

Порядок установки (продублирован в комментариях в [Vagrantfile](#)):

1. Скачиваем и устанавливаем virtualbox: <https://www.virtualbox.org/wiki/Downloads>
2. Можно установить Oracle VM VirtualBox Extension Pack - плагин для virtualbox
- 2.1. Скачиваем плагин, чтобы все летало:
https://download.virtualbox.org/virtualbox/6.0.2/Oracle_VM_VirtualBox_Extension_Pack-6.0.2.vbox-extpack
- 2.2. Открываем virtualbox, нажимаем “Настройки”, затем “Плагины”, а там небольшую кнопку “+”
- 2.3. Выбираем только что скачанный файл
Oracle_VM_VirtualBox_Extension_Pack-6.0.2.vbox-extpack и подтверждаем установку плагина
3. Скачиваем и устанавливаем vagrant для вашей ОС: если работаете на компьютере с Windows - то скачиваете 32 бит или 64 бит версию, если на маке - нужна версия под macOS. (это утилита для консоли, которая будет производить CAMA по команде из п.6.2 подъем виртуальной машины с убунтой): <https://www.vagrantup.com/downloads.html>

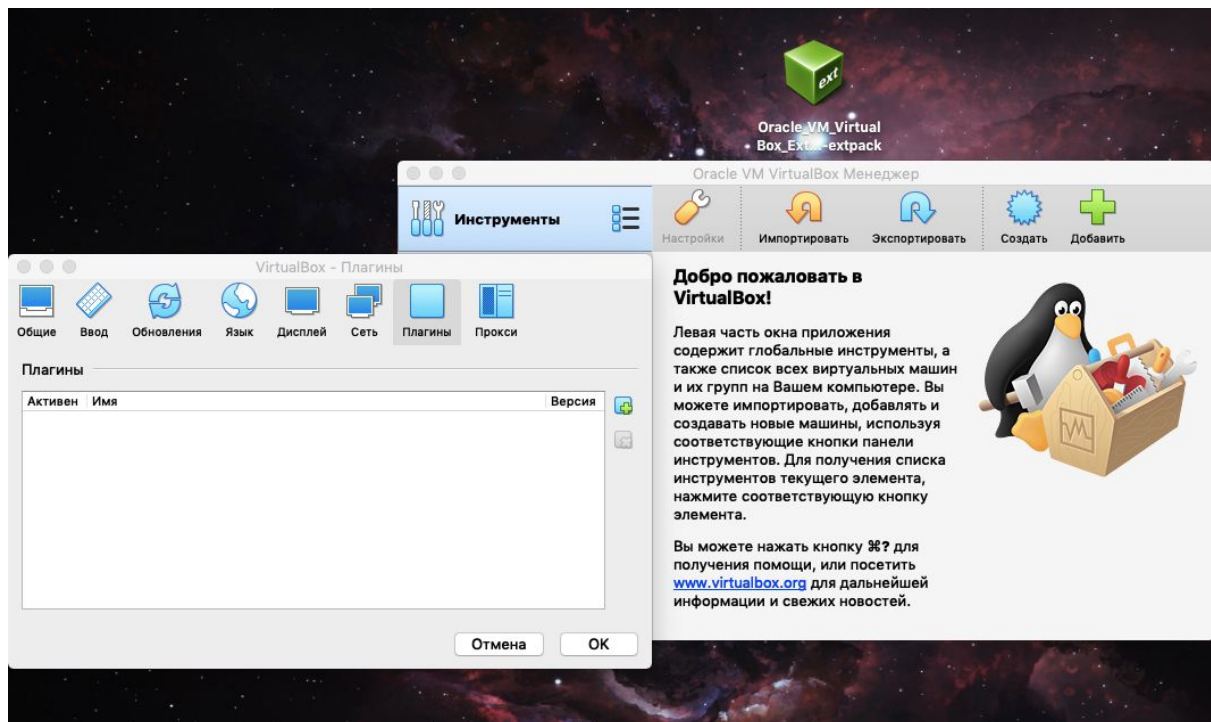
Версию вашей ОС Windows определяете так (на примере Win7):

Откройте компонент “Система”. Для этого нажмите кнопку Пуск Изображение кнопки , щелкните правой кнопкой мыши Компьютер и выберите пункт Свойства.
В разделе Система можно просмотреть тип системы.

