28.040/28.383/0.346 ms
root@server:/home/user#
```

To direct input to this VM, click inside or press Ctrl+G.



Настройка DNS-сервера

1. Вводим команды:

```
root@server:/# cd /etc/
root@server:/etc# cp resolv.conf resolv.conf.default
root@server:/etc# ls -ld res*
lwxrwxrwx 1 root root 39 Sep 11 2024 resolv.conf -> ../../run/systemd/resolve/stub-resolv.conf
-rw-r--r-- 1 root root 920 Nov 1 12:11 resolv.conf.default
root@server:/etc#
```

To direct input to this VM, click inside or press Ctrl+G.



```
root@server:/etc# cat > /etc/resolv.conf
nameserver 8.8.8.8

^C
root@server:/etc# more /etc/resolv.conf
nameserver 8.8.8.8

root@server:/etc# _
```

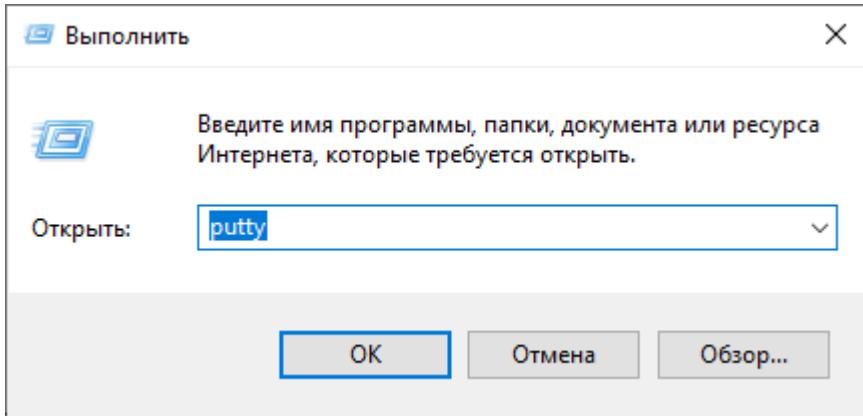
To direct input to this VM, click inside or press Ctrl+G.



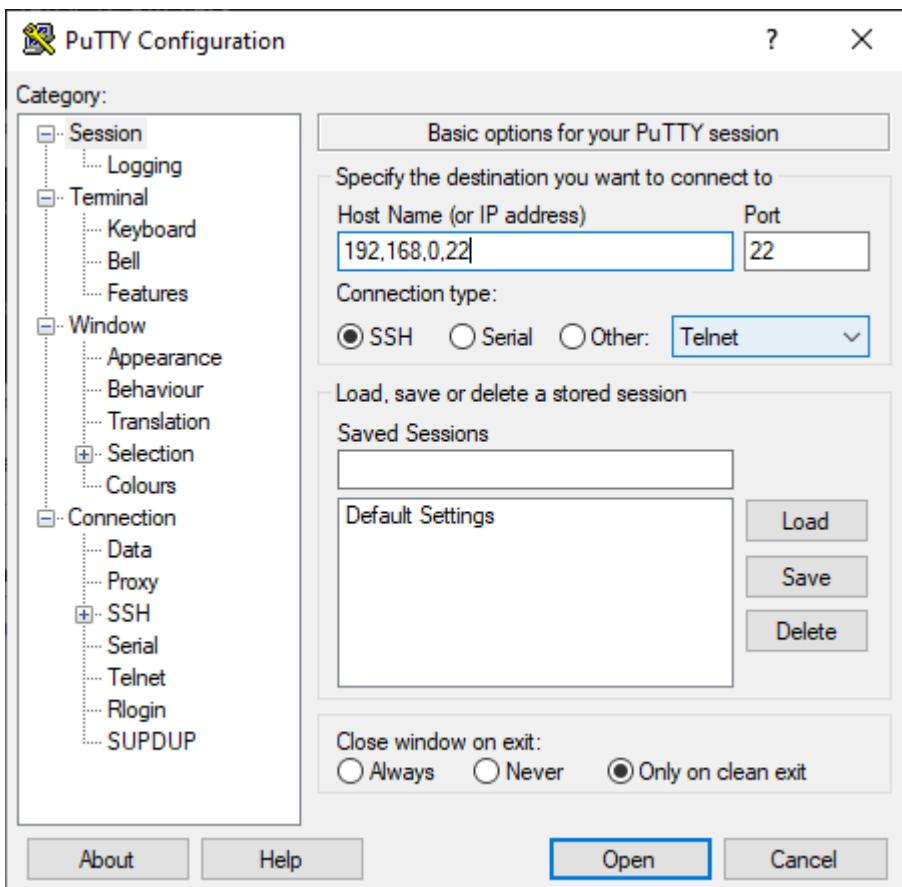
2. Скачиваем putty и переносим этот файл в папку Windows на диске:

<https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html>

3. Открываем putty через пуск -> ВЫПОЛНИТЬ



Подключаемся к SSH-серверу:



4. После этого нажимаем в следующем окне Accept

5. Входим сначала под user, а затем под root

* полезно: ПКМ на верхней панели – создать новый терминал

The screenshot shows a terminal window with a black background and white text. The terminal title is "root@server: /home/user". The content of the terminal is a series of shell commands numbered 28 to 50:

```
28 su uswer
29 sudo user
30 su user
31 su uswer
32 #
33 ls -lh
34 chmod u+x "Hello world"
35 ls -lh
36 cd files
37 ls -lh
38 cp new_file.txt
39 cp new_file.txt files/
40 cp new_file.txt ../files/
41 shutdown -h 0
42 cd /etc
43 cp inputrc inputrc.default
44 ls -dl input*
45 vi /etc/inputrc
46 exit
47 history
48 vi /etc/inputrc
49 exit
50 history
root@server:/home/user#
```

A context menu is open on the right side of the terminal window. The menu items are:

- Восстановить
- Переместить
- Размер
- Свернуть
- Развернуть
- Закрыть** Alt+F4
- Copy
- Paste
- Special Command >
- Event Log
- New Session...
- Duplicate Session**
- Saved Sessions >
- Change Settings...
- Copy All to Clipboard
- Clear Scrollback
- Reset Terminal

6. Перезагружаем с помощью команды:

init 6

7. Входим под user -> root в виртуалке и проверяем, что настройки слетели:

```
UbuntuHome - VMware Workstation
File Edit View VM Tabs Help || Home UbuntuHome FreeBSD Ubuntu PV_319
https://ubuntu.com/engage/secure-kubernetes-at-the-edge
Expanded Security Maintenance for Applications is not enabled.

58 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

New release '24.04.3 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

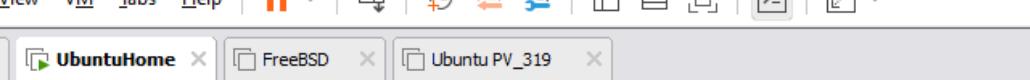
Last login: Sat Nov  1 12:23:37 UTC 2025 from 192.168.0.107 on pts/2
user@server:~$ sudo su
[sudo] password for user:
root@server:/home/user# ifconfig
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
      inet 192.168.0.108  netmask 255.255.255.0  broadcast 192.168.0.255
      inet6 fe80::20c:29ff:febf:56a3  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:bf:56:a3  txqueuelen 1000  (Ethernet)
          RX packets 26  bytes 3234 (3.2 KB)
          RX errors 0  dropped 0  overruns 0  frame 0
          TX packets 23  bytes 2764 (2.7 KB)
          TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
      inet 127.0.0.1  netmask 255.0.0.0
      inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
          RX packets 84  bytes 6480 (6.4 KB)
          RX errors 0  dropped 0  overruns 0  frame 0
          TX packets 84  bytes 6480 (6.4 KB)
          TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

root@server:/home/user#
To direct input to this VM, click inside or press Ctrl+G.
```

Настройка постоянных настроек

1. Копируем файл 50-cloud-init.yaml



```
root@server:/etc/netplan# cd /etc/netplan/
root@server:/etc/netplan# ls -l
total 4
-rw----- 1 root root 390 Nov  1 12:28 50-cloud-init.yaml
root@server:/etc/netplan# cp 50-cloud-init.yaml 5-cloud-init.yaml
root@server:/etc/netplan# ls -l
total 8
-rw----- 1 root root 390 Nov  1 12:28 50-cloud-init.yaml
-rw----- 1 root root 390 Nov  1 12:36 5-cloud-init.yaml
root@server:/etc/netplan#
```

2. Переходим в файл с помощью:

vi 50-cloud-init.yaml

3. Меняем содержимое файлы на следующее:

4. Проверяем соединение:

ping 8.8.8.8

5. Устанавливаем команду traceroute:

```
apt install traceroute
```

6. Трасируем маршрут с помощью traceroute -nw 1 8.8.8.8

```
root@server:/etc/netplan# traceroute -nw 1 8.8.8.8
traceroute to 8.8.8.8 (8.8.8.8), 30 hops max, 60 byte packets
 1  192.168.0.1  0.355 ms  0.431 ms  0.393 ms
 2  * * *
 3  109.194.8.30  4.436 ms  4.623 ms  4.590 ms
 4  72.14.215.165  11.665 ms *  11.809 ms
 5  * * 188.234.131.225  12.360 ms
 6  * * *
 7  108.170.226.90  14.273 ms 108.170.225.36  9.987 ms  10.246 ms
 8  192.178.241.232  34.932 ms * *
 9  142.251.238.82  28.084 ms 142.250.238.138  25.171 ms  24.112 ms
10  72.14.232.86  24.250 ms 142.251.238.72  34.638 ms 108.170.235.64  25.177 ms
11  142.250.208.25  24.864 ms 142.250.56.221  28.462 ms 172.253.64.57  24.699 ms
12  * * *
13  * * *
14  * * *
15  * * *
16  * * *
17  * * *
18  * * *
19  * * *
20  * * *
21  8.8.8.8  27.128 ms  27.012 ms  24.600 ms
root@server:/etc/netplan#
```

7. Выполняем команду:

```
cp 50-cloud-init.yaml 99-cloud-init.yaml
```

! Создаем копию файла из-за того, что файл 50-cloud-init.yaml перезаписывается после каждой перезагрузки системы

8. После изменения конфигурационного файла необходимо выполнить команду:

```
netplan apply
```

9. Перезагружаем виртуалку и проверяем, что настройки применились:

```
shutdown -r 0
```

```
Ifconfig
```

```
UbuntuHome - VMware Workstation
File Edit View VM Tabs Help || Home FreeBSD Ubuntu PV_319

