

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ ФГАОУ ВО «СЕВЕРО-КАВКАЗСКИЙ
ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт цифрового развития

Кафедра инфокоммуникаций

Дисциплина: «основы кроссплатформенного программирования»

ОТЧЕТ

по лабораторной работе №1

Выполнил: студент 1 курса группы ИТС-21-1

Снадный Михаил Сергеевич

Проверил: к.ф.-м.н., доцент кафедры инфокоммуникаций

Воронкин Роман Александрович

Работа защищена с оценкой: _____

Ставрополь, 2021

Тема:

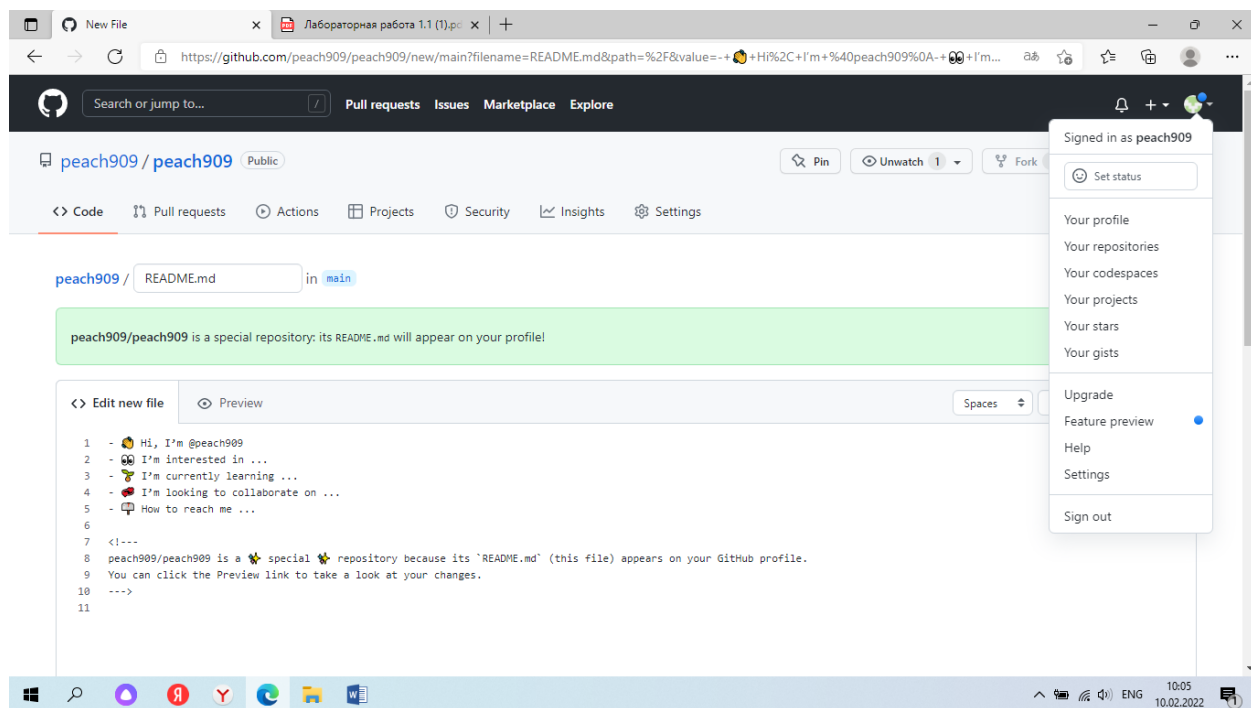
Основы языка Python

Цель работы:

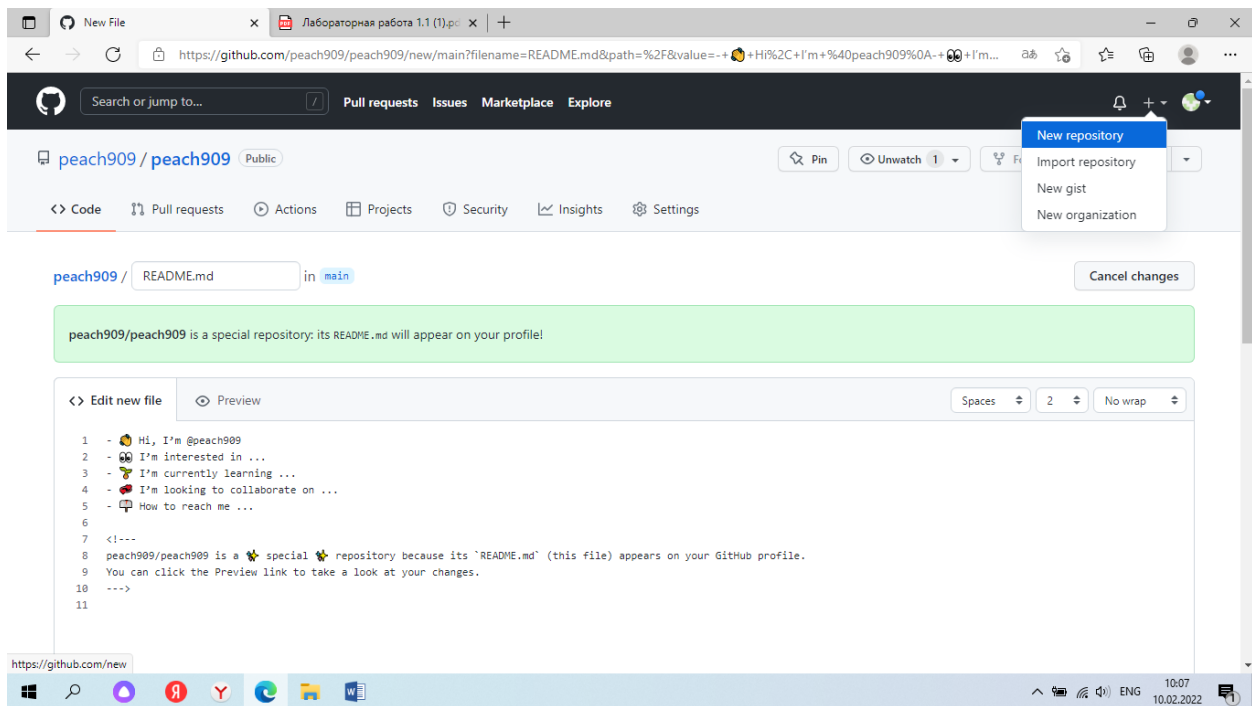
исследовать базовые возможности системы контроля версий Git и веб-сервиса для хостинга IT-проектов GitHub.

Ход работы:

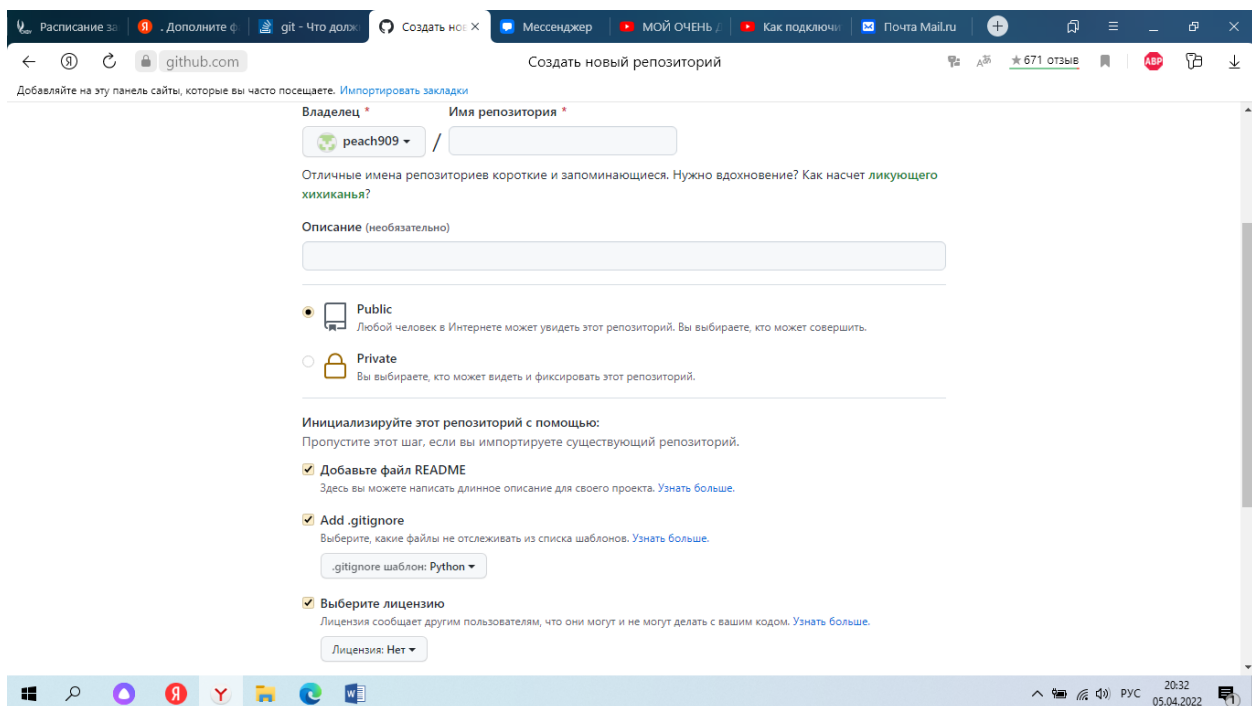
Зарегистрировался на сайте Git



Затем создал новый репозиторий



Заполнил настройки моего репозитория:



Скопировал адрес моего репозитория

peach909/peach-1 x Лабораторная работа 1.1 (1).pdf x

https://github.com/peach909/peach-1

Search or jump to... Pull requests Issues Marketplace Explore

peach909/peach-1 (Public) Pin Unwatch 1 Fork 0 Star 0

<> Code Issues Pull requests Actions Projects Wiki Security Insights Settings

Quick setup — if you've done this kind of thing before

Set up in Desktop or HTTPS SSH <https://github.com/peach909/peach-1.git>

Get started by creating a new file or uploading an existing file. We recommend every repository include a README, LICENSE, and .gitignore.