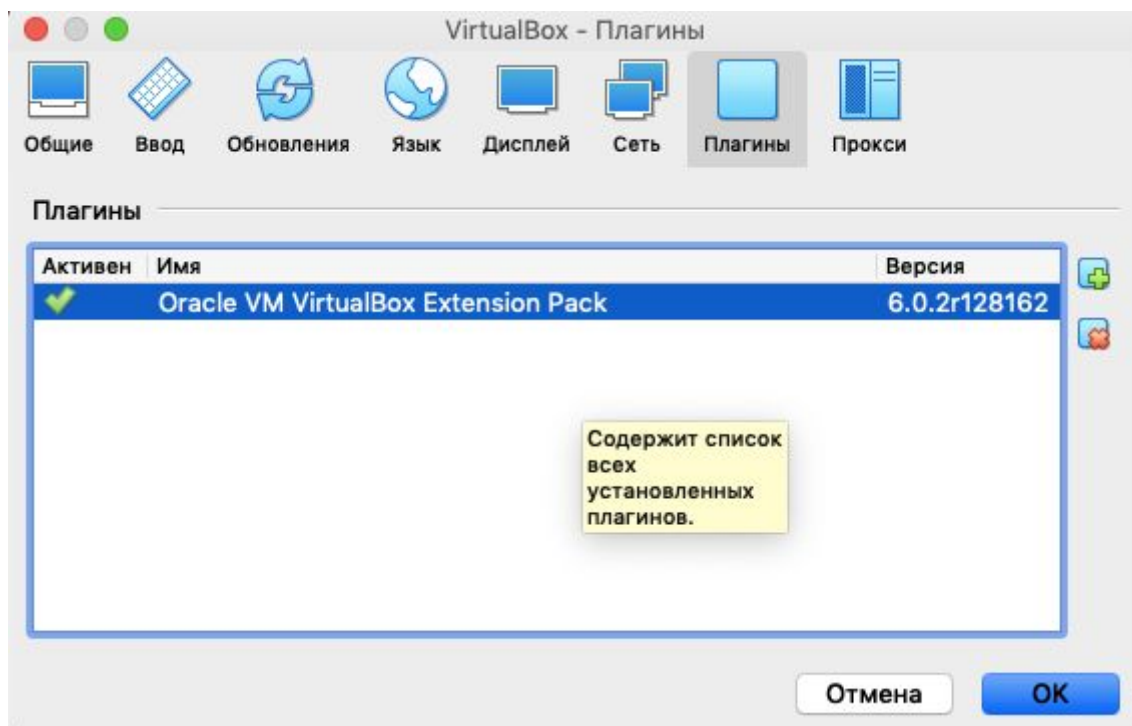
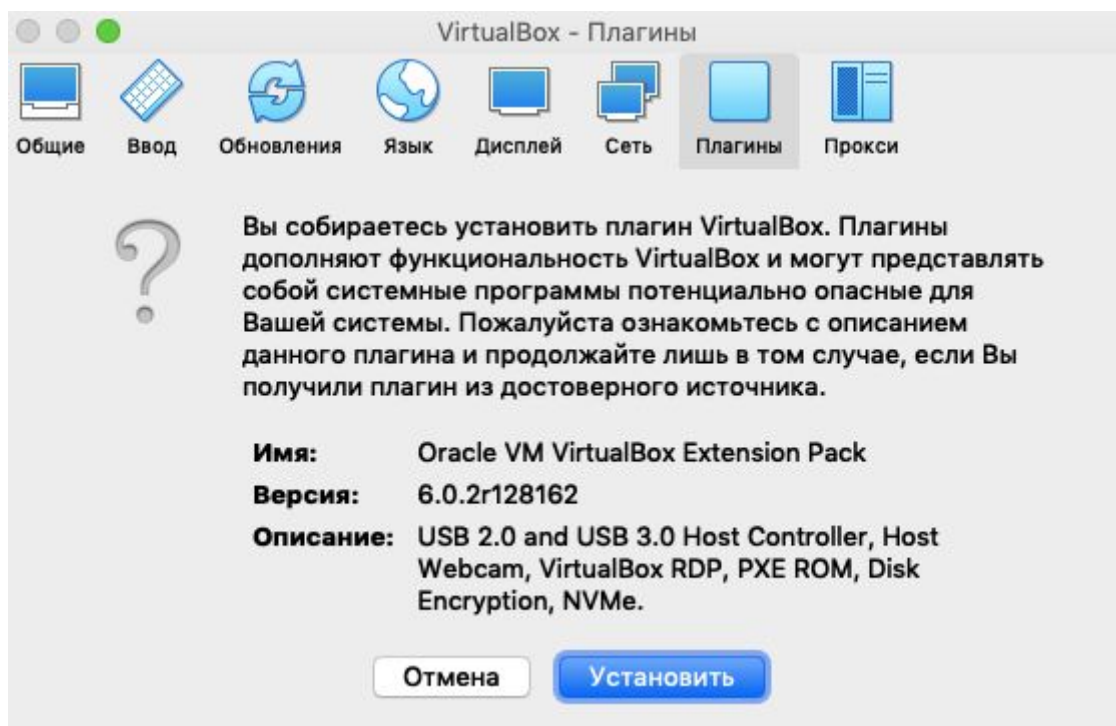
4. Создаем папку для нашего проекта в корне диска C, ручками. Например “C:\NETOLOGY_DATA”.
5. Копируем в нее файл “Vagrantfile”, ручками
6. В консоли рабочей машины (**cmd**), переходим в эту папку командой
cd C:\NETOLOGY_DATA
- 6.1. Вводим команду “vagrant up” и ждем завершения загрузки и установки (минут 10 максимум).
- 6.2. Вводим команду “vagrant ssh” и оказываемся в консоли linux
7. команды подключения к базам данных (**начните с первой, все разом запускать не нужно**):
 - 7.1. psql - стандартная консоль PostgreSQL (уже создана БД и суперпользователь vagrant - никаких параметров вводить не нужно)
 - 7.2. mongo - консоль mongodb (**понадобится позднее, сейчас не нужна**)

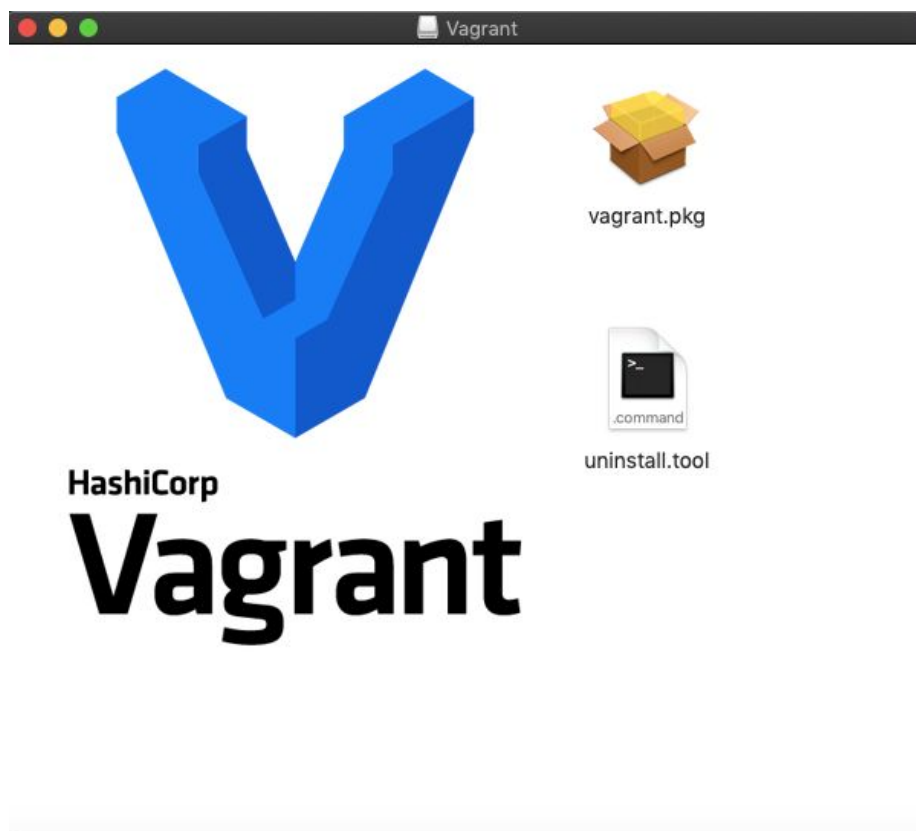
Скриншоты прохождения по шагам:

2.2

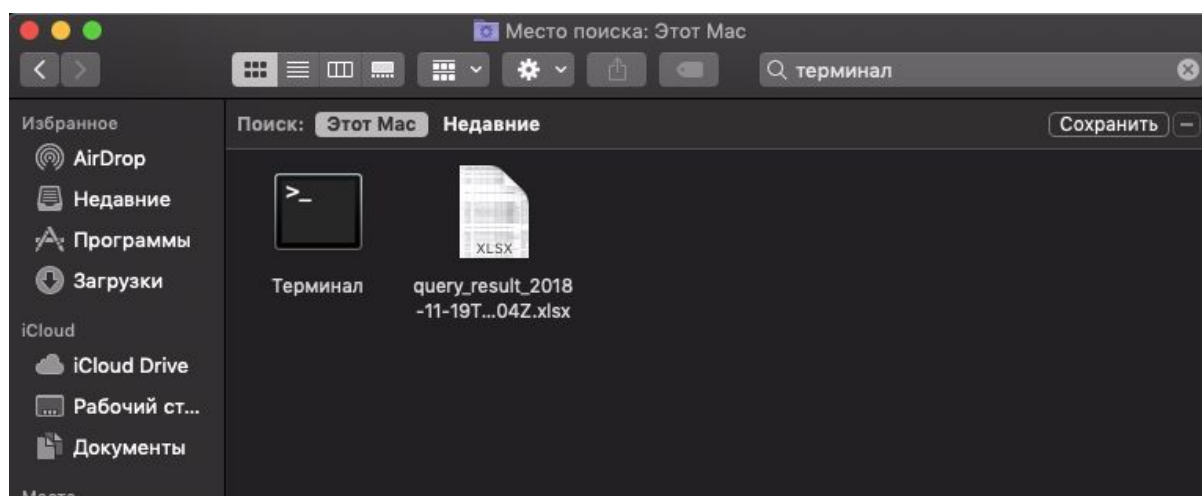


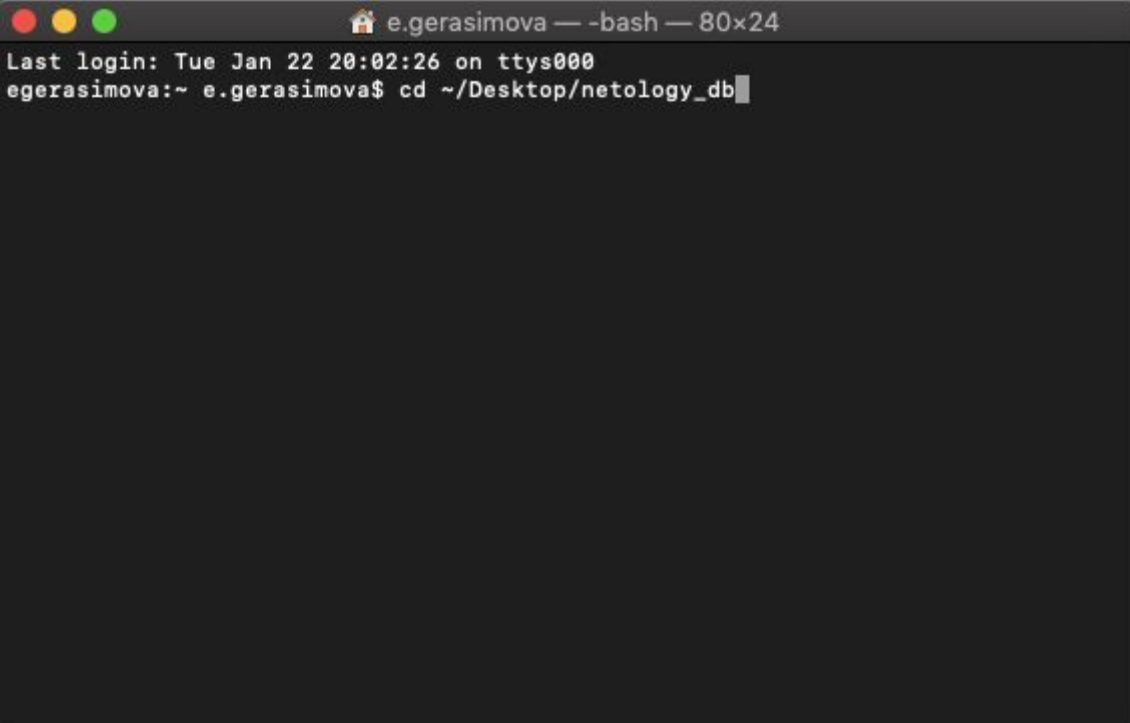
2.3





6





```
e.gerasimova — -bash — 80x24
Last login: Tue Jan 22 20:02:26 on ttys000
e.gerasimova:~ e.gerasimova$ cd ~/Desktop/netology_db
```

6.1

```
netology_db — curl - vagrant up — 80x24
Last login: Tue Jan 22 20:02:26 on ttys000
[egearasimova:~ e.gerasimova$ cd ~/Desktop/netology_db
[egearasimova:netology_db e.gerasimova$ vagrant up]
Bringing machine 'default' up with 'virtualbox' provider...
==> default: Box 'generic/ubuntu1804' could not be found. Attempting to find and
install...
    default: Box Provider: virtualbox
    default: Box Version: >= 0
==> default: Loading metadata for box 'generic/ubuntu1804'
    default: URL: https://vagrantcloud.com/generic/ubuntu1804
==> default: Adding box 'generic/ubuntu1804' (v1.8.56) for provider: virtualbox
    default: Downloading: https://vagrantcloud.com/generic/boxes/ubuntu1804/vers
ions/1.8.56/providers/virtualbox.box
    default: Download redirected to host: vagrantcloud-files-production.s3.amazo
naws.com
    default: Progress: 0% (Rate: 0*/s, Estimated time remaining: --:--:--)
```