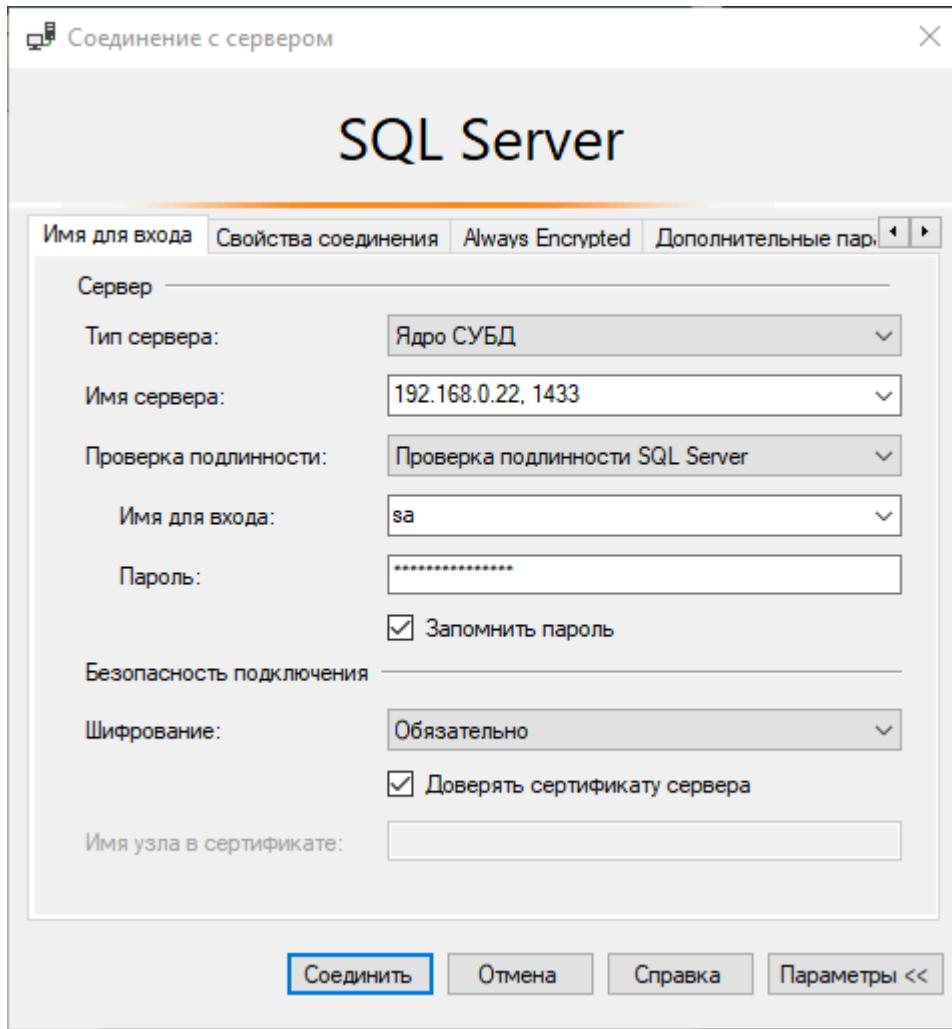
root@server:/etc/netplan# ifconfig
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.0.22 netmask 255.255.255.0 broadcast 192.168.0.255
        inet6 fe80::20c:29ff:febf:56a3 prefixlen 64 scopeid 0x20<link>
            ether 00:0c:29:bf:56:a3 txqueuelen 1000 (Ethernet)
                RX packets 21 bytes 2886 (2.8 KB)
                RX errors 0 dropped 0 overruns 0 frame 0
                TX packets 21 bytes 2538 (2.5 KB)
                TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
        inet6 ::1 prefixlen 128 scopeid 0x10<host>
            loop txqueuelen 1000 (Local Loopback)
                RX packets 84 bytes 6480 (6.4 KB)
                RX errors 0 dropped 0 overruns 0 frame 0
                TX packets 84 bytes 6480 (6.4 KB)
                TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
root@server:/etc/netplan#
```

Установка MS SQL Server

<https://learn.microsoft.com/en-us/sql/linux/quickstart-install-connect-ubuntu?view=sql-server-ver17&tabs=ubuntu2004%2C2025ubuntu2204%2Codbc-ubuntu-2004>

При выполнении команды sudo /opt/mssql/bin/mssql-conf setup в шаге 4 нужно выбрать 4 версию.

Подключение к SSMS:



Установка приложений в Unix-системах

В Unix есть 2 основных способа установки приложений: установка из пакетов (packages) и установка из портов. Пакеты – это уже скомпилированные бинарные файлы, упакованные в архив. Установка из пакетов представляет собой распаковку бинарных и конфигурационных текстовых файлов в определенные папки на диске. Порты – это ссылки на исходные файлы приложения в Интернете. При установке из портов эти исходные файлы загружаются из Интернета на локальный компьютер, выполняется их компиляция и после завершения компиляции происходит такая же установка как из пакета. То есть полученные бинарные файлы копируются в соответствующие папки на диске. Установка из-за компиляции установка из портов занимает гораздо больше времени и вычислительных ресурсов, но лучше адаптирует приложение под конкретную систему. Очень часто при установке из портов также собирается соответствующий пакет. Пакеты также как и порты чаще всего скачиваются из Интернета.

`apt list | grep mssql` - вывод доступных пакетов в репозиториях (| grep mssql – выводит по пакету mssql)

Установка ASP.NET

<https://learn.microsoft.com/en-us/dotnet/core/install/linux-scripted-manual#scripted-install>

*поэтому туториалу установить не получилось. Вот рабочие команды:

`sudo apt update`

`sudo apt install -y dotnet-sdk-8.0`

проверяем что установилось:

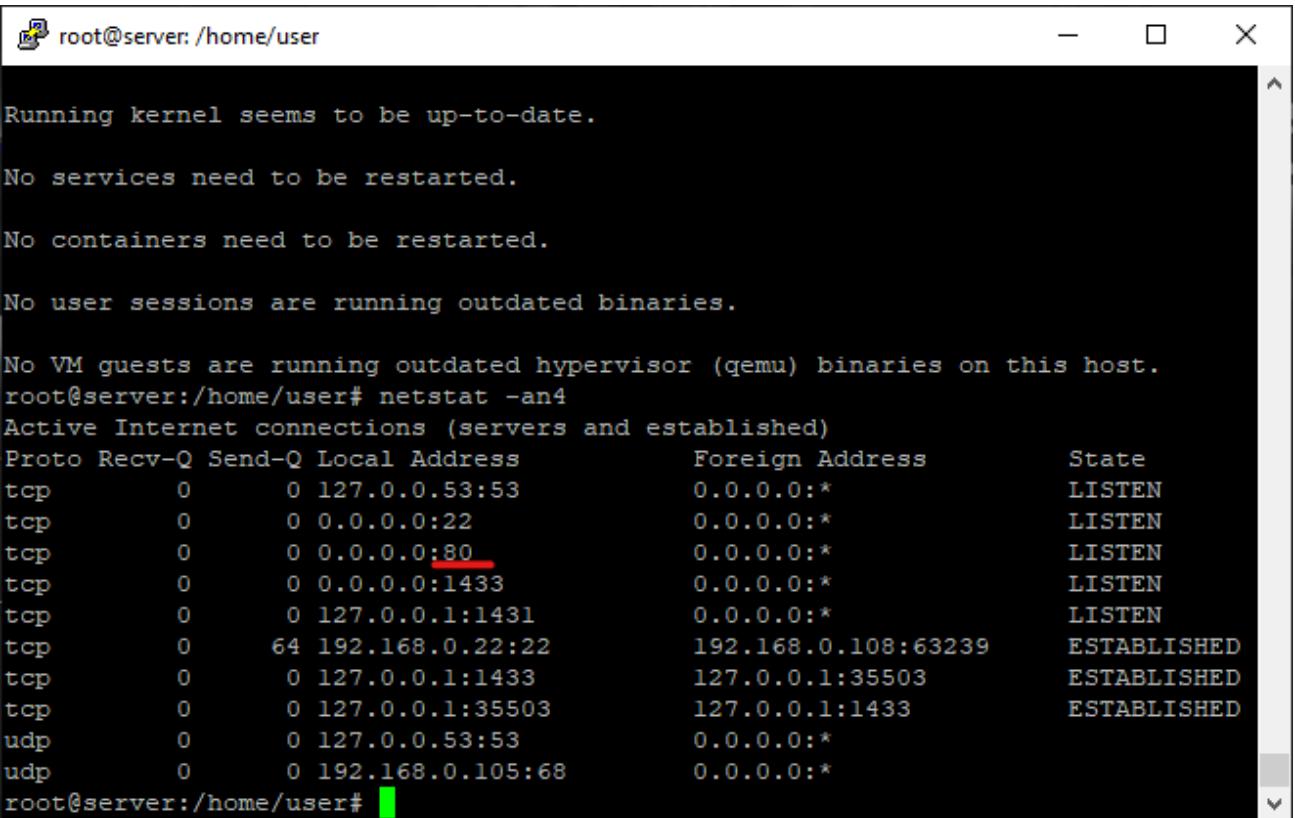
`dotnet --info`

Установка nginx:

Под root вводим: `apt-get install nginx`

Проверяем что установилось:

Вводим команду `netstat -an4` и должен появиться 80 порт:



```
root@server: /home/user
Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
root@server:/home/user# netstat -an4
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address          Foreign Address        State
tcp      0      0 127.0.0.53:53            0.0.0.0:*              LISTEN
tcp      0      0 0.0.0.0:22              0.0.0.0:*              LISTEN
tcp      0      0 0.0.0.0:80              0.0.0.0:*              LISTEN
tcp      0      0 0.0.0.0:1433            0.0.0.0:*              LISTEN
tcp      0      0 127.0.0.1:1431            0.0.0.0:*              LISTEN
tcp      0      64 192.168.0.22:22         192.168.0.108:63239   ESTABLISHED
tcp      0      0 127.0.0.1:1433            127.0.0.1:35503       ESTABLISHED
tcp      0      0 127.0.0.1:35503           127.0.0.1:1433       ESTABLISHED
udp      0      0 127.0.0.53:53            0.0.0.0:*              LISTEN
udp      0      0 192.168.0.105:68          0.0.0.0:*              LISTEN
root@server:/home/user#
```

Хостинг asp-приложения:

1. Находим строку подключения к базе через visual studio:

The screenshot shows the Visual Studio IDE interface with the following details:

- File Menu:** File, Edit, View, Git, Project, Build, Debug, Test, Analyze, Tools, Extensions.
- Search Bar:** Search for "Intr...lazor".
- Toolbox:** Standard icons for file operations.
- Project Selector:** BlazorAcademy, https://github.com/BlazorAcademy/IntroductionToBlazor.
- Solution Explorer:** Shows "Index.razor" as the active file.
- SQL Server Object Explorer:** A sidebar with options like "Add SQL Server...", "Data Comparison...", "Refresh", and "Properties".
- Code Editor:** The "launchSettings.json" file is open, showing configuration for IIS Express and a development profile named "Project".

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <launchSettings>
3      <profile name="IIS Express">
4          <sdkVersion>6.0.100-rc.1-20020-00</sdkVersion>
5          <commandName>Project</commandName>
6          <environmentVariables>
7              <environmentVariable name="ASPNETCORE_ENVIRONMENT" value="Development" />
8          </environmentVariables>
9      </profile>
10     <profile name="Project">
11         <commandName>Project</commandName>
12         <environmentVariables>
13             <environmentVariable name="ASPNETCORE_ENVIRONMENT" value="Development" />
14         </environmentVariables>
15         <iisExpress>
16             <applicationUrl>http://localhost:53015</applicationUrl>
17             <sslPort>44397</sslPort>
18         </iisExpress>
19     </profile>
20     <profile name="Development" >
21         <commandName>Project</commandName>
22         <environmentVariables>
23             <environmentVariable name="ASPNETCORE_ENVIRONMENT" value="Development" />
24         </environmentVariables>
25     </profile>
26     <profile name="Production" >
27         <commandName>Project</commandName>
28         <environmentVariables>
29             <environmentVariable name="ASPNETCORE_ENVIRONMENT" value="Production" />
30         </environmentVariables>
31     </profile>
32 
```
- Error List:** Shows 0 Errors, 0 Warnings, and 0 Messages.
- Output:** Shows "Ready".

 Connect X

History **Browse**

Type here to filter the list

► Local
► Network
► Azure

Server Name:

Authentication:

User Name:

Password:

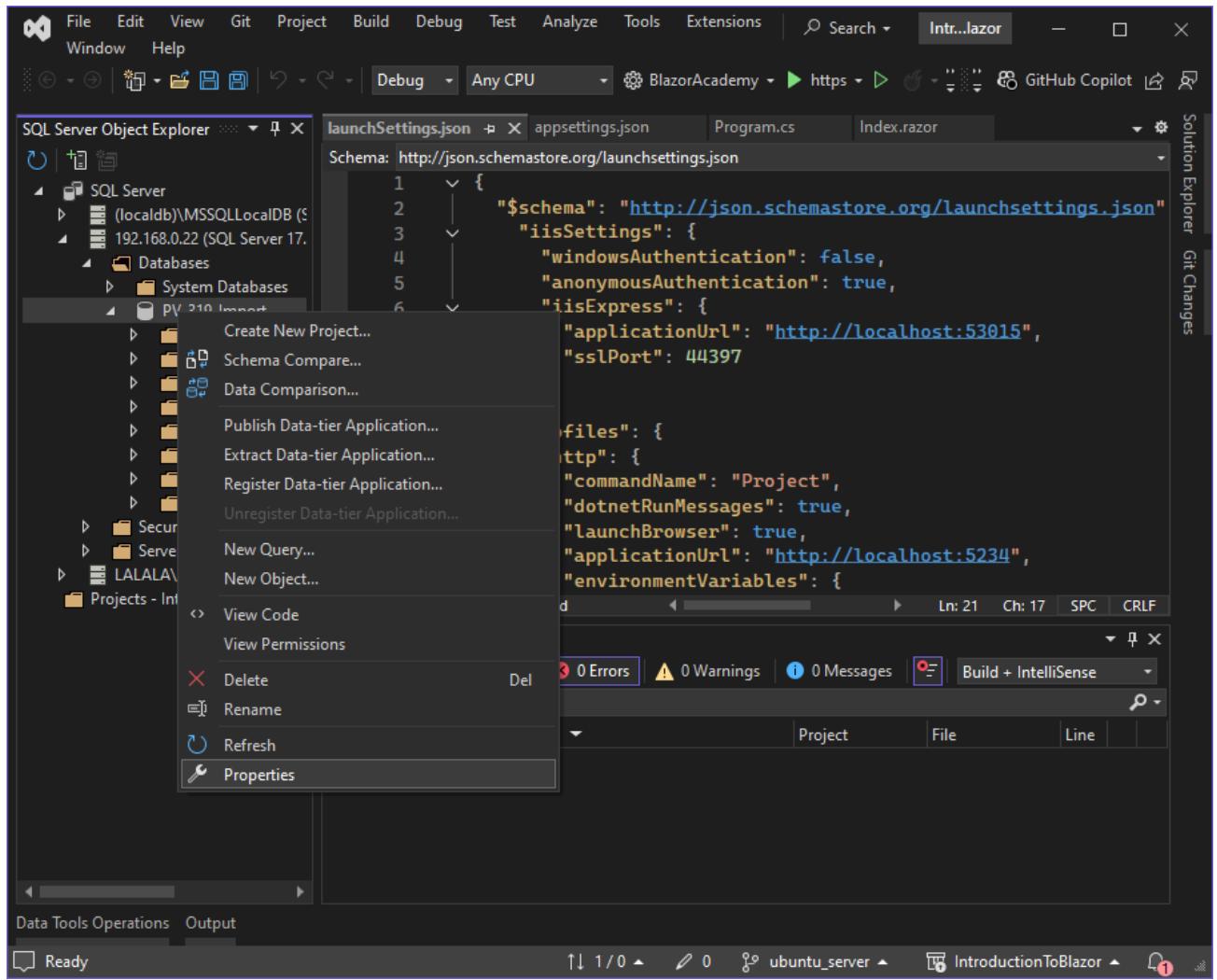
Remember Password

Database Name:

Encrypt:

Trust Server Certificate:

[Advanced...](#)



Копируем строку.