...or create a new repository on the command line

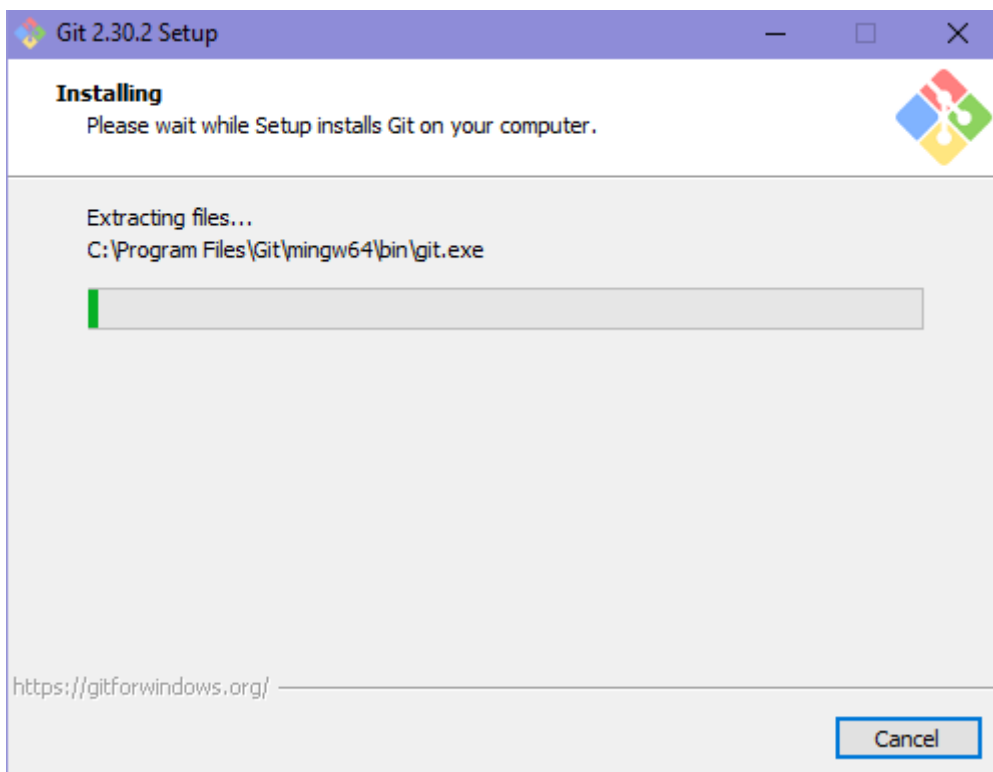
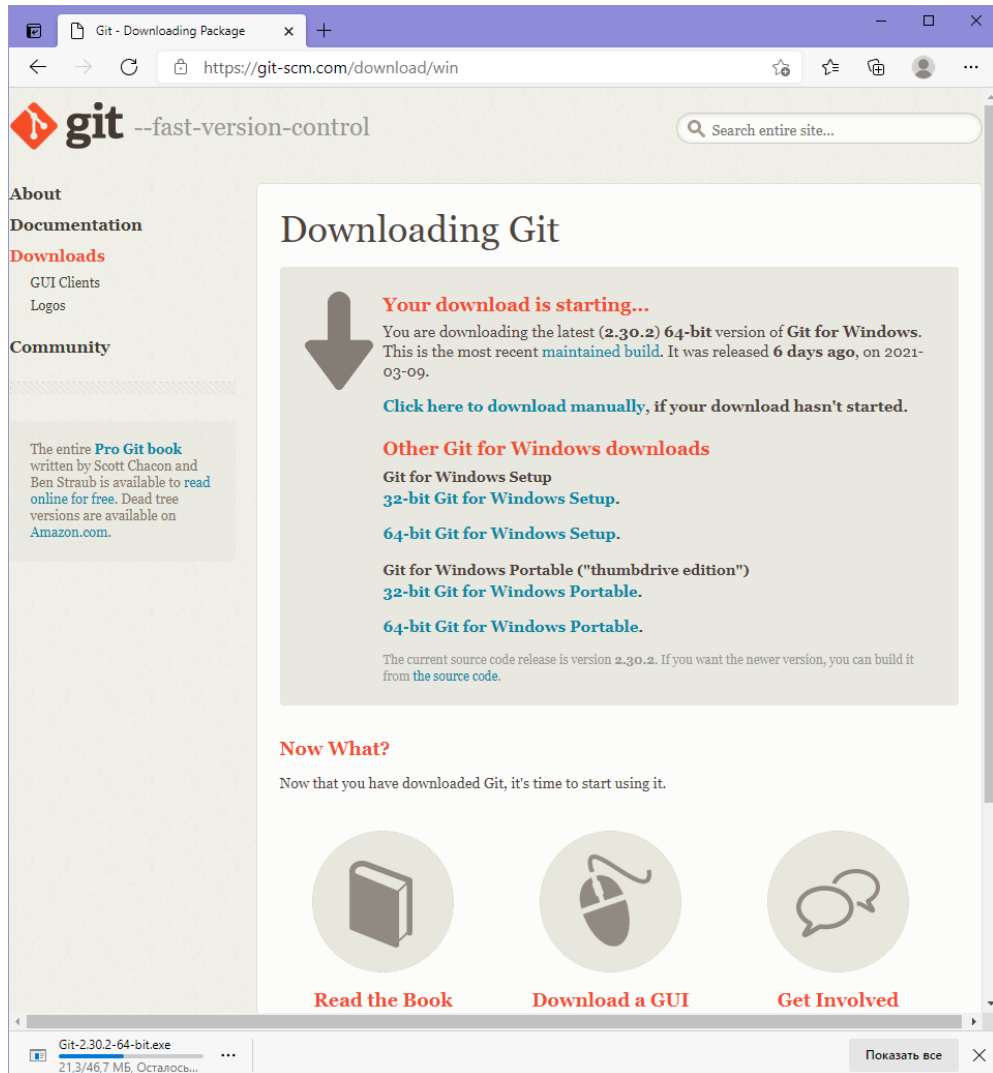
```
echo "# peach-1" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/peach909/peach-1.git
git push -u origin main
```

...or push an existing repository from the command line

```
git remote add origin https://github.com/peach909/peach-1.git
git branch -M main
```

Windows taskbar: 10:15 10.02.2022

Устанавливаем Git на компьютер.