Установлено!

```
netology_db — -bash — 80x24
e.gerasimova:netology_db e.gerasimova$ vagrant up
Bringing machine 'default' up with 'virtualbox' provider...
==> default: Box 'generic/ubuntu1804' could not be found. Attempting to find and
install...
    default: Box Provider: virtualbox
    default: Box Version: >= 0
==> default: Loading metadata for box 'generic/ubuntu1804'
    default: URL: https://vagrantcloud.com/generic/ubuntu1804
==> default: Adding box 'generic/ubuntu1804' (v1.8.58) for provider: virtualbox
    default: Downloading: https://vagrantcloud.com/generic/boxes/ubuntu1804/vers
ions/1.8.58/providers/virtualbox.box
    default: Download redirected to host: vagrantcloud-files-production.s3.amazo
naws.com
==> default: Successfully added box 'generic/ubuntu1804' (v1.8.58) for 'virtualb
ox'!
==> default: Importing base box 'generic/ubuntu1804'...
==> default: Matching MAC address for NAT networking...
==> default: Checking if box 'generic/ubuntu1804' version '1.8.58' is up to date
...
==> default: Setting the name of the VM: netology_db_default_1548228122274_50075
==> default: Clearing any previously set network interfaces...
==> default: Preparing network interfaces based on configuration...
    default: Adapter 1: nat
```



```
netology_db — -bash — 80x24
default: Setting up binutils-x86-64-linux-gnu (2.30-21ubuntu1~18.04) ...
default: Setting up cpp (4:7.3.0-3ubuntu2.1) ...
default: Setting up libpython3.6-dev:amd64 (3.6.7-1~18.04) ...
default: Setting up binutils (2.30-21ubuntu1~18.04) ...
default: Setting up python3.6-dev (3.6.7-1~18.04) ...
default: Setting up gcc-7 (7.3.0-27ubuntu1~18.04) ...
default: Setting up gcc (4:7.3.0-3ubuntu2.1) ...
default: Processing triggers for libc-bin (2.27-3ubuntu1) ...
default: CREATE DATABASE vagrant ENCODING 'UTF-8' TEMPLATE template0;
default: CREATE ROLE vagrant WITH LOGIN PASSWORD 'vagrant';
default: GRANT ALL PRIVILEGES ON DATABASE vagrant TO vagrant;
default: ALTER USER vagrant WITH SUPERUSER;
default: ALTER USER vagrant WITH CREATEROLE;
default: ALTER USER vagrant WITH CREATEDB;
default: ALTER USER vagrant WITH REPLICATION;
default: CREATE DATABASE
default: CREATE ROLE
default: GRANT
default: ALTER ROLE
default: ALTER ROLE
default: ALTER ROLE
default: ALTER ROLE
e.gerasimova:netology_db e.gerasimova$
e.gerasimova:netology_db e.gerasimova$
```

6.2

```
netology_db — vagrant@ubuntu1804: ~ — ssh - vagrant ssh — 80x24
default: Setting up cpp (4:7.3.0-3ubuntu2.1) ...
default: Setting up libpython3.6-dev:amd64 (3.6.7-1~18.04) ...
default: Setting up binutils (2.30-21ubuntu1~18.04) ...
default: Setting up python3.6-dev (3.6.7-1~18.04) ...
default: Setting up gcc-7 (7.3.0-27ubuntu1~18.04) ...
default: Setting up gcc (4:7.3.0-3ubuntu2.1) ...
default: Processing triggers for libc-bin (2.27-3ubuntu1) ...
default: CREATE DATABASE vagrant ENCODING 'UTF-8' TEMPLATE template0;
default: CREATE ROLE vagrant WITH LOGIN PASSWORD 'vagrant';
default: GRANT ALL PRIVILEGES ON DATABASE vagrant TO vagrant;
default: ALTER USER vagrant WITH SUPERUSER;
default: ALTER USER vagrant WITH CREATEROLE;
default: ALTER USER vagrant WITH CREATEDB;
default: ALTER USER vagrant WITH REPLICATION;
default: CREATE DATABASE
default: CREATE ROLE
default: GRANT
default: ALTER ROLE
default: ALTER ROLE
default: ALTER ROLE
default: ALTER ROLE
e.gerasimova:netology_db e.gerasimova$
[e.gerasimova:netology_db e.gerasimova$ vagrant ssh
vagrant@ubuntu1804:~$
```

На windows разных версий могут возникнуть ошибки с виртуальной машиной и не только. Они разобраны ниже:

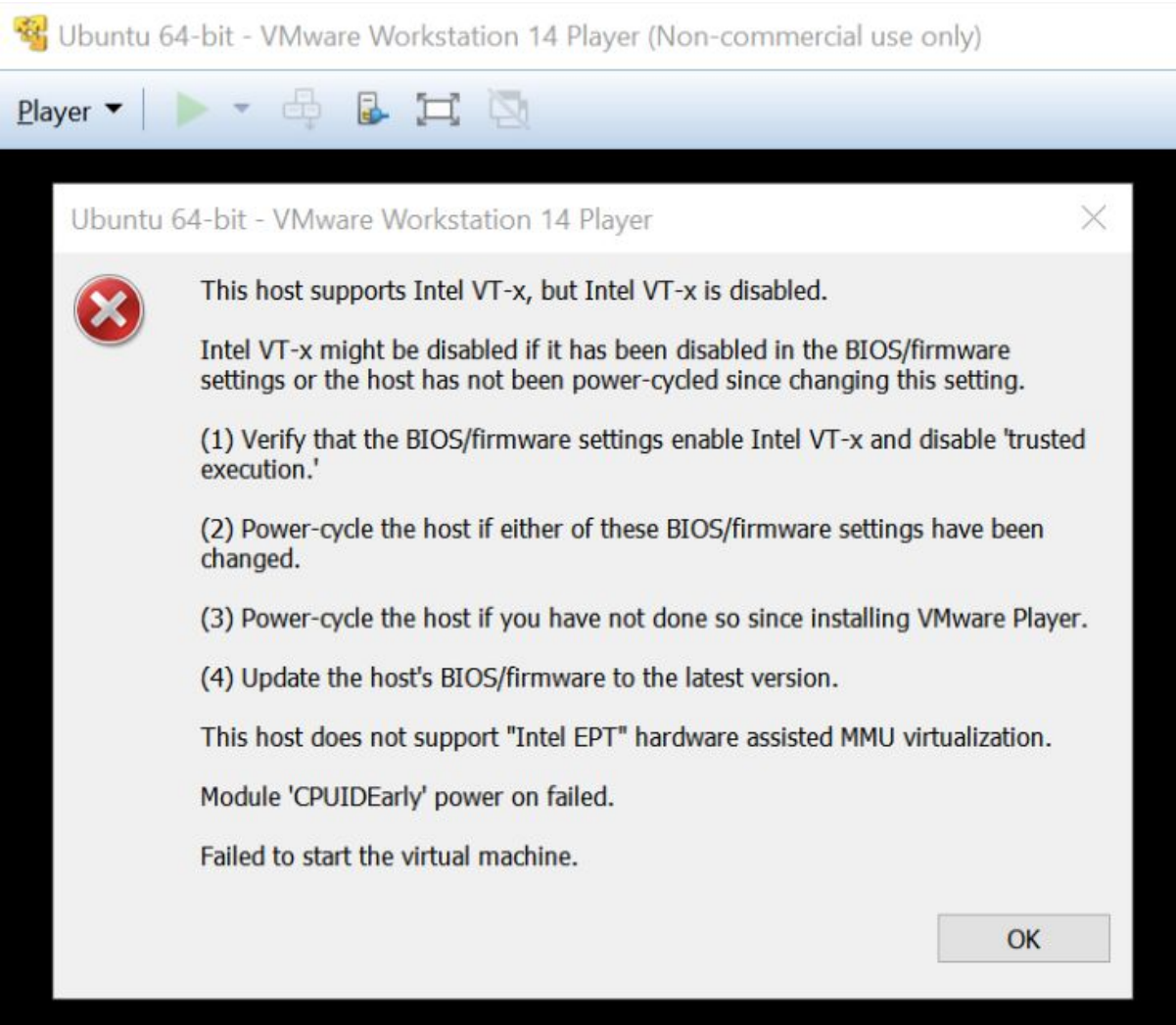
BIOS:

```
C:\Users\anton\Documents\NETOLOGY_DATA>vagrant up
Bringing machine 'default' up with 'virtualbox' provider...
==> default: Box 'generic/ubuntu1804' could not be found. Attempting to find and install...
    default: Box Provider: virtualbox
    default: Box Version: >= 0
==> default: Loading metadata for box 'generic/ubuntu1804'
    default: URL: https://vagrantcloud.com/generic/ubuntu1804
==> default: Adding box 'generic/ubuntu1804' (v1.8.58) for provider: virtualbox
    default: Downloading: https://vagrantcloud.com/generic/boxes/ubuntu1804/versions/1.8.58/providers/virtualbox.box
==> default: Box download is resuming from prior download progress
    default: Download redirected to host: vagrantcloud-files-production.s3.amazonaws.com
    default:
==> default: Successfully added box 'generic/ubuntu1804' (v1.8.58) for 'virtualbox'!
==> default: Importing base box 'generic/ubuntu1804'...
==> default: Matching MAC address for NAT networking...
==> default: Checking if box 'generic/ubuntu1804' version '1.8.58' is up to date...
==> default: Setting the name of the VM: NETOLOGY_DATA_default_1548314876404_59298
==> default: Clearing any previously set network interfaces...
==> default: Preparing network interfaces based on configuration...
    default: Adapter 1: nat
==> default: Forwarding ports...
    default: 22 (guest) => 2222 (host) (adapter 1)
==> default: Running 'pre-boot' VM customizations...
==> default: Booting VM...
There was an error while executing 'VBoxManage', a CLI used by Vagrant
for controlling VirtualBox. The command and stderr is shown below.

Command: ["startvm", "84acb74c-5e2f-4080-a9a5-4e0faee34d02", "--type", "headless"]

Stderr: VBoxManage.exe: error: Not in a hypervisor partition (HVP=0) (VERR_NEM_NOT_AVAILABLE).
VBoxManage.exe: error: VT-x is disabled in the BIOS for all CPU modes (VERR_VMX_MSR_ALL_VMX_DISABLED)
VBoxManage.exe: error: Details: code E_FAIL (0x80004005), component ConsoleWrap, interface IConsole

C:\Users\anton\Documents\NETOLOGY_DATA>
```

Если коротко, то:

В главном меню компьютера reboot (или перезагрузка), как только начнется перезагрузка - нажать F12 или Delete или F2 (на разных версиях по-разному) -> откроется синий экран. И в нем обязательно есть менюшка Virtualization.

Убедиться, что стоит **Enabled**.



<https://ourcodeworld.com/articles/read/708/how-to-solve-virtualbox-exception-vt-x-is-disabled-in-the-bios-for-both-all-cpu-modes-verr-vmx-msr-all-vmx-disabled>

Powershell:

Ошибка с версией:

```

C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
(c) icrosoft Corporation. All rights reserved.
C:\Users\User>cd /d E:\netology_db
E:\netology_db>vagrant up
The version of powershell currently installed on this host is less than
the required minimum version. Please upgrade the installed version of
powershell to the minimum required version and run the command again.

    Installed version: 2
    Minimum required version: 3
E:\netology_db>

```

1/ через сам powershell исполнить команду:

1/ From the Start Menu

Click Start, type PowerShell, and then click Windows PowerShell.

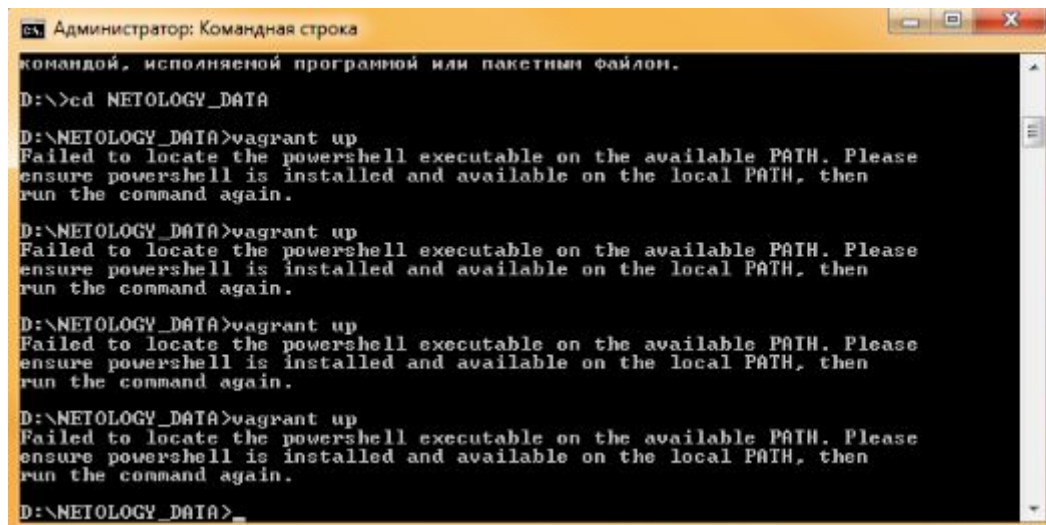
2/ From the Start menu, click Start, click All Programs, click Accessories, click the Windows PowerShell folder, and then click Windows PowerShell.