Открываем файл appsettings.json и вставляем строку подключения туда. Также пишем в ней пароль.

The screenshot shows the Visual Studio Code interface with the 'appsettings.json' file open. The file contains configuration settings for a Blazor application, including logging levels and connection strings for different contexts.

```
Schema: https://www.schemastore.org/appsettings.json
{
  "Logging": {
    "LogLevel": {
      "Default": "Information",
      "Microsoft.AspNetCore": "Warning"
    }
  },
  "AllowedHosts": "*",
  "ConnectionStrings": {
    "BlazorAcademyContext": "Data Source=ALALA\SQLEXPRESS;Initial Catalog=PV_319_Import;Integrated Security=True;Connect Timeout=30;Encrypt=True;Trust Server Certificate=True;Application Intent=ReadWrite;Multi Subnet Failover=False",
    "BlazorAcademyContext": "Data Source=192.168.0.22;Initial Catalog=PV_319_Import;User ID=sa;Password=0$4$rd;Connect Timeout=30;Encrypt=True;Trust Server Certificate=True;Application Intent=ReadWrite;Multi Subnet Failover=False",
    "BlazorAcademyContext": "Server=(localdb)\mssqllocaldb;Database=BlazorAcademyContext-f0798831-8634-40c8-8881-df98c1109c7c;Trusted_Connection=True;MultipleActiveResultSets=true"
  }
}
```

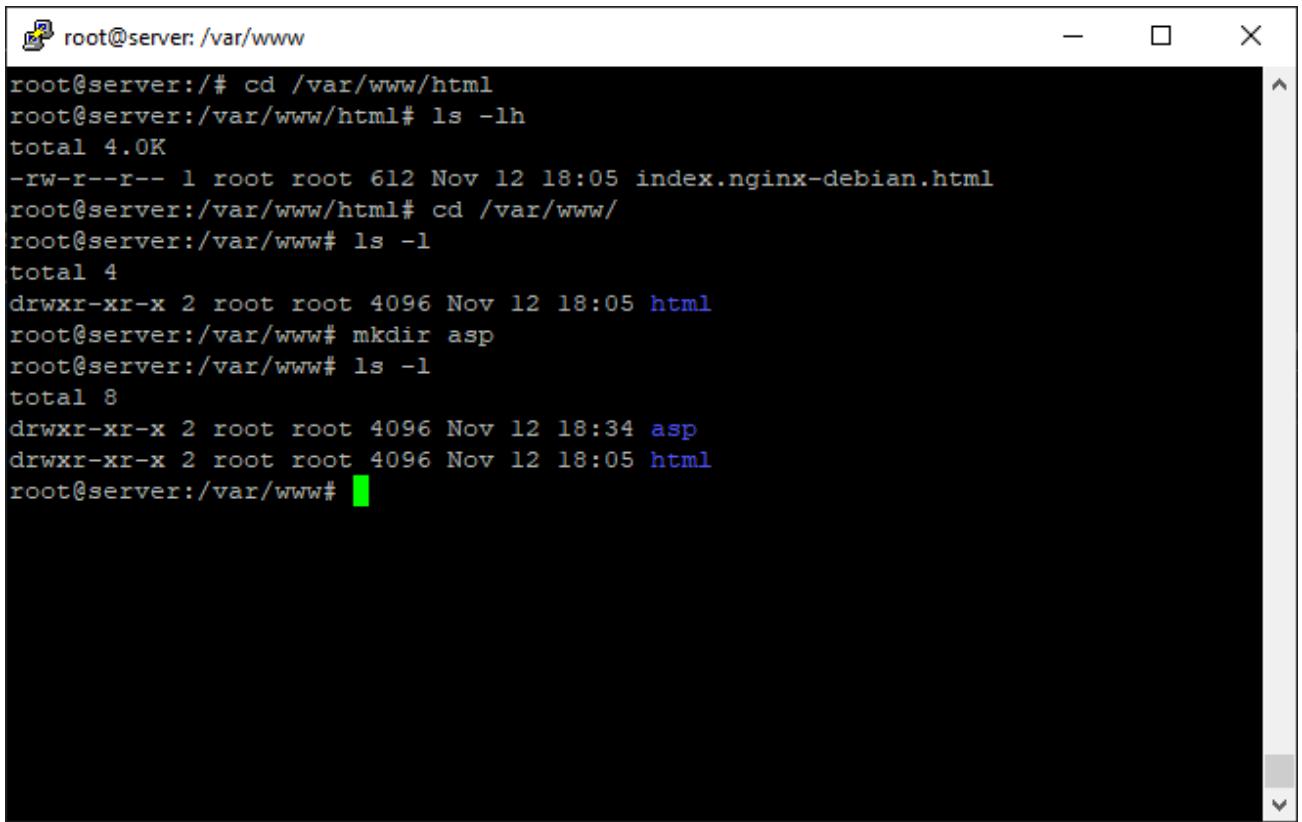
Проверяем что приложение работает. Ctrl + F5 в виртуалке.

Проверяем что приложение запущено на сервере (запущены порты 1433):

The screenshot shows a terminal window running as root on a server. The command 'netstat -an4' is executed, displaying a list of active network connections. The output shows several TCP connections established on port 1433, both incoming and outgoing.

```
root@server:/home/user# netstat -an4
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address          Foreign Address        State
tcp      0      0 127.0.0.53:53            0.0.0.0:*              LISTEN
tcp      0      0 0.0.0.0:22              0.0.0.0:*              LISTEN
tcp      0      0 0.0.0.0:80              0.0.0.0:*              LISTEN
tcp      0      0 0.0.0.0:1433             0.0.0.0:*              LISTEN
tcp      0      0 127.0.0.1:1431            0.0.0.0:*              LISTEN
tcp      0      64 192.168.0.22:22         192.168.0.108:63239   ESTABLISHED
tcp      0      0 127.0.0.1:1433             127.0.0.1:58823       ESTABLISHED
tcp      0      0 192.168.0.22:1433           192.168.0.108:64818   ESTABLISHED
tcp      0      0 127.0.0.1:58823            127.0.0.1:1433       ESTABLISHED
udp      0      0 127.0.0.53:53            0.0.0.0:*              LISTEN
udp      0      0 192.168.0.105:68           0.0.0.0:*              LISTEN
root@server:/home/user#
```

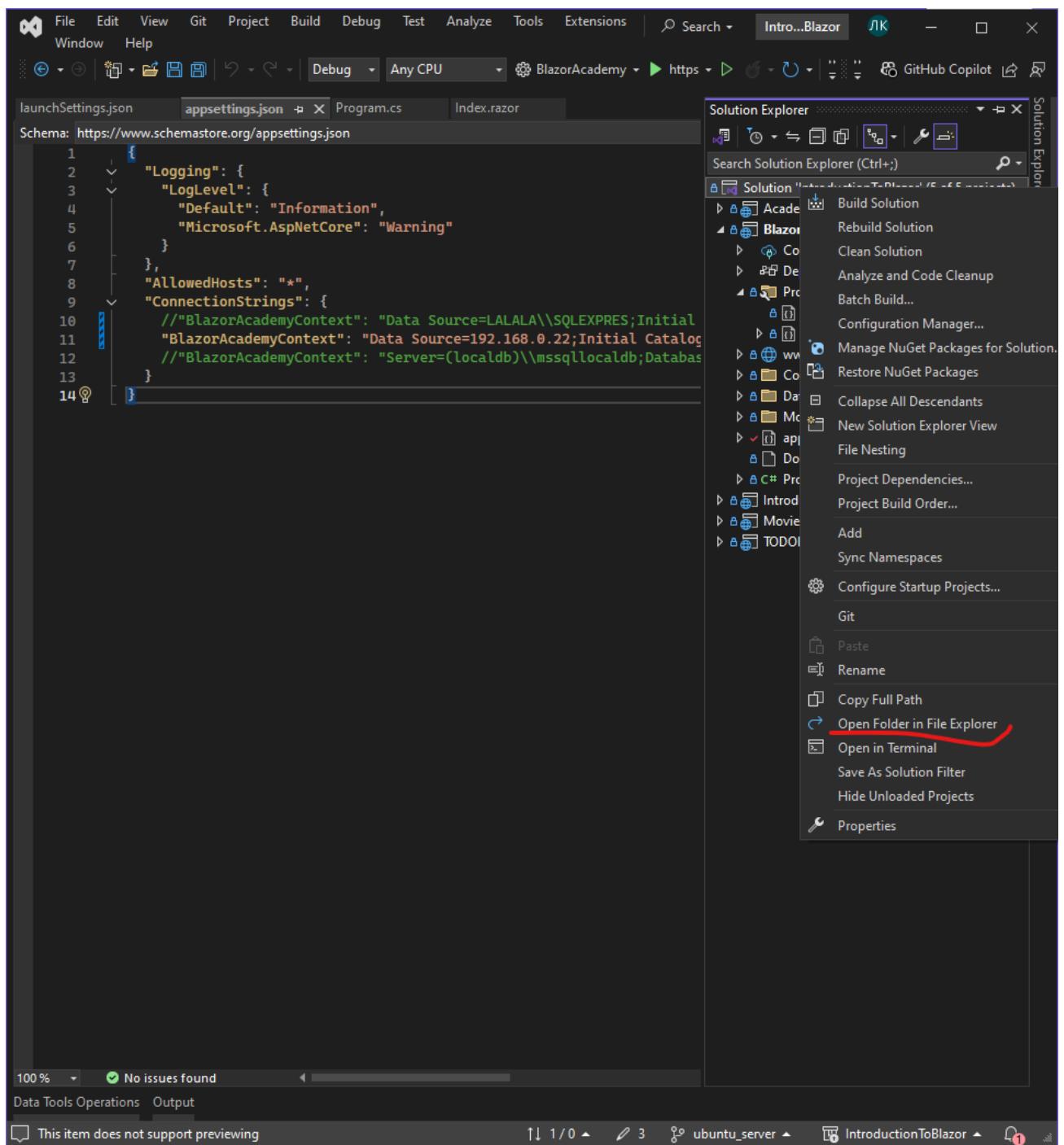
На сервере вводим:

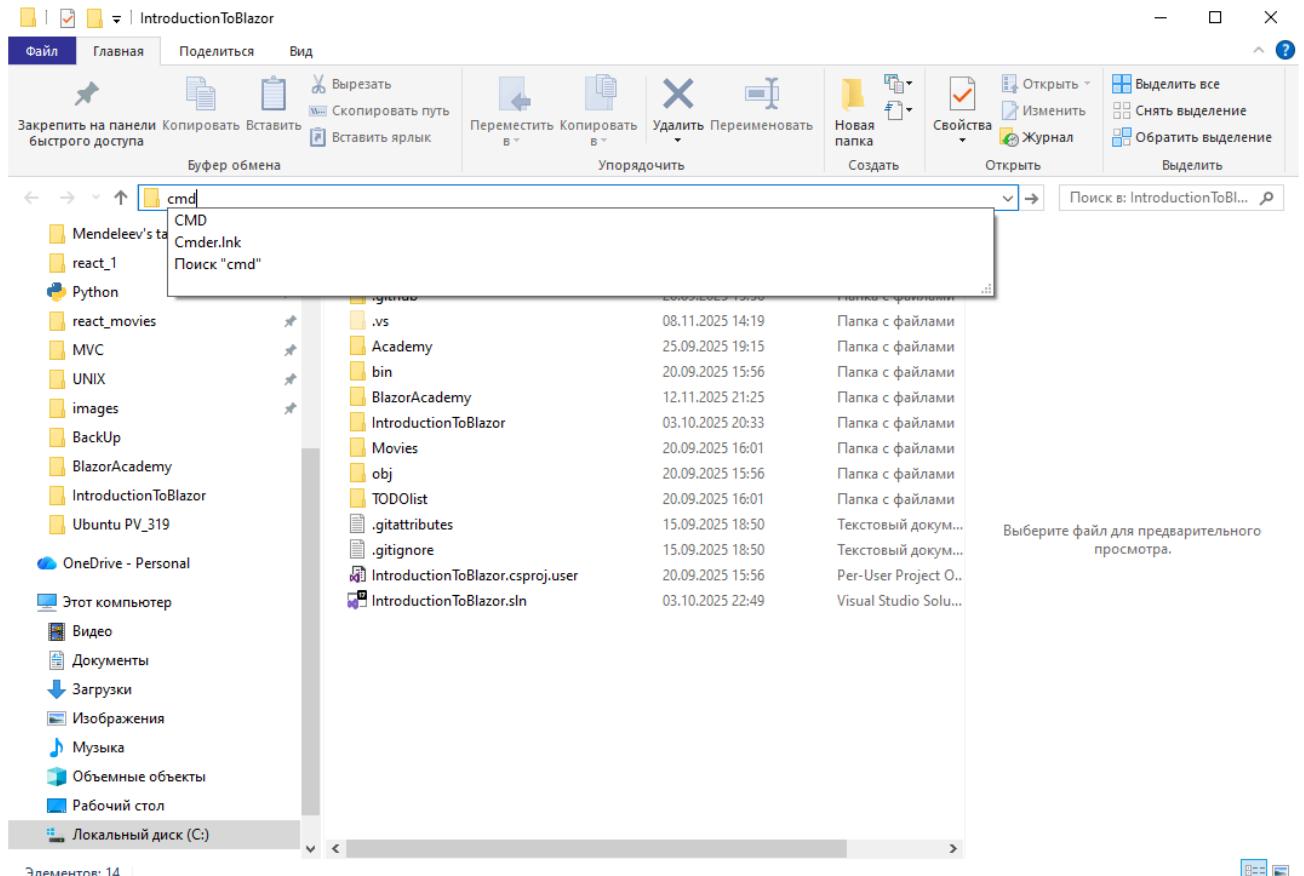


The screenshot shows a terminal window with the following command history:

```
root@server:/var/www# cd /var/www/html
root@server:/var/www/html# ls -lh
total 4.0K
-rw-r--r-- 1 root root 612 Nov 12 18:05 index.nginx-debian.html
root@server:/var/www/html# cd /var/www/
root@server:/var/www# ls -l
total 4
drwxr-xr-x 2 root root 4096 Nov 12 18:05 html
root@server:/var/www# mkdir asp
root@server:/var/www# ls -l
total 8
drwxr-xr-x 2 root root 4096 Nov 12 18:34 asp
drwxr-xr-x 2 root root 4096 Nov 12 18:05 html
root@server:/var/www#
```

