

# Работа с Oracle VM VirtualBox и системой контроля версий git

---

Соболев М. С.

08 снетября 2022

Российский университет дружбы народов, Москва, Россия

# Информация

---

- Соболев Максим Сергеевич
- Студент 4 курса, 1032192035
- Направление: Бизнес-информатика
- Российский университет дружбы народов
- sobolek322lorek@gmail.com

# **Вводная часть**

---

- Владение git/Markdown/Oracle VM VirtualBox и Linux жизненно необходимо для дальнейшего выполнения лабораторных работ.

- Объектом исследования является: git, Linux, Oracle VM VirtualBox, Markdown
- Программное обеспечение для создания презентаций
- Входные и выходные форматы презентаций

- Создать презентацию в Markdown
- Пройдемся по самым основным пунктам лабораторной работы без лишней воды

- Отчет по ранее выполненной работе



## **Выполнение работы**

---

# Создание виртуальной машины

Указываем название, директорию установки и тип гостевой ОС.

← Create Virtual Machine

Name and operating system

Name: 1032192035

Machine Folder: Z:\VM

Type: Microsoft Windows

Version: Windows 7 (64-bit)

Memory size

4 MB 65536 MB 2048 MB

Hard disk

☐ Do not add a virtual hard disk

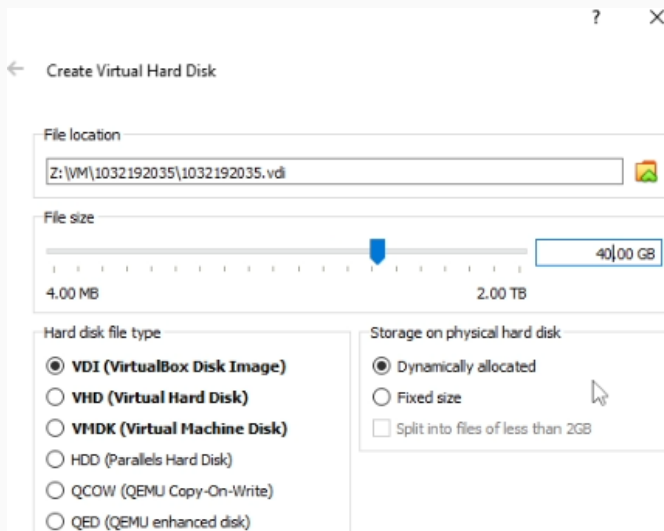
☒ Create a virtual hard disk now

☐ Use an existing virtual hard disk file

Empty

## Настраиваем диск

Указываем объем виртуального жесткого диска, место его хранения.



← Create Virtual Hard Disk

File location  
Z:\VM\1032192035\1032192035.vdi

File size  
4.00 MB 2.00 TB 40.00 GB

Hard disk file type

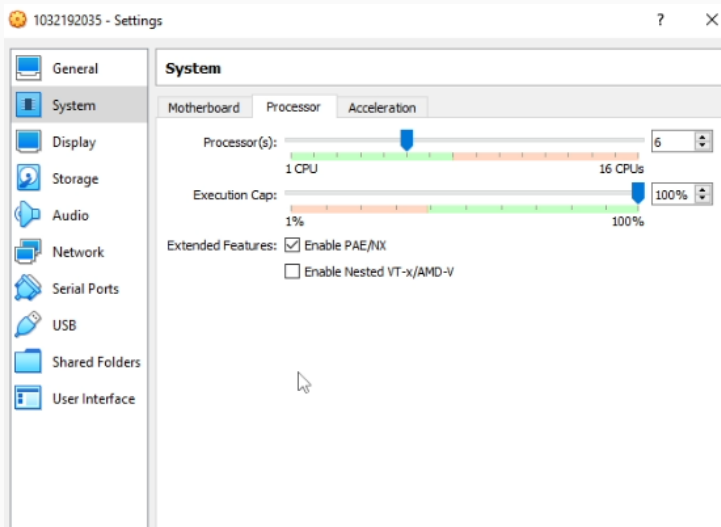
- ☒ VDI (VirtualBox Disk Image)
- ☐ VHD (Virtual Hard Disk)
- ☐ VMDK (Virtual Machine Disk)
- ☐ HDD (Parallels Hard Disk)
- ☐ QCOW (QEMU Copy-On-Write)
- ☐ QED (QEMU enhanced disk)

Storage on physical hard disk

- ☒ Dynamically allocated
- ☐ Fixed size
- ☐ Split into files of less than 2GB

# Настройка виртуального ЦП

Добавляем виртуальные ядра.