2/ обновить powershell: сразу до 3 версии

<https://www.microsoft.com/en-us/download/details.aspx?id=34595>

3/ обновить версию windows - это если есть время

Ошибка с наличием:



```
Администратор: Командная строка
командой, исполняемой программой или пакетным файлом.
D:\>cd NETOLOGY_DATA
D:\NETOLOGY_DATA>vagrant up
Failed to locate the powershell executable on the available PATH. Please
ensure powershell is installed and available on the local PATH, then
run the command again.
D:\NETOLOGY_DATA>vagrant up
Failed to locate the powershell executable on the available PATH. Please
ensure powershell is installed and available on the local PATH, then
run the command again.
D:\NETOLOGY_DATA>vagrant up
Failed to locate the powershell executable on the available PATH. Please
ensure powershell is installed and available on the local PATH, then
run the command again.
D:\NETOLOGY_DATA>vagrant up
Failed to locate the powershell executable on the available PATH. Please
ensure powershell is installed and available on the local PATH, then
run the command again.
D:\NETOLOGY_DATA>
```

Попробуйте скачать отсюда версию 3:

<https://www.microsoft.com/en-us/download/details.aspx?id=34595>

Vagrant:

Ошибка при выполнении “vagrant up”:

```
.../HashiCorp/Vagrant/embedded/gems/2.2.3/gems/vagrant-2.2.3/lib/vag
rant/action/warden.rb:34:in `call'
  from C:/HashiCorp/Vagrant/embedded/gems/2.2.3/gems/vagrant-2.2.3/plugins
/providers/virtualbox/action/check_virtualbox.rb:26:in `call'
  from C:/HashiCorp/Vagrant/embedded/gems/2.2.3/gems/vagrant-2.2.3/lib/vag
rant/action/warden.rb:34:in `call'
  from C:/HashiCorp/Vagrant/embedded/gems/2.2.3/gems/vagrant-2.2.3/lib/vag
rant/action/builder.rb:116:in `call'
  from C:/HashiCorp/Vagrant/embedded/gems/2.2.3/gems/vagrant-2.2.3/lib/vag
rant/action/runner.rb:66:in `block in run'
  from C:/HashiCorp/Vagrant/embedded/gems/2.2.3/gems/vagrant-2.2.3/lib/vag
rant/util/busy.rb:19:in `busy'
  from C:/HashiCorp/Vagrant/embedded/gems/2.2.3/gems/vagrant-2.2.3/lib/vag
rant/action/runner.rb:66:in `run'
  from C:/HashiCorp/Vagrant/embedded/gems/2.2.3/gems/vagrant-2.2.3/lib/vag
rant/machine.rb:239:in `action_raw'
  from C:/HashiCorp/Vagrant/embedded/gems/2.2.3/gems/vagrant-2.2.3/lib/vag
rant/machine.rb:208:in `block in action'
  from C:/HashiCorp/Vagrant/embedded/gems/2.2.3/gems/vagrant-2.2.3/lib/vag
rant/environment.rb:614:in `lock'
  from C:/HashiCorp/Vagrant/embedded/gems/2.2.3/gems/vagrant-2.2.3/lib/vag
rant/machine.rb:194:in `call'
  from C:/HashiCorp/Vagrant/embedded/gems/2.2.3/gems/vagrant-2.2.3/lib/vag
rant/machine.rb:194:in `action'
  from C:/HashiCorp/Vagrant/embedded/gems/2.2.3/gems/vagrant-2.2.3/lib/vag
rant/batch_action.rb:82:in `block (2 levels) in run'

C:\NETOLOGY_DATA>vagrant ssh
VM must be created before running this command. Run `vagrant up` first.
```

Убедитесь, что в настройках Virtualbox путь к виртуальным машинам по-умолчанию, не содержит букв кириллицы:

Открываем Virtualbox, настройки. В разделе “Общие” видим параметр “Папка для машин по умолчанию”. Если значение содержит русские символы, его нужно поправить. Например:

было: C:\Users\Павел\VirtualBox VMs

надо: C:\VirtualBox VMs

Может потребоваться установить переменную окружения:

VAGRANT_HOME = 'c:\HashiCorp'

Если все еще не помогло (сразу лучше так не делать - только по необходимости):

Найдите файл c:\HashiCorp\Vagrant\embedded\gems\gems\vagrant-1.6.3\bin\vagrant

Откройте его в текстовом редакторе (не Word!) и после строчки

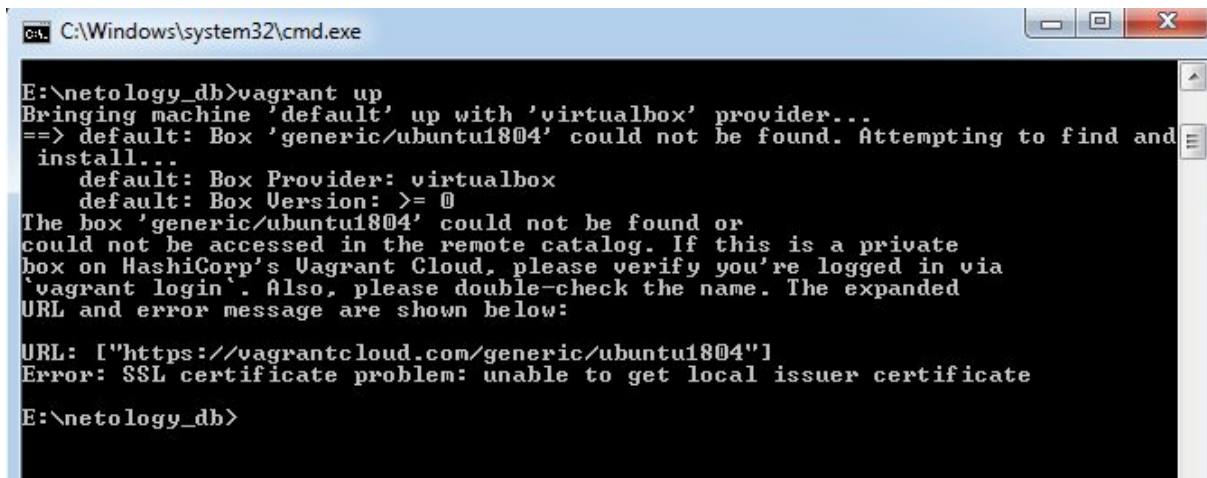
```
#!/usr/bin/env ruby
```

добавьте эти две строки:

```
Encoding.default_external = Encoding.find('Windows-1251')
```

```
Encoding.default_internal = Encoding.find('Windows-1251')
```