Переходим в папку с проектом на компьютере:





```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19045.6456]
(c) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.

C:\Users\PC\source\repos\IntroductionToBlazor>scp
usage: scp [-346ABCOpqRrsTv] [-c cipher] [-D sftp_server_path] [-F ssh_config]
           [-i identity_file] [-J destination] [-l limit] [-o ssh_option]
           [-P port] [-S program] [-X sftp_option] source ... target

C:\Users\PC\source\repos\IntroductionToBlazor>
```

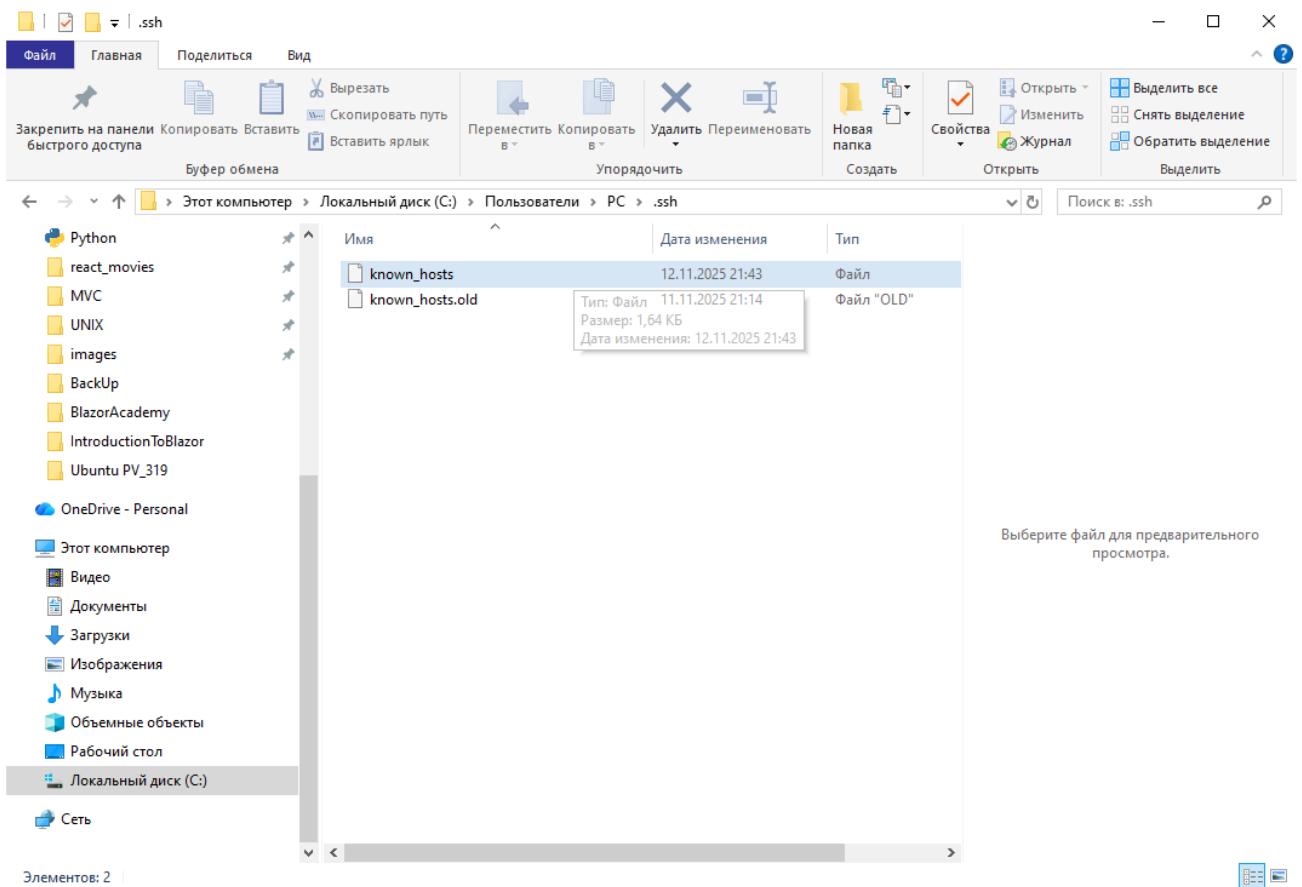
```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19045.6456]
(c) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.

C:\Users\PC\source\repos\IntroductionToBlazor>scp
usage: scp [-346ABCOpqRrsTv] [-c cipher] [-D sftp_server_path] [-F ssh_config]
           [-i identity_file] [-J destination] [-l limit] [-o ssh_option]
           [-P port] [-S program] [-X sftp_option] source ... target

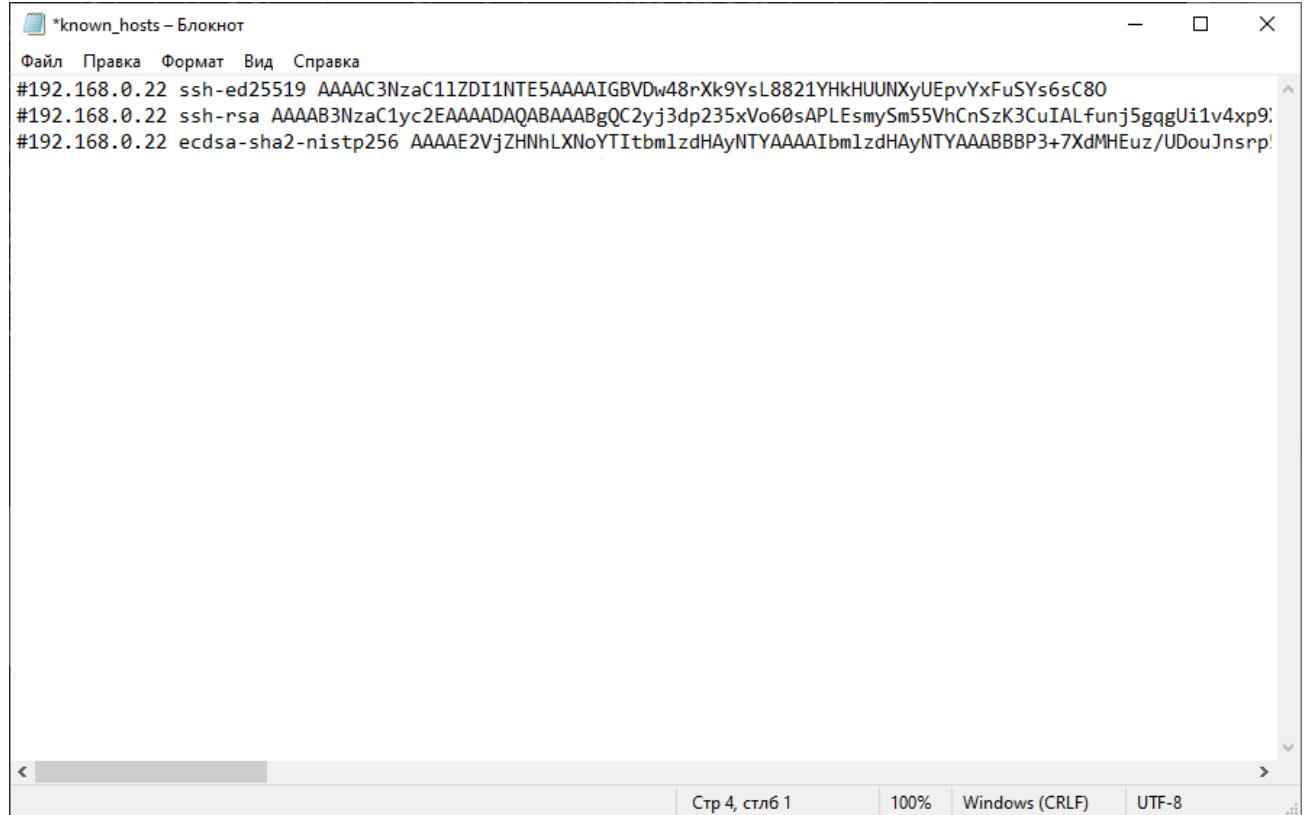
C:\Users\PC\source\repos\IntroductionToBlazor>scp -r BlazorAcademy root@192.168.0.22:/var/www/asp/
@@@@@@@
@    WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED!    @
@@@@@@@
IT IS POSSIBLE THAT SOMEONE IS DOING SOMETHING NASTY!
Someone could be eavesdropping on you right now (man-in-the-middle attack)!
It is also possible that a host key has just been changed.
The fingerprint for the ED25519 key sent by the remote host is
SHA256:GuMBVnx5AFAc9TrBHKNU16y67UX+z5GVE/vxSYh85Y.
Please contact your system administrator.
Add correct host key in C:\\\\Users\\\\PC\\\\.ssh\\\\known_hosts to get rid of this message.
Offending ECDSA key in C:\\\\Users\\\\PC\\\\.ssh\\\\known_hosts:6
Host key for 192.168.0.22 has changed and you have requested strict checking.
Host key verification failed.
scp: Connection closed

C:\Users\PC\source\repos\IntroductionToBlazor>
```

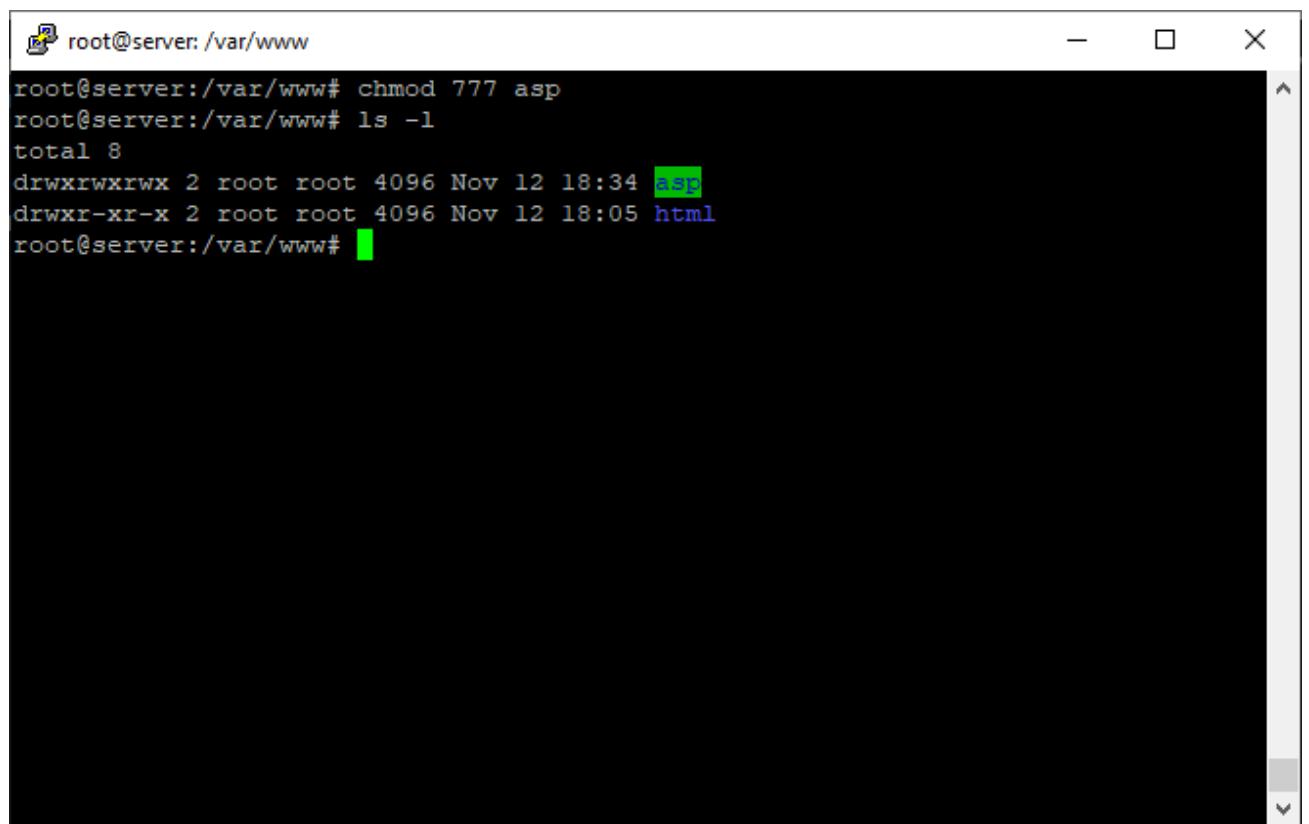
Для исправления ошибки открываем файл:



И перед строками ставим #:



```
*known_hosts - Блокнот
Файл Правка Формат Вид Справка
#192.168.0.22 ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAIGBVw48rXk9YsL8821YHkHUUNXyUEpvYxFuSYs6sC80
#192.168.0.22 ssh-rsa AAAAB3NzaC1yc2EAAAQABAAgQC2yj3dp235xVo60sAPLEsmySm55VhCnSzK3CuIALfunj5gqgUi1v4xp9;
#192.168.0.22 ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbm1zdHAyNTYAAAAIbm1zdHAyNTYAAABBP3+7XdMHEuz/UDouJnsrp!
```



```
root@server:/var/www
root@server:/var/www# chmod 777 asp
root@server:/var/www# ls -l
total 8
drwxrwxrwx 2 root root 4096 Nov 12 18:34 asp
drwxr-xr-x 2 root root 4096 Nov 12 18:05 html
root@server:/var/www#
```

Вводим следующую команду для копирования проекта на сервер:

```
C:\Windows\System32\cmd.exe
C:\Users\PC\source\repos\IntroductionToBlazor>scp -r BlazorAcademy user@192.168.0.22:/var/www/asp/
user@192.168.0.22's password:
appsettings.Development.json                                100%   127    124.0KB/s  00:00
appsettings.json                                             100%   837    817.3KB/s  00:00
Humanizer.resources.dll                                     100%   16KB    7.9MB/s  00:00
appsettings.Development.json                                100%   127     0.1KB/s  00:00
appsettings.json                                             100%   837     0.8KB/s  00:00
Humanizer.resources.dll                                     100%   21KB   10.3MB/s  00:00
Humanizer.resources.dll                                     100%   16KB   15.7MB/s  00:00
Azure.Core.dll                                              100%  392KB   54.6MB/s  00:00
Azure.Identity.dll                                         100%  335KB   65.4MB/s  00:00
Humanizer.resources.dll                                     100%   17KB   16.2MB/s  00:00
BlazorAcademy.deps.json                                    100% 122KB   19.9MB/s  00:00
BlazorAcademy.dll                                         100% 148KB   20.6MB/s  00:00
BlazorAcademy.exe                                         100% 136KB   26.5MB/s  00:00
BlazorAcademy.pdb                                         100% 126KB   30.8MB/s  00:00
BlazorAcademy.runtimeconfig.json                           100%   485     0.5KB/s  00:00
BlazorAcademy.staticwebassets.endpoints.json            100%   13KB    6.1MB/s  00:00
BlazorAcademy.staticwebassets.runtime.json              100% 1500     1.4MB/s  00:00
Humanizer.resources.dll                                     100%   17KB   16.2MB/s  00:00
Humanizer.resources.dll                                     100%   18KB   17.7MB/s  00:00
Microsoft.CodeAnalysis.CSharp.Features.resources.dll    100%   45KB   22.1MB/s  00:00
Microsoft.CodeAnalysis.CSharp.resources.dll             100% 412KB   50.3MB/s  00:00
Microsoft.CodeAnalysis.CSharp.Workspaces.resources.dll  100%   18KB    4.3MB/s  00:00
Microsoft.CodeAnalysis.Features.resources.dll           100% 187KB   30.4MB/s  00:00
Microsoft.CodeAnalysis.resources.dll                     100%   46KB   11.2MB/s  00:00
Microsoft.CodeAnalysis.Scripting.resources.dll         100%   18KB   17.4MB/s  00:00
Microsoft.CodeAnalysis.Workspaces.resources.dll        100%   47KB   23.0MB/s  00:00
Humanizer.resources.dll                                     100%   17KB    5.6MB/s  00:00
Humanizer.resources.dll                                     100%   19KB   18.7MB/s  00:00
```

```
root@server: /var/www/asp/BlazorAcademy
drwx----- 9 user user 4.0K Nov 12 18:52 BlazorAcademy
root@server:/var/www/asp# cd BlazorAcademy/
root@server:/var/www/asp/BlazorAcademy# dotnet run

Welcome to .NET 8.0!
-----
SDK Version: 8.0.121

-----
Installed an ASP.NET Core HTTPS development certificate.
To trust the certificate, view the instructions: https://aka.ms/dotnet-https-linux

-----
Write your first app: https://aka.ms/dotnet-hello-world
Find out what's new: https://aka.ms/dotnet-whats-new
Explore documentation: https://aka.ms/dotnet-docs
Report issues and find source on GitHub: https://github.com/dotnet/core
Use 'dotnet --help' to see available commands or visit: https://aka.ms/dotnet-cli
-----
-----
Building...
```

Должно получиться так:

```
root@server: /var/www/asp/BlazorAcademy
/var/www/asp/BlazorAcademy/Models/Human.cs(10,23): warning CS8618: Non-nullable
property 'last_name' must contain a non-null value when exiting constructor. Con
sider declaring the property as nullable. [/var/www/asp/BlazorAcademy/BlazorAcad
emy.csproj]
/var/www/asp/BlazorAcademy/Models/Human.cs(12,23): warning CS8618: Non-nullable
property 'first_name' must contain a non-null value when exiting constructor. Co
nsider declaring the property as nullable. [/var/www/asp/BlazorAcademy/BlazorAca
demacy.csproj]
/var/www/asp/BlazorAcademy/Models/Discipline.cs(14,23): warning CS8618: Non-nullable
property 'discipline_name' must contain a non-null value when exiting const
ructor. Consider declaring the property as nullable. [/var/www/asp/BlazorAcademy
/BlazorAcademy.csproj]
warn: Microsoft.AspNetCore.DataProtection.KeyManagement.XmlKeyManager[35]
      No XML encryptor configured. Key {e14af602-44db-49e9-9b59-e96d8025ebbl} ma
y be persisted to storage in unencrypted form.
info: Microsoft.Hosting.Lifetime[14]
      Now listening on: http://localhost:5234
info: Microsoft.Hosting.Lifetime[0]
      Application started. Press Ctrl+C to shut down.
info: Microsoft.Hosting.Lifetime[0]
      Hosting environment: Development
info: Microsoft.Hosting.Lifetime[0]
      Content root path: /var/www/asp/BlazorAcademy
```

В первом info запоминаем вот это: <http://localhost:5234>

Необязательный этап:

Заходим в visual Studio в launchSettings.json:

The screenshot shows the Visual Studio IDE interface. The main area displays the `launchSettings.json` file in the code editor. The file contains configuration for IIS Express, HTTP, and HTTPS launch profiles. The `applicationUrl` for HTTPS is currently set to `https://localhost:7273;http://localhost:52`. The Solution Explorer on the right shows a solution named 'IntroductionToBlazor' containing five projects: Academy, BlazorAcademy, IntroductionToBlazor, Movies, and TODOlist.

```
Schema: http://json.schemastore.org/launchsettings.json
1  {
2      "$schema": "http://json.schemastore.org/launchsettings.json",
3      "iisSettings": {
4          "windowsAuthentication": false,
5          "anonymousAuthentication": true,
6          "iisExpress": {
7              "applicationUrl": "http://localhost:53015",
8              "sslPort": 44397
9          }
10     },
11     "profiles": {
12         "http": {
13             "commandName": "Project",
14             "dotnetRunMessages": true,
15             "launchBrowser": true,
16             "applicationUrl": "http://localhost:5234",
17             "environmentVariables": {
18                 "ASPNETCORE_ENVIRONMENT": "Development"
19             }
20         },
21         "https": {
22             "commandName": "Project",
23             "dotnetRunMessages": true,
24             "launchBrowser": true,
25             "applicationUrl": "https://localhost:7273;http://localhost:52",
26             "environmentVariables": {
27                 "ASPNETCORE_ENVIRONMENT": "Development"
28             }
29         },
30         "IIS Express": {
31             "commandName": "IISExpress",
32             "launchBrowser": true,
33             "environmentVariables": {
34                 "ASPNETCORE_ENVIRONMENT": "Development"
35             }
36         }
37     }
38 }
```

Меняем в 16 и 25 строках:

The screenshot shows the Visual Studio IDE interface with the following details:

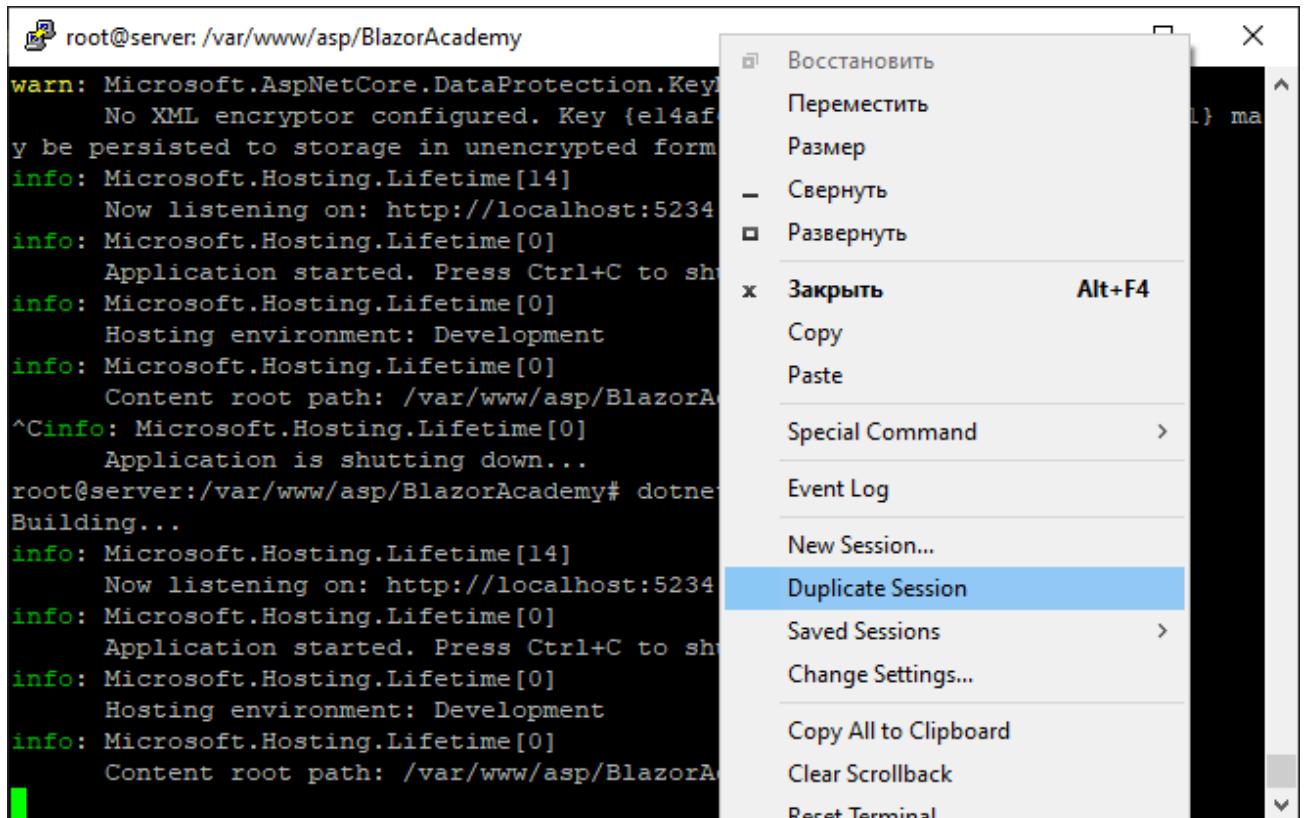
- Menu Bar:** File, Edit, View, Git, Project, Build, Debug, Test, Analyze, Tools, Extensions.
- Toolbar:** Standard icons for file operations like Open, Save, Print, etc.
- Search Bar:** Search field with a dropdown menu.
- Project Explorer:** Shows "BlazorAcademy" as the active project.
- Solution Explorer:** Shows "Solution Explorer" and "Git Changes".
- Code Editor:** The "launchSettings.json" file is open. The schema is defined as `$schema: "http://json.schemastore.org/launchsettings.json"`. It contains configurations for IIS Express, HTTP, and HTTPS. The IIS Express section is currently selected.
- Status Bar:** Shows "100%" zoom, "No issues found", line number "Ln: 30", character number "Ch: 23", and file endings "SPC CRLF".
- Taskbar:** Shows "Ready", "ubuntu_server", and "IntroductionToBlazor".

```
Schema: http://json.schemastore.org/launchsettings.json
1  {
2      "$schema": "http://json.schemastore.org/launchsettings.json",
3      "iisSettings": {
4          "windowsAuthentication": false,
5          "anonymousAuthentication": true,
6          "iisExpress": {
7              "applicationUrl": "http://localhost:53015",
8              "sslPort": 44397
9          }
10     },
11     "profiles": {
12         "http": {
13             "commandName": "Project",
14             "dotnetRunMessages": true,
15             "launchBrowser": true,
16             "applicationUrl": "http://0.0.0.0:5234",
17             "environmentVariables": {
18                 "ASPNETCORE_ENVIRONMENT": "Development"
19             }
20         },
21         "https": {
22             "commandName": "Project",
23             "dotnetRunMessages": true,
24             "launchBrowser": true,
25             "applicationUrl": "https://0.0.0.0:7273;http://0.0.0.0:5234",
26             "environmentVariables": {
27                 "ASPNETCORE_ENVIRONMENT": "Development"
28             }
29         }
30     },
31     "IIS Express": {
32         "commandName": "IISExpress",
33         "launchBrowser": true,
34         "environmentVariables": {
35             "ASPNETCORE_ENVIRONMENT": "Development"
36         }
37     }
38 }
```

*Этот шаг необязательный, т. к. сервер этот файл не видит, а он нужен по сути только там.