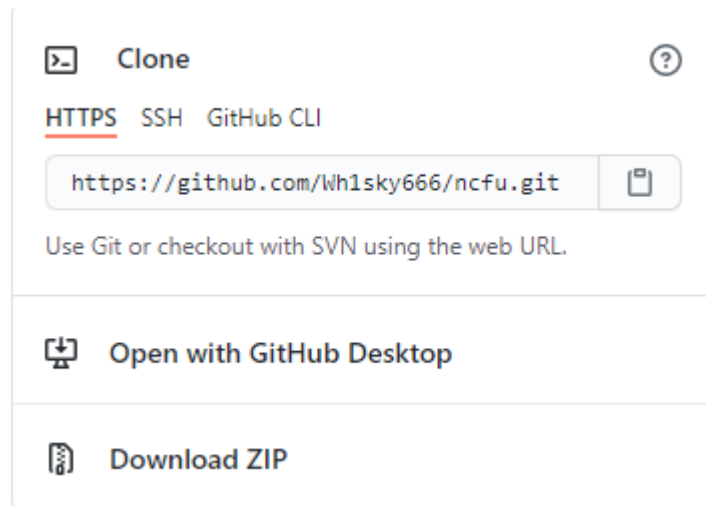


Для проверки того, установлен ли Git. Достаточно ввести команду, которая выведет нам версию Git в терминале.

```
C:\Users\Popa>git version
git version 2.30.2.windows.1
C:\Users\Popa>
```

Рисунок 5. Проверка установки Git.



Воспользуемся командой `>git clone`, чтобы скопировать репозиторий. После команды вводим ссылку, которую скопировали на Рисунке 6.



```
C:\Users\Popa>cd /d D:\ripZ
D:\ripZ>git clone https://github.com/Wh1sky666/ncfu.git
Cloning into 'ncfu'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (3/3), done.
D:\ripZ>
```

Напишем маленькую программу и сохраним ее в нашем репозитории.

```
Python 3.10 (64-bit)
Python 3.10.3 (tags/v3.10.3:a342a49, Mar 16 2022, 13:07:40) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 1+3
4
>>> print(персик)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'персик' is not defined
>>> print("персик")
персик
>>>
```

 .gitignore	15.03.2021 12:12	Текстовый докум...	2 КБ
 hi	15.03.2021 12:22	Python File	1 КБ

Добавим этот файл в сам репозиторий.

```
D:\ripZ\ncfu>git status
On branch main
Your branch is up to date with 'origin/main'.

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

nothing added to commit but untracked files present (use "git add" to track)
D:\ripZ\ncfu>git add hi.py
D:\ripZ\ncfu>git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   hi.py
```

Добавим коммит командой git commit.

```
D:\ripZ\ncfu>git commit -m "New!"
[main 7b94496] New!
1 file changed, 1 insertion(+)
create mode 100644 hi.py
```

```
D:\ripZ\ncfu>git add README.txt

D:\ripZ\ncfu>git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   README.txt
```

Теперь всю проделанную работу, все созданные файлы отправим в удаленный репозиторий GitHub, с помощью команды git push.

```
D:\ripZ\ncfu>git push
info: please complete authentication in your browser...
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 2 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 289 bytes | 289.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/Wh1sky666/ncfu.git
 507aa27..7b94496  main -> main
```

Ответы на контрольные вопросы:

1. Что такое СКВ и каково ее назначение?

Система контроля версий (СКВ) — это система, регистрирующая изменения в одном или нескольких файлах этих файлов.

2. В чем недостатки локальных и централизованных СКВ?

Это единая точка отказа, представленная централизованным сервером. Если этот сервер выйдет из строя на час, то в течение этого времени никто не сможет использовать контроль версий для сохранения изменений, над которыми работает, а также никто не сможет обмениваться этими изменениями с другими разработчиками.

3. К какой СКВ относится Git?

К распределённым системам контроля версий

4. В чем концептуальное отличие Git от других СКВ?

Git не хранит и не обрабатывает данные таким же способом как другие СКВ.

5. Как обеспечивается целостность хранимых данных в Git?

В Git для всего вычисляется хеш-сумма, и только потом происходит сохранение. В дальнейшем обращение к сохранённым объектам происходит по этой хеш-сумме.

6. В каких состояниях могут находиться файлы в Git? Как связаны эти состояния?

1) Зафиксированный значит, что файл уже сохранён в вашей локальной базе.

2) К изменённым относятся файлы, которые поменялись, но ещё не были зафиксированы.