SSL/TLS:

A screenshot of a Windows command prompt window titled "C:\Windows\system32\cmd.exe". The prompt is at "E:\netology_db>". The user has entered "vagrant up". The output shows Vagrant attempting to bring up a machine named 'default' using the 'virtualbox' provider. It reports that the box 'generic/ubuntu1804' could not be found and attempts to find and install it. It shows the provider is 'virtualbox' and the version is '0'. Then it states the box could not be found or accessed in the remote catalog. It provides a URL: ["https://vagrantcloud.com/generic/ubuntu1804"] and an error: "Error: SSL certificate problem: unable to get local issuer certificate". The prompt returns to "E:\netology_db>".

```
E:\netology_db>vagrant up
Bringing machine 'default' up with 'virtualbox' provider...
==> default: Box 'generic/ubuntu1804' could not be found. Attempting to find and
install...
    default: Box Provider: virtualbox
    default: Box Version: >= 0
The box 'generic/ubuntu1804' could not be found or
could not be accessed in the remote catalog. If this is a private
box on HashiCorp's Vagrant Cloud, please verify you're logged in via
'vagrant login'. Also, please double-check the name. The expanded
URL and error message are shown below:

URL: ["https://vagrantcloud.com/generic/ubuntu1804"]
Error: SSL certificate problem: unable to get local issuer certificate

E:\netology_db>
```

Отключить фильтрацию протокола SSL/TLS в работающем антивирусе

Пароль консоли ubuntu, если вдруг где-то попросит:

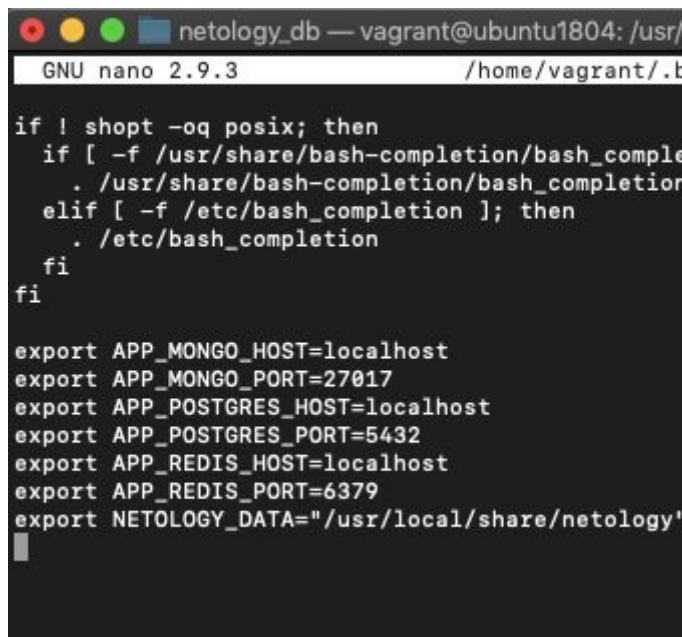
vagrant / vagrant

=====

Если все ок, то просто смотрим дальше.

В процессе установки через vagrant, в файл ~/.bashrc автоматически добавятся такие строки:

```
export APP_MONGO_HOST=localhost
export APP_MONGO_PORT=27017
export APP_POSTGRES_HOST=localhost
export APP_POSTGRES_PORT=5432
export APP_REDIS_HOST=localhost
export APP_REDIS_PORT=6379
export NETOLOGY_DATA="/usr/local/share/netology"
```

```
netology_db — vagrant@ubuntu1804: /usr/
GNU nano 2.9.3 /home/vagrant/.b

if ! shopt -oq posix; then
  if [ -f /usr/share/bash-completion/bash_comple
    . /usr/share/bash-completion/bash_completion
  elif [ -f /etc/bash_completion ]; then
    . /etc/bash_completion
  fi
fi

export APP_MONGO_HOST=localhost
export APP_MONGO_PORT=27017
export APP_POSTGRES_HOST=localhost
export APP_POSTGRES_PORT=5432
export APP_REDIS_HOST=localhost
export APP_REDIS_PORT=6379
export NETOLOGY_DATA="/usr/local/share/netology"
```

Чтобы проверить, что они применились, выполним в консоли команду `echo $NETOLOGY_DATA` - должны увидеть в результат `/usr/local/share/netology_data`.

Также, автоматически создались следующие папки:

```
$NETOLOGY_DATA;
$NETOLOGY_DATA/raw_data
$NETOLOGY_DATA/pg_data
$NETOLOGY_DATA/data
```

Перейдём в созданную директорию:

```
cd $NETOLOGY_DATA
```

Загрузка дампа БД и csv данных

Теперь данные, которые я заранее залил на Google Drive нужно перенести на локальную машину. Для этого склонируем полезный репозиторий (содержит утилиту для скачивания с Google Cloud).

Мы используем данные The Movies Dataset с Kaggle.

Исполняем в консоли ubuntu:

```
rm -rf download_google_drive
```

```
git clone https://github.com/chentinghao/download_google_drive.git
```

```
netology_db — vagrant@ubuntu1804: /usr/local/share/netology — ssh - vagran...
[vagrant@ubuntu1804: /usr/local/share/netology$ python download_google_drive/downl
oad_gdrive.py 1D3CcWOSw-MUx6YvJ_4dqOLHZA h-6uTxK data.zip
23.6MB [00:04, 5.28MB/s]
[vagrant@ubuntu1804: /usr/local/share/netology$ unzip data.zip -d "$NETOLOGY_DATA"
/raw_data;
Archive: data.zip
  inflating: /usr/local/share/netology/raw_data/credits.csv
  inflating: /usr/local/share/netology/raw_data/keywords1.csv
  inflating: /usr/local/share/netology/raw_data/keywords.csv
  inflating: /usr/local/share/netology/raw_data/links.csv
  inflating: /usr/local/share/netology/raw_data/links_small.csv
  inflating: /usr/local/share/netology/raw_data/movies_metadata.csv
  inflating: /usr/local/share/netology/raw_data/ratings.csv
  inflating: /usr/local/share/netology/raw_data/ratings_small.csv
  inflating: /usr/local/share/netology/raw_data/simple_tags.json
  inflating: /usr/local/share/netology/raw_data/test.csv
  inflating: /usr/local/share/netology/raw_data/test.json
[vagrant@ubuntu1804: /usr/local/share/netology$ git clone https://github.com/Dju99
9/data_analytics.git
Cloning into 'data_analytics'...
remote: Enumerating objects: 8, done.
remote: Counting objects: 100% (8/8), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 212 (delta 3), reused 8 (delta 3), pack-reused 204
```