Обязательный шаг:

```
root@server:/var/www/asp/BlazorAcademy
warn: Microsoft.AspNetCore.DataProtection.Key
      No XML encryptor configured. Key {e14af
y be persisted to storage in unencrypted form
info: Microsoft.Hosting.Lifetime[14]
      Now listening on: http://localhost:5234
info: Microsoft.Hosting.Lifetime[0]
      Application started. Press Ctrl+C to sh
info: Microsoft.Hosting.Lifetime[0]
      Hosting environment: Development
info: Microsoft.Hosting.Lifetime[0]
      Content root path: /var/www/asp/BlazorA
^Cinfo: Microsoft.Hosting.Lifetime[0]
      Application is shutting down...
root@server:/var/www/asp/BlazorAcademy# dotne
Building...
info: Microsoft.Hosting.Lifetime[14]
      Now listening on: http://localhost:5234
info: Microsoft.Hosting.Lifetime[0]
      Application started. Press Ctrl+C to sh
info: Microsoft.Hosting.Lifetime[0]
      Hosting environment: Development
info: Microsoft.Hosting.Lifetime[0]
      Content root path: /var/www/asp/BlazorA
```



A context menu is open on the right side of the terminal window. The menu items include: Восстановить (Restore), Переместить (Move), Размер (Size), Свернуть (Minimize), Развернуть (Maximize), Закрыть (Close Alt+F4), Copy, Paste, Special Command >, Event Log, New Session..., Duplicate Session (highlighted in blue), Saved Sessions >, Change Settings..., Copy All to Clipboard, Clear Scrollback, and Reset Terminal.

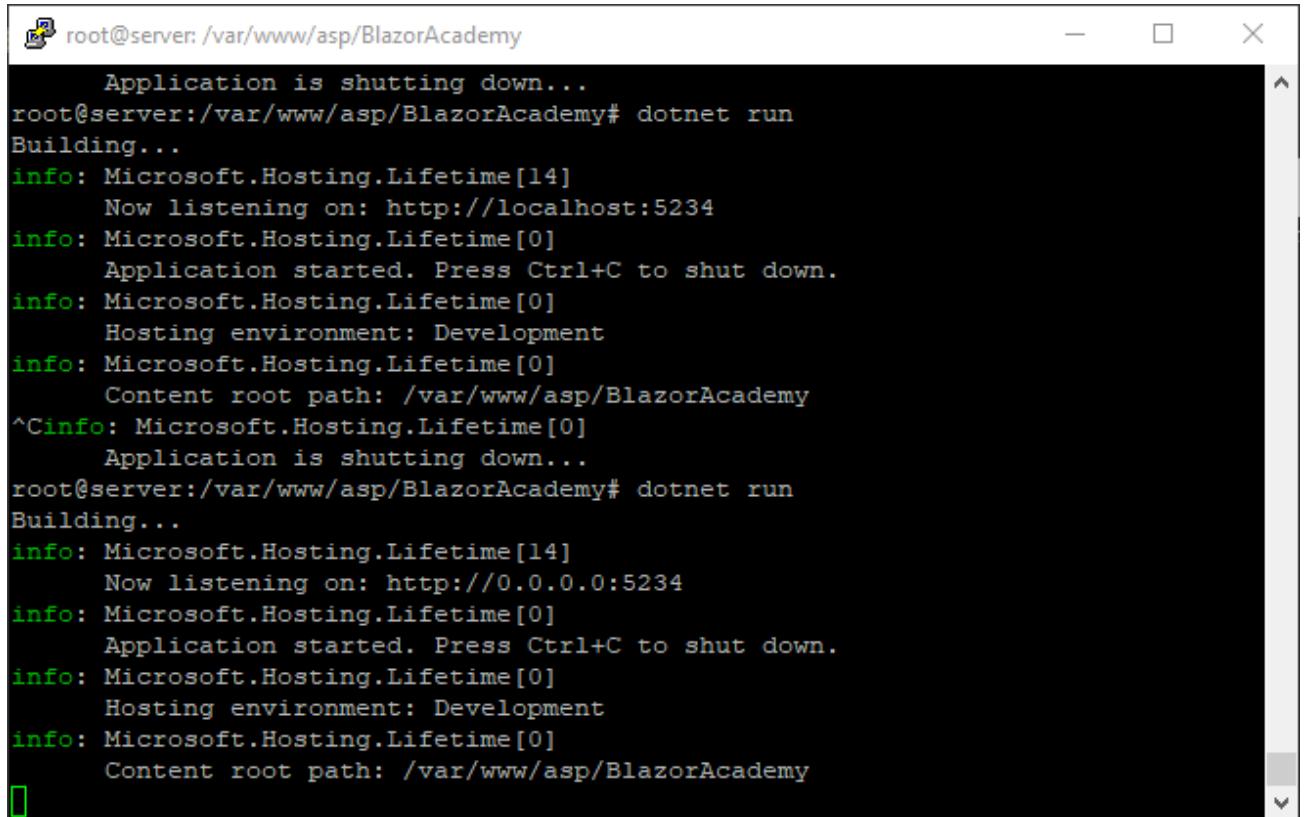
```
root@server:/var/www/asp/BlazorAcademy/Properties
total 52K
-rw-rw-r-- 1 user user 127 Nov 12 18:51 appsettings.Development.json
-rw-rw-r-- 1 user user 837 Nov 12 18:51 appsettings.json
drwx----- 4 user user 4.0K Nov 12 18:52 bin
-rw-rw-r-- 1 user user 927 Nov 12 18:52 BlazorAcademy.csproj
-rw-rw-r-- 1 user user 585 Nov 12 18:52 BlazorAcademy.csproj.user
drwx----- 4 user user 4.0K Nov 12 18:52 Components
drwx----- 2 user user 4.0K Nov 12 18:52 Data
-rw-rw-r-- 1 user user 348 Nov 12 18:52 Dockerfile
drwx----- 2 user user 4.0K Nov 12 18:52 Models
drwx----- 4 user user 4.0K Nov 12 18:54 obj
-rw-rw-r-- 1 user user 1.3K Nov 12 18:52 Program.cs
drwx----- 2 user user 4.0K Nov 12 18:52 Properties
drwx----- 3 user user 4.0K Nov 12 18:52 wwwroot
root@server:/var/www/asp/BlazorAcademy# cd Pro
bash: cd: Pro: No such file or directory
root@server:/var/www/asp/BlazorAcademy# cd Properties/
root@server:/var/www/asp/BlazorAcademy/Properties# ls -lh
total 16K
-rw-rw-r-- 1 user user 1.1K Nov 12 18:52 launchSettings.json
-rw-rw-r-- 1 user user 142 Nov 12 18:52 serviceDependencies.json
-rw-rw-r-- 1 user user 148 Nov 12 18:52 serviceDependencies.local.json
-rw-rw-r-- 1 user user 153 Nov 12 18:52 serviceDependencies.local.json.user
root@server:/var/www/asp/BlazorAcademy/Properties#
```

```
root@server:/var/www/asp/BlazorAcademy/Properties
root@server:/var/www/asp/BlazorAcademy/Properties# vi launchSettings.json
```

Также редактируем как в visual studio:

```
root@server:/var/www/asp/BlazorAcademy/Properties
8         "sslPort": 44397
9     }
10    },
11    "profiles": {
12      "http": {
13        "commandName": "Project",
14        "dotnetRunMessages": true,
15        "launchBrowser": true,
16        "applicationUrl": "http://0.0.0.0:5234",
17        "environmentVariables": {
18          "ASPNETCORE_ENVIRONMENT": "Development"
19        }
20      },
21      "https": {
22        "commandName": "Project",
23        "dotnetRunMessages": true,
24        "launchBrowser": true,
25        "applicationUrl": "https://0.0.0.0:7273;http://0.0.0.0:5234",
26        "environmentVariables": {
27          "ASPNETCORE_ENVIRONMENT": "Development"
28        }
29      },
30      "IIS Express": {
-- INSERT --
```

Опять прерываем работу сервера и запускаем заново:

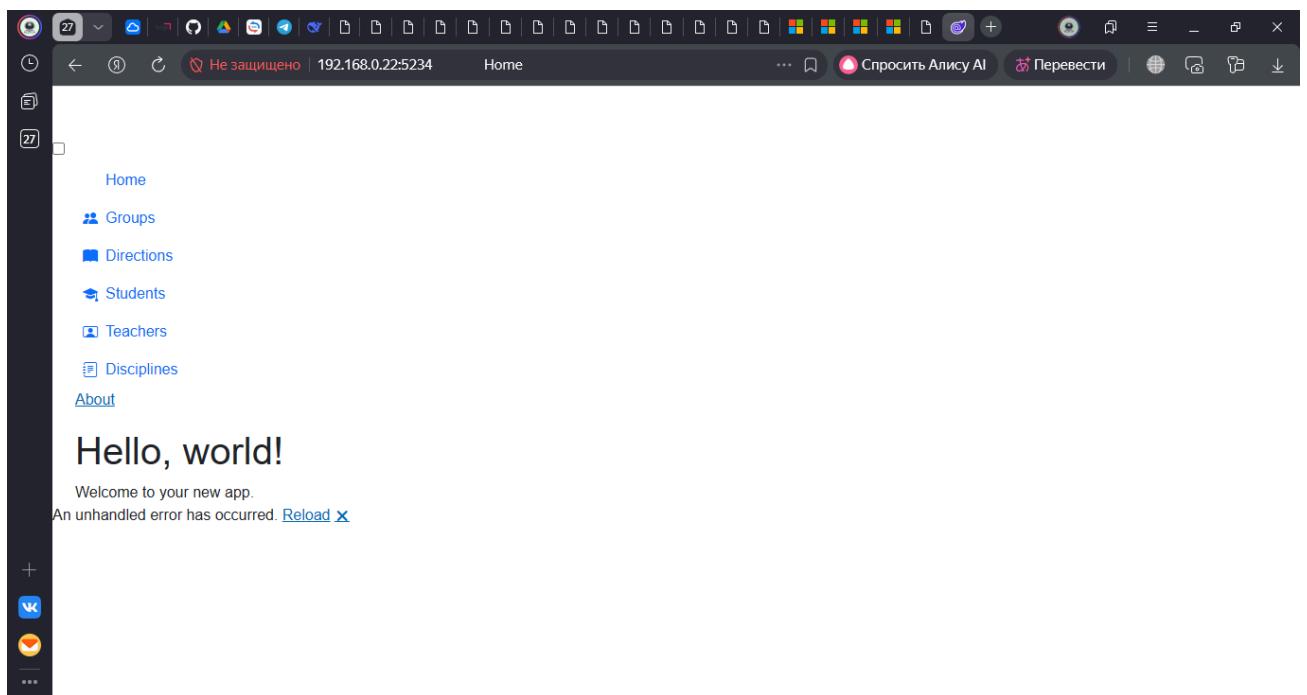


```
root@server:/var/www/asp/BlazorAcademy
Application is shutting down...
root@server:/var/www/asp/BlazorAcademy# dotnet run
Building...
info: Microsoft.Hosting.Lifetime[14]
      Now listening on: http://localhost:5234
info: Microsoft.Hosting.Lifetime[0]
      Application started. Press Ctrl+C to shut down.
info: Microsoft.Hosting.Lifetime[0]
      Hosting environment: Development
info: Microsoft.Hosting.Lifetime[0]
      Content root path: /var/www/asp/BlazorAcademy
^Cinfo: Microsoft.Hosting.Lifetime[0]
      Application is shutting down...
root@server:/var/www/asp/BlazorAcademy# dotnet run
Building...
info: Microsoft.Hosting.Lifetime[14]
      Now listening on: http://0.0.0.0:5234
info: Microsoft.Hosting.Lifetime[0]
      Application started. Press Ctrl+C to shut down.
info: Microsoft.Hosting.Lifetime[0]
      Hosting environment: Development
info: Microsoft.Hosting.Lifetime[0]
      Content root path: /var/www/asp/BlazorAcademy
```

Готово! Пишем в браузере:

192.168.0.22:5234

последние 4 цифры зависят от ранее полученных (выше в конспекте это обозначено как нужно запомнить)



Настройка css-стилей:

На новом терминале переходим в папку с проектом и вводим команды `dotnet clean` и `dotnet build`

```
root@server:/var/www/asp/BlazorAcademy# cd /var/www/asp/BlazorAcademy/
root@server:/var/www/asp/BlazorAcademy# dotnet clean
MSBuild version 17.8.43+f0ccb1397 for .NET
Build started 11/20/2025 12:44:19 AM.
  1>Project "/var/www/asp/BlazorAcademy/BlazorAcademy.csproj" on node 1 (Clean target(s)).
    1>CoreClean:
```

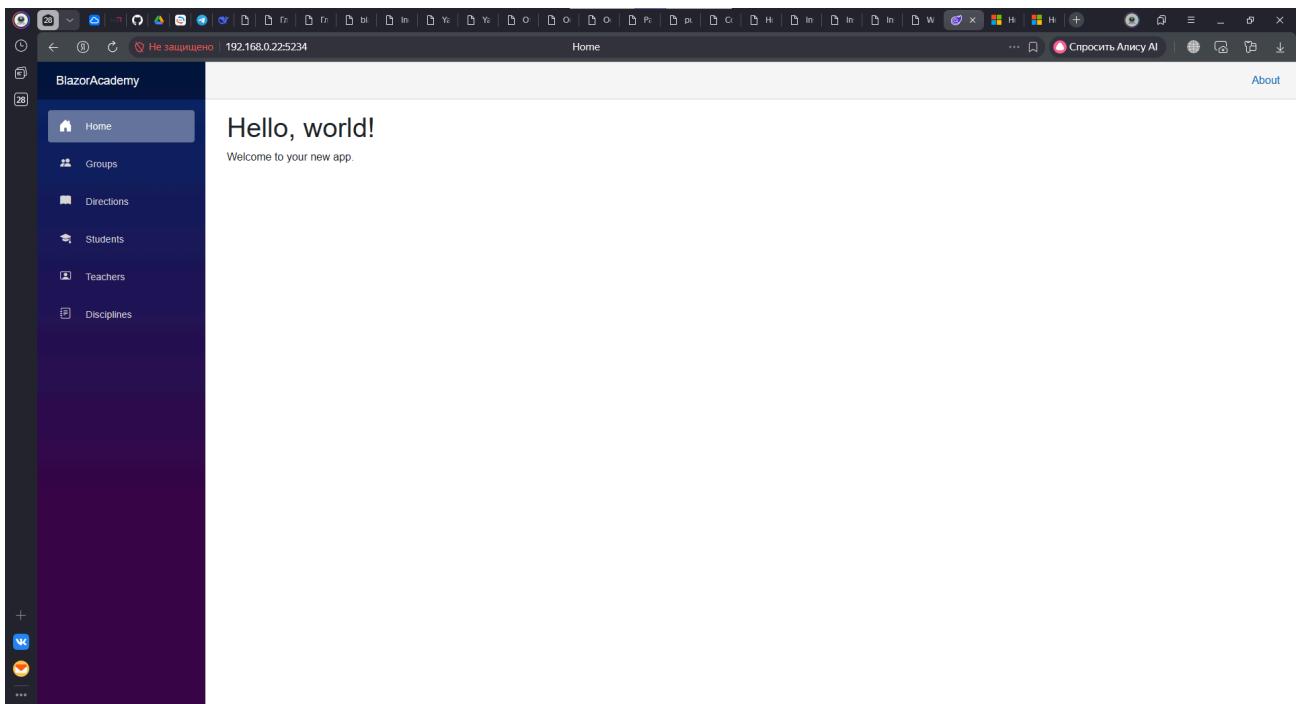
```
root@server:/var/www/asp/BlazorAcademy
1>Done Building Project "/var/www/asp/BlazorAcademy/BlazorAcademy.csproj" (Clean target(s)).

Build succeeded.
  0 Warning(s)
  0 Error(s)

Time Elapsed 00:00:00.53
root@server:/var/www/asp/BlazorAcademy# dotnet build
MSBuild version 17.8.43+f0ccb1397 for .NET
  Determining projects to restore...
  All projects are up-to-date for restore.
/var/www/asp/BlazorAcademy/Models/Discipline.cs(14,23): warning CS8618: Non-nullable property 'discipline_name' must contain a non-null value when exiting constructor. Consider declaring the property as nullable. [/var/www/asp/BlazorAcademy/BlazorAcademy.csproj]
/var/www/asp/BlazorAcademy/Models/Human.cs(10,23): warning CS8618: Non-nullable property 'last_name' must contain a non-null value when exiting constructor. Consider declaring the property as nullable. [/var/www/asp/BlazorAcademy/BlazorAcademy.csproj]
/var/www/asp/BlazorAcademy/Models/Human.cs(12,23): warning CS8618: Non-nullable
```

(сам сервер можно не останавливать)

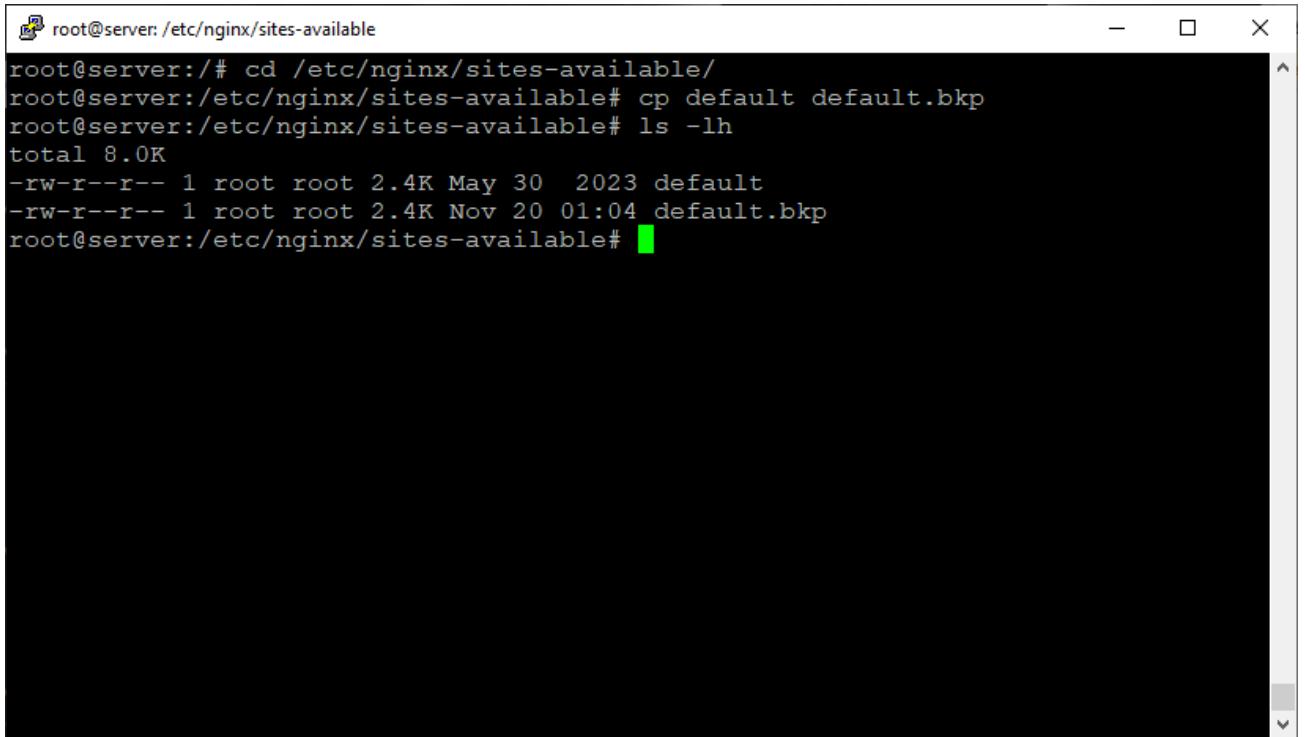
By-a-ля! Все заработало.



Настройка подключения без порта при вводе (вместо 192.168.0.22:порт можно будет вводить просто 192.168.0.22)

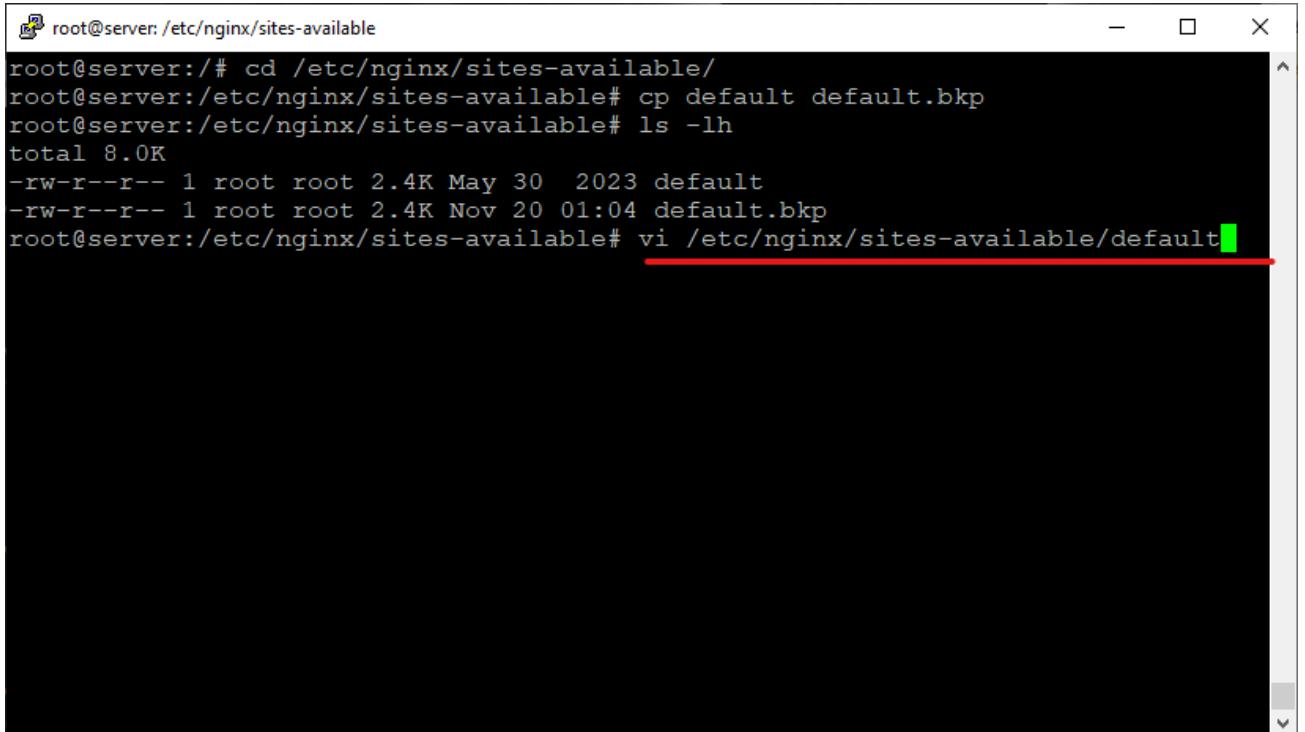
<https://learn.microsoft.com/en-us/aspnet/core/host-and-deploy/linux-nginx?view=aspnetcore-9.0&tabs=linux-ubuntu>

делаем копию файла



```
root@server:/etc/nginx/sites-available
root@server:/# cd /etc/nginx/sites-available/
root@server:/etc/nginx/sites-available# cp default default.bkp
root@server:/etc/nginx/sites-available# ls -lh
total 8.0K
-rw-r--r-- 1 root root 2.4K May 30 2023 default
-rw-r--r-- 1 root root 2.4K Nov 20 01:04 default.bkp
root@server:/etc/nginx/sites-available#
```

Редактируем файл:



```
root@server:/etc/nginx/sites-available
root@server:/# cd /etc/nginx/sites-available/
root@server:/etc/nginx/sites-available# cp default default.bkp
root@server:/etc/nginx/sites-available# ls -lh
total 8.0K
-rw-r--r-- 1 root root 2.4K May 30 2023 default
-rw-r--r-- 1 root root 2.4K Nov 20 01:04 default.bkp
root@server:/etc/nginx/sites-available# vi /etc/nginx/sites-available/default
```

Set number

Добавляем в 22-25 строки:

```
root@server: /etc/nginx/sites-available
14 # available underneath a path with that package name, such as /drupal8.
15 #
16 # Please see /usr/share/doc/nginx-doc/examples/ for more detailed examples.
17 ##
18
19 # Default server configuration
20 #
21
22 map $http_connection $connection_upgrade {
23     "~*Upgrade" $http_connection;
24     default keep-alive;
25 }
26
27 server {
28     listen 80 default_server;
29     listen [::]:80 default_server;
30
31     # SSL configuration
32     #
33     # listen 443 ssl default_server;
34     # listen [::]:443 ssl default_server;
35     #
36     # Note: You should disable gzip for SSL traffic.
-- INSERT --
```

31,1 17%

Комментируем 29 строку:

```
root@server: /etc/nginx/sites-available
12 # This file will automatically load configuration files provided by other
13 # applications, such as Drupal or Wordpress. These applications will be made
14 # available underneath a path with that package name, such as /drupal8.
15 #
16 # Please see /usr/share/doc/nginx-doc/examples/ for more detailed examples.
17 ##
18
19 # Default server configuration
20 #
21
22 map $http_connection $connection_upgrade {
23     "~*Upgrade" $http_connection;
24     default keep-alive;
25 }
26
27 server {
28     listen 80 default_server;
29     #listen [::]:80 default_server;
30
31     # SSL configuration
32     #
33     # listen 443 ssl default_server;
34     # listen [::]:443 ssl default_server;
-- INSERT --
```