3) Подготовленные файлы — это изменённые файлы, отмеченные для включения в следующий коммит.

7. Что такое профиль пользователя в GitHub?

Профиль - это наша публичная страница на GitHub, как и в социальных сетях. В нем другие пользователи могут посмотреть ваши работы.

8. Какие бывают репозитории в GitHub?

9. Укажите основные этапы модели работы с GitHub.

1) Регистрация.

2) Создание репозитория.

3) Клонирование репозитория.

4) Добавление новых файлов.

10. Как осуществляется первоначальная настройка Git после установки?

Убедимся, что Git установлен используя команду: `git version`.
Перейдём в папку с локальным репозиторием используя команду:
`cd /d <Расположения папки на компьютере>`. Свяжем локальный

репозиторий и удалённый командами: `git config --global user.name <YOUR_NAME>`
`git config --global user.email <EMAIL>`

11. Опишите этапы создания репозитория в GitHub.

1) В правом верхнем углу, рядом с аватаром есть кнопка с плюсиком, нажимая которую мы переходим к созданию нового репозитория.

2) В результате будет выполнен переход на страницу создания

репозитория. Наиболее важными на ней являются следующие поля: Имя репозитория. Описание (Description). Public/private. “Initialize this repository with a README” .gitignore и LICENSE

12. Какие типы лицензий поддерживаются GitHub при создании репозитория?

Microsoft Reciprocal License, The Code Project Open License (CPO), The Common Development and Distribution License (CDDL), The Microsoft Public License (Ms-PL), The Mozilla Public License 1.1 (MPL 1.1), The Common Public License Version 1.0 (CPL), The Eclipse Public License 1.0, The MIT License, The BSD License, The Apache License, Version 2.0, The Creative Commons Attribution-ShareAlike 2.5 License, The zlib/libpng License, A Public Domain dedication, The Creative Commons Attribution 3.0 Unported License, The Creative Commons)

13. Как осуществляется клонирование репозитория GitHub? Зачем нужно клонировать репозиторий?

После создания репозитория его необходимо клонировать на ваш компьютер. Для этого на странице репозитория необходимо найти кнопку Clone или Code и щелкнуть по ней, чтобы отобразить адрес репозитория для клонирования

Откройте командную строку или терминал и перейдите в каталог, куда вы хотите скопировать хранилище. Затем напишите `git clone` и введите адрес.

14. Как проверить состояние локального репозитория Git?

`git status`

15. Как изменяется состояние локального репозитория Git после выполнения следующих операций: добавления/изменения файла в локальный репозиторий Git; добавления нового/измененного файла под версионный контроль с помощью команды `git add` ; фиксации (коммита) изменений с помощью команды `git commit` и отправки изменений на сервер с помощью команды `git push` ?

Файлы обновятся на репозитории.

16. У Вас имеется репозиторий на GitHub и два рабочих компьютера, с помощью которых Вы можете осуществлять работу над некоторым проектом с использованием этого репозитория. Опишите последовательность команд, с помощью которых оба локальных репозитория, связанных с репозиторием GitHub будут находиться в синхронизированном состоянии.

`git clone.`

`git pull`

17. GitHub является не единственным сервисом, работающим с Git. Какие сервисы еще Вам известны? Приведите сравнительный анализ одного из таких сервисов с GitHub.

Ответ:

1) GitLab — альтернатива GitHub номер один. GitLab предоставляет не только веб-сервис для совместной работы, но и программное обеспечение с открытым исходным кодом.

2) BitBucket — это служба хостинга репозитория и управления версиями от Atlassian. Она тесно интегрирована с другими инструментами Atlassian — Jira, HipChat и Confluence.

18. Интерфейс командной строки является не единственным и далеко не самым удобным способом работы с Git. Какие Вам известны программные средства с графическим интерфейсом пользователя для работы с Git? Приведите как реализуются описанные в лабораторной работе операции Git с помощью одного из таких программных средств.

GitHub Desktop это совершенно бесплатное приложение с открытым исходным кодом, разработанное GitHub. С его помощью можно взаимодействовать с GitHub (что и не удивительно), а также с другими платформами (включая Bitbucket и GitLab).

Вывод:

исследовал базовые возможности системы контроля версий Git и веб-сервиса для хостинга IT-проектов GitHub.