Загружаем дамп БД, чтобы создать нужные для работы таблицы

**Далее, если команды python3.... выдают ошибку, попробуйте писать python ... ;
[уберите цифру 3]**

команда длинная - копируйте обе строки вместе:

```
python3 download_google_drive/download_gdrive.py
1uWZjmm9vw xZMpIMqUtnOr-M16a0ochQa $NETOLOGY_DATA/data/all_tables.dump
```

Теперь скачаем текстовые данные в формате csv, которые могут пригодиться для загрузки в Python в следующих частях курса. **Запускаем скачивание файла - zip архива с данными. Архив весит примерно 23Mb**

команда длинная - копируйте обе строки вместе:

```
python3 download_google_drive/download_gdrive.py
1D3CcWOSw-MUx6YvJ_4dqOLHZA h-6uTxK data.zip
```

Сначала распаковываем архив с данными.

```
unzip data.zip -d "$NETOLOGY_DATA"/raw_data;
```

Мы увидим процесс извлечения данных - это csv и json файлы. **Выглядит это так:**

```
Archive: data.zip
  inflating: /tmp/data/ratings.csv
```

```
inflating: /tmp/data/ratings_small.csv
inflating: /tmp/data/links.csv
inflating: /tmp/data/links_small.csv
inflating: /tmp/data/keywords.csv
inflating: /tmp/data/movies_metadata.csv
inflating: /tmp/data/credits.csv
```

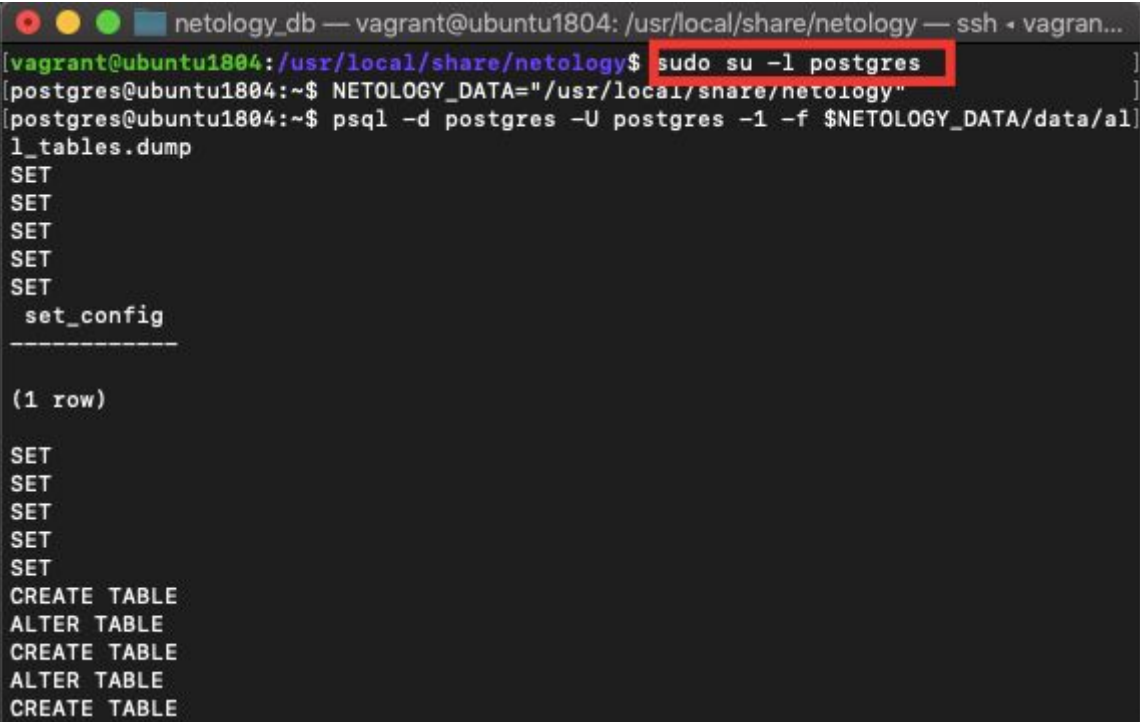
Теперь скачиваем репозиторий курса - там хранятся материалы для домашних работ.

```
git clone https://github.com/Dju999/data_analytics.git
```

Теперь нужно загрузить дампы в Postgres. Для этого перейдем в консоль Postgres

Исполнить команду:

```
sudo su -l postgres
```

A terminal window titled 'netology_db — vagrant@ubuntu1804: /usr/local/share/netology — ssh • vagran...' shows the following commands and output:

```
[vagrant@ubuntu1804:~$ sudo su -l postgres]
[postgres@ubuntu1804:~$ NETOLOGY_DATA="/usr/local/share/netology"
[postgres@ubuntu1804:~$ psql -d postgres -U postgres -1 -f $NETOLOGY_DATA/data/all_tables.dump
SET
SET
SET
SET
SET
set_config
-----
(1 row)

SET
SET
SET
SET
SET
SET
CREATE TABLE
ALTER TABLE
CREATE TABLE
ALTER TABLE
CREATE TABLE
```

Мы переключимся в сеанс пользователя Postgres. Экспортируем переменную среды для доступа к данным

```
NETOLOGY_DATA="/usr/local/share/netology"
```

Осталось загрузить текстовый дамп в PostgreSQL

```
psql -d postgres -U postgres -1 -f $NETOLOGY_DATA/data/all_tables.dump
```

Подключимся к БД и проверим, что данные загружены

```
psql -U postgres -c "SELECT COUNT(*) FROM ratings;"
```

Результат:

```
count
-----
777776
(1 row)
```

```
SET
SET
SET
SET
SET
CREATE TABLE
ALTER TABLE
CREATE TABLE
ALTER TABLE
CREATE TABLE
ALTER TABLE
COPY 46419
COPY 45843
COPY 777776
[postgres@ubuntu1804:~$ psql -U postgres -c "SELECT COUNT(*) FROM ratings;" ]
count
-----
777776
(1 row)
```

Если все именно так, то ты молодец и мой герой <3