29,10 14%

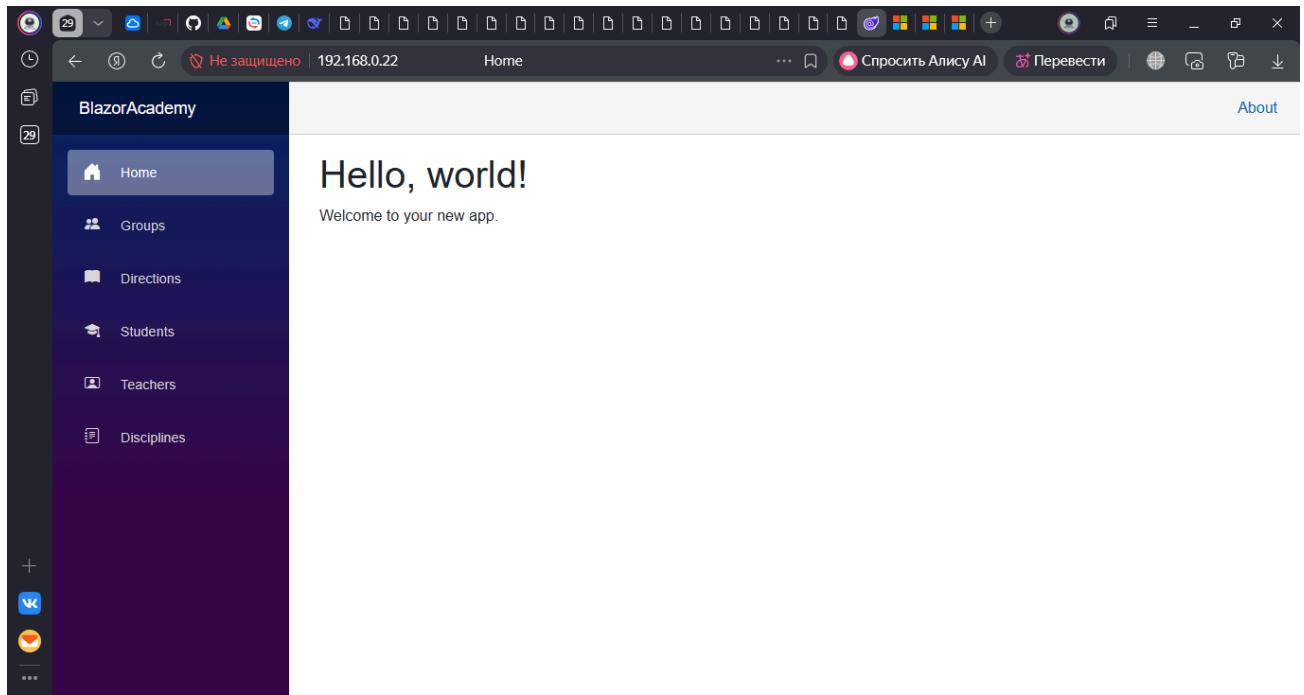
```
root@server: /etc/nginx/sites-available
50         index index.html index.htm index.nginx-debian.html;
51
52         server_name _;
53
54         location / {
55             # First attempt to serve request as file, then
56             # as directory, then fall back to displaying a 404.
57             #try_files $uri $uri/ =404;
58             proxy_pass          http://127.0.0.1:5234/;
59             proxy_http_version  1.1;
60             proxy_set_header    Upgrade $http_upgrade;
61             proxy_set_header    Connection $connection_upgrade;
62             proxy_set_header    Host $host;
63             proxy_cache_bypass $http_upgrade;
64             proxy_set_header    X-Forwarded-For $proxy_add_x_forwarded_for;
65             proxy_set_header    X-Forwarded-Proto $scheme;
66         }
67
68         # pass PHP scripts to FastCGI server
69         #
70         #location ~ \.php$ {
71         #     include snippets/fastcgi-php.conf;
72         #
-- INSERT --                                     65,29-41      59% ▼
```

Сохраняем.

Переходим в папку с проектом и перезагружаем nginx с помощью команды `service nginx restart`:

```
root@server: /var/www/asp/BlazorAcademy
root@server:/etc/nginx/sites-available# cd ../../..
root@server:# cd /var/www/asp/BlazorAcademy/
root@server:/var/www/asp/BlazorAcademy# service nginx restart
root@server:/var/www/asp/BlazorAcademy# netstat -an4
Active Internet connections (servers and established)
Proto Recv-Q Local Address          Foreign Address        State
tcp     0     0.0.0.0:5234           0.0.0.0:*            LISTEN
tcp     0     0.0.0.0:1433           0.0.0.0:*            LISTEN
tcp     0     0.127.0.0.1:1431       0.0.0.0:*            LISTEN
tcp     0     0.127.0.0.53:53        0.0.0.0:*            LISTEN
tcp     0     0.0.0.0:80             0.0.0.0:*            LISTEN
tcp     0     0.0.0.0:22             0.0.0.0:*            LISTEN
tcp     0     64.192.168.0.22:22     192.168.0.108:53991 ESTABLISHED
tcp     0     0.127.0.0.1:1433       127.0.0.1:38213      ESTABLISHED
tcp     0     0.192.168.0.22:22     192.168.0.108:53296 ESTABLISHED
tcp     0     0.127.0.0.1:38213       127.0.0.1:1433      ESTABLISHED
udp     0     0.127.0.0.53:53        0.0.0.0:*            LISTEN
udp     0     0.192.168.0.105:68      0.0.0.0:*            LISTEN
root@server:/var/www/asp/BlazorAcademy# █
```

Теперь по адресу <http://192.168.0.22/> все работает:



Автоматический запуск сервера при старте системы:

Проблема заключается в том, что при перезагрузке сервера нужно заново переходить в папку с проектом и запускать dotnet. Это очень неудобно. Исправим это. По сути, ниже мы просто создаем службу, которая запускает сервер при запуске системы.

<https://learn.microsoft.com/en-us/aspnet/core/host-and-deploy/linux-nginx?view=aspnetcore-9.0&tabs=linux-ubuntu#create-the-service-file>

```
root@server: /var/www/asp/BlazorAcademy
root@server:/var/www/asp/BlazorAcademy# vi /etc/systemd/system/kestrel-blazoracademy.service
```

```
root@server:/var/www/asp/BlazorAcademy
[Unit]
Description=BlazorAcademy PV_319

[Service]
WorkingDirectory=/var/www/asp/BlazorAcademy
ExecStart=/usr/bin/dotnet /var/www/asp/BlazorAcademy/bin/Debug/net8.0/BlazorAcademy.dll
Restart=always
RestartSec=10
KillSignal=SIGINT
SyslogIdentifier=BlazorAcademy
User=root
Environment=ASPNETCORE_ENVIRONMENT=Development
Environment=DOTNET_NOLOGO=true

[Install]
WantedBy=multi-user.target
~
~
~
~
~
~
~
-- INSERT --
```

16,27

All

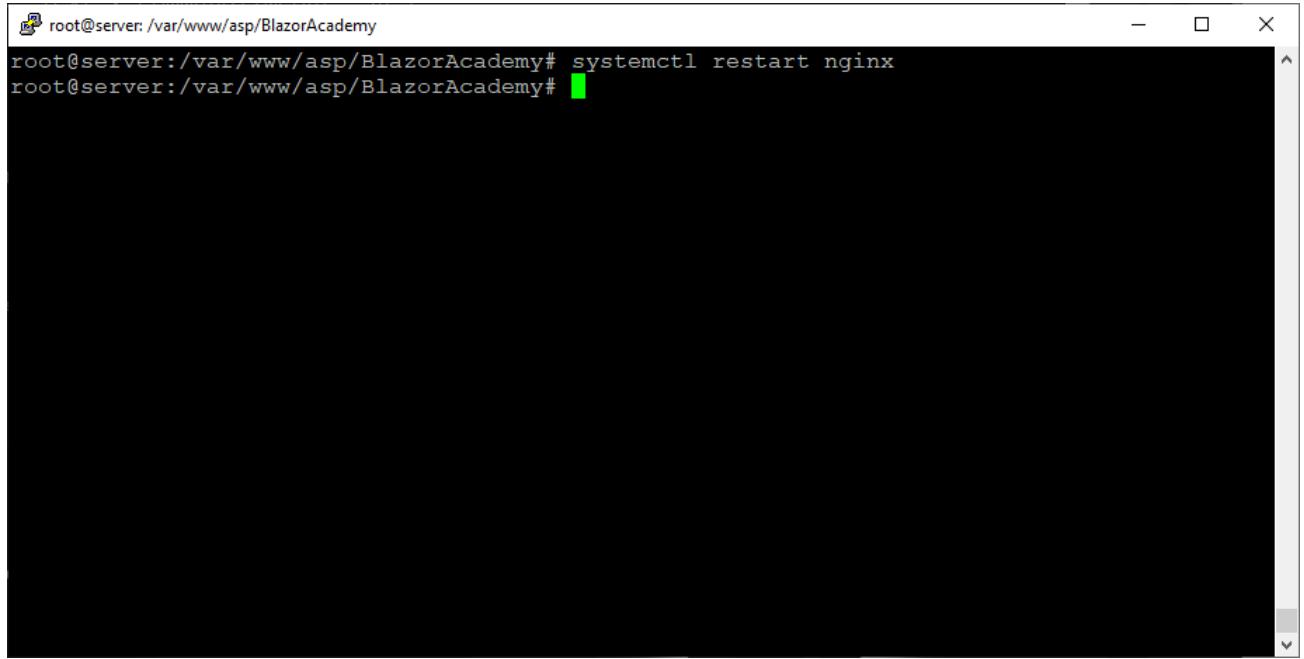
```
root@server:/var/www/asp/BlazorAcademy
Try: apt install <deb name>
root@server:/var/www/asp/BlazorAcademy# systemctl enable kestrel-blazoracademy.service
Created symlink /etc/systemd/system/multi-user.target.wants/kestrel-blazoracademy.service.
→ /etc/systemd/system/kestrel-blazoracademy.service.
root@server:/var/www/asp/BlazorAcademy# netstat -an4
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address          Foreign Address        State
tcp     0      0 0.0.0.0:5234            0.0.0.0:*              LISTEN
tcp     0      0 0.0.0.0:1433            0.0.0.0:*              LISTEN
tcp     0      0 127.0.0.1:1431           0.0.0.0:*              LISTEN
tcp     0      0 127.0.0.53:53           0.0.0.0:*              LISTEN
tcp     0      0 0.0.0.0:80              0.0.0.0:*              LISTEN
tcp     0      0 0.0.0.0:22              0.0.0.0:*              LISTEN
tcp     0      0 192.168.0.22:42094       13.107.246.53:443    TIME_WAIT
tcp     0      0 192.168.0.22:22           192.168.0.108:53991   ESTABLISHED
tcp     0      0 127.0.0.1:44363          127.0.0.1:1433         ESTABLISHED
tcp     0      0 192.168.0.22:22           192.168.0.108:53296   ESTABLISHED
tcp     0      0 192.168.0.22:49706       91.189.91.81:80        TIME_WAIT
tcp     0      0 192.168.0.22:60426       213.180.204.183:80     TIME_WAIT
tcp     0      0 127.0.0.1:1433          127.0.0.1:44363        ESTABLISHED
udp     0      0 127.0.0.53:53           0.0.0.0:*              ESTABLISHED
udp     0      0 192.168.0.105:68          0.0.0.0:*              ESTABLISHED
root@server:/var/www/asp/BlazorAcademy# systemctl start kestrel-blazoracademy.service
root@server:/var/www/asp/BlazorAcademy#
```

Переходим в файл и меняем порт на 5000:

```
root@server:/var/www/asp/BlazorAcademy
root@server:/var/www/asp/BlazorAcademy# vi /etc/nginx/sites-available/default
root@server:/var/www/asp/BlazorAcademy#
```

```
root@server:/var/www/asp/BlazorAcademy
root@server:/var/www/asp/BlazorAcademy# vi /etc/nginx/sites-available/default
location / {
    # First attempt to serve request as file, then
    # as directory, then fall back to displaying a 404.
    #try_files $uri $uri/ =404;
    proxy_pass http://127.0.0.1:5000/;
    proxy_http_version 1.1;
    proxy_set_header Upgrade $http_upgrade;
    proxy_set_header Connection $connection_upgrade;
    proxy_set_header Host $host;
    proxy_set_header $http_upgrade;
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    proxy_set_header X-Forwarded-Proto $scheme;
}

# pass PHP scripts to FastCGI server
#
#location ~ \.php$ {
#    include snippets/fastcgi-php.conf;
#
#    # With php-fpm (or other unix sockets):
#    fastcgi_pass unix:/run/php/php7.4-fpm.sock;
#    # With php-cgi (or other tcp sockets):
#}
"/etc/nginx/sites-available/default" 105L, 2913B written      58,35-61      63% ▼
```



```
root@server:/var/www/asp/BlazorAcademy
root@server:/var/www/asp/BlazorAcademy# systemctl restart nginx
root@server:/var/www/asp/BlazorAcademy#
```

Перезагружаем тачку и сразу проверяем без ввода dotnet run, что все работает!