## Нажимаем кнопку “Start”

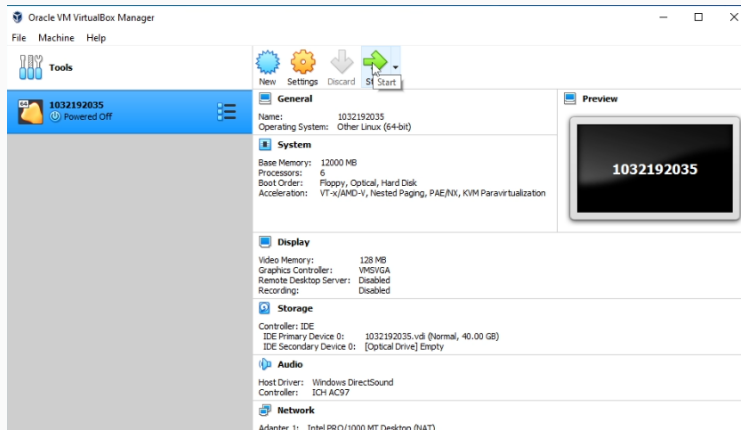
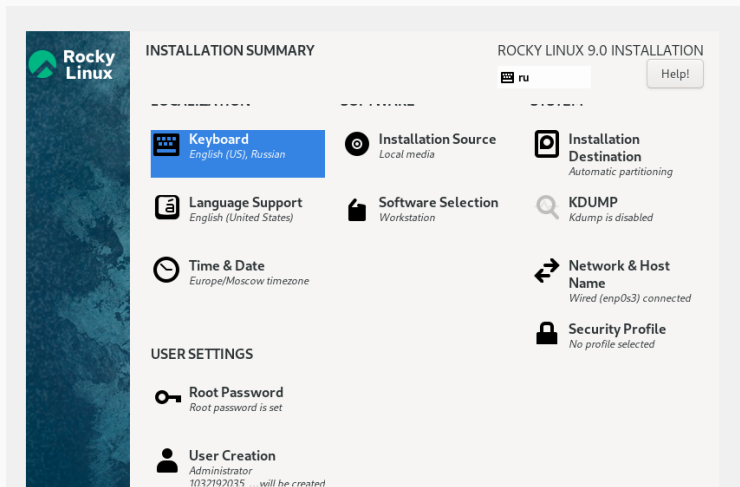


Figure 4: 4

# Окно с общими сведениями о параметрах будущей установки

Перед установкой мы можем увидеть общие сведения о параметрах будущей установки и настроить их. После настройки наблюдаем за установкой. Конец.



**Мы получили - установленную  
ОС**

---

# Работа с GitHub

---



gh – это утилита командной строки для управления репозиториями на GitHub.

```
[1032192035_pfur.ru@1032192035 ~]$ dnf install 'dnf-command(config-manager)'
Error: This command has to be run with superuser privileges (under the root user on most systems).
[1032192035_pfur.ru@1032192035 ~]$ sudo !!
sudo dnf install 'dnf-command(config-manager)'
[sudo] password for 1032192035_pfur.ru:
Last metadata expiration check: 0:02:42 ago on Tue 06 Sep 2022 07:07:33 PM MSK.
Package dnf-plugins-core-4.0.24-4.el9_0.noarch is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[1032192035_pfur.ru@1032192035 ~]$ sudo su
[root@1032192035 1032192035_pfur.ru]# dnf config-manager --add-repo https://cli.github.com/packages/rpm/gh-cli.repo
Adding repo from: https://cli.github.com/packages/rpm/gh-cli.repo
[root@1032192035 1032192035_pfur.ru]# dnf install gh
```

Figure 6: 6

Вводим свои данные. Настраиваем в соответствии с заданием.

```
[root@1032192035 gitflow]# git config --global user.name "Maksim Sobolev"  
[root@1032192035 gitflow]# git config --global user.email "sobolek322lorek@gmail.com"  
[root@1032192035 gitflow]# git config --global core.quotepath false  
[root@1032192035 gitflow]# git config --global init.defaultBranch master  
[root@1032192035 gitflow]# git config --global core.autocrlf input  
[root@1032192035 gitflow]# git config --global core.safecrlf warn
```

Figure 7: 7

# Настройка автоматических подписей коммитов git

Логинимся в GitHub с использованием утилиты командной строки gh.

```
[root@1032192035 gitflow]# git config --global user.signingKey 964FDB6455A76ABD  
[root@1032192035 gitflow]# git config --global commit.gpgSign true  
[root@1032192035 gitflow]# git config --global gpg.program $(which gpg)
```

Figure 8: 8

# Создание репозитория курса на основе шаблона

В соответствии с заданием подготавливаем папки и создаем репозиторий курса на основе шаблона

```
[1032192035_pfur.ru@1032192035 ~]$ mkdir -p ~/work/study/2022-2023/"Операционные системы"
[1032192035_pfur.ru@1032192035 ~]$ cd ~/work/study/2022-2023/Операционные\ системы/
[1032192035_pfur.ru@1032192035 Операционные системы]$ gh repo create study_2022-2023_os-intro --template
=yamadharma/course-directory-student-template --public
GraphQL: Could not resolve to a Repository with the name 'yamadharma/course-directory-student-template'.
(repository)
[1032192035_pfur.ru@1032192035 Операционные системы]$ gh repo create study_2022-2023_os-intro --template
=yamadharma/course-directory-student-template --public
✓ Created repository Soblya/study_2022-2023_os-intro on GitHub
```

Figure 9: 9

## Настраиваем каталог курса, создаем необходимые каталоги, отправляем результат на сервер

```
[1032192035_pfur.ru@1032192035 infosec]$ make COURSE=infosec
[1032192035_pfur.ru@1032192035 infosec]$ git status
On branch master
Your branch is up to date with 'origin/master'.

Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        deleted:    package.json

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        labs/
        prepare

no changes added to commit (use "git add" and/or "git commit -a")
[1032192035_pfur.ru@1032192035 infosec]$ git add .
[1032192035_pfur.ru@1032192035 infosec]$ git commit -am 'feat(main): make course structure'
[1032192035_pfur.ru@1032192035 infosec]$ git push
Enumerating objects: 20, done
```

Figure 10: 10

Настроили и научились работать с git

- По итогу работы мы научились работать с Oracle VM VirtualBox, git, Linux, Markdown