

лабораторная работа №2

Студент: Мухамедияр Адиль
Группа: НКНбд-01-20

Цель:

Изучить идеологию и применение средств контроля версий.

Ход работы:

1) Создал учетную запись на github.

The screenshot shows the GitHub profile page for user 'adil-cpu'. The header includes the GitHub logo, a search bar, and navigation links for Pull requests, Issues, Marketplace, and Explore. The profile section features a green and white pixelated avatar, the username 'adil-cpu', an 'Edit profile' button, and a note 'Joined 7 days ago'. A 'ProTip!' banner encourages updating the profile. The 'Popular repositories' section is currently empty. The '11 contributions in the last year' section displays a calendar grid with contributions marked by green squares, primarily in April. A 'Contribution settings' dropdown is visible next to the calendar.

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

Overview Repositories 4 Projects Packages

ProTip! Updating your profile with your name, location, and a profile picture helps other GitHub users get to know you. Edit profile X

Popular repositories Customize your pins

adil-cpu Edit profile

Joined 7 days ago

11 contributions in the last year Contribution settings

May Jun Jul Aug Sep Oct Nov Dec Jan Feb Mar Apr

Mon

Wed

Fri

Learn how we count contributions Less More

2) Обозначил рабочий каталог как test, создав ее командой mkdir.

После перешел в данный каталог командой cd. Инициализировал систему git командой git init.

Создаю заготовку для файла README.md:

```
echo "# lab02" >> README.md
```

```
git add README.md
```

– Делаю первый коммит и выкладываем на github:

```
git commit -m "first commit"
```

```
git remote add origin git@github.com:<username>/sciproc-intro.git
```

```
git push -u origin master
```

```
[amukhamediyar@amukhamediyar ~]$ mkdir test
[amukhamediyar@amukhamediyar ~]$ cd test
[amukhamediyar@amukhamediyar test]$ echo "# lab02" >> README.md
[amukhamediyar@amukhamediyar test]$ git init
Initialized empty Git repository in /home/amukhamediyar/test/.git/
[amukhamediyar@amukhamediyar test]$ git add README.md
[amukhamediyar@amukhamediyar test]$ git commit -m "first commit"
[master (root-commit) 18b2760] first commit
 1 file changed, 1 insertion(+)
 create mode 100644 README.md
[amukhamediyar@amukhamediyar test]$ git branch -M main
[amukhamediyar@amukhamediyar test]$ git remote add origin git@github.com:adil-cpu/lab02
.git
[amukhamediyar@amukhamediyar test]$ git push -u origin main
The authenticity of host 'github.com (140.82.121.4)' can't be established.
RSA key fingerprint is SHA256:nThbg6kXUpJWGL7E1IG0CspRomTxdCARLviKw6E5SY8.
RSA key fingerprint is MD5:16:27:ac:a5:76:28:2d:36:63:1b:56:4d:eb:df:a6:48.
Are you sure you want to continue connecting (yes/no)? y
Please type 'yes' or 'no': yes
Warning: Permanently added 'github.com,140.82.121.4' (RSA) to the list of known hosts.
Counting objects: 3, done.
Writing objects: 100% (3/3), 217 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To git@github.com:adil-cpu/lab02.git
 * [new branch]      main -> main
Branch main set up to track remote branch main from origin.
```

3) Добавил файл лицензий:

```
[amukhamediyar@amukhamediyar test]$ wget https://creativecommons.org/licenses/by/4.0/legalcode.txt
--2021-04-29 20:13:39-- https://creativecommons.org/licenses/by/4.0/legalcode.txt
Распознаётся creativecommons.org (creativecommons.org)... 104.20.151.16, 104.20.150.16, 172.67.34.140, ...
Подключение к creativecommons.org (creativecommons.org)|104.20.151.16|:443... соединени
е установлено.
HTTP-запрос отправлен. Ожидание ответа... 200 OK
Длина: нет данных [text/plain]
Сохранение в: «legalcode.txt»

[ <=> ] 18 657 --.-K/s за 0,001s

2021-04-29 20:13:41 (14,6 MB/s) - «legalcode.txt» сохранён [18657]
```

4)Для начало посмотрим список имеющихся шаблонов

```
[amukhamediyar@amukhamediyar test]$ curl -L -s https://www.gitignore.io/api/list
lc,lc-bitrix,a-frame,actionsript,ada
adobe,advancedinstaller,adventuregamestudio,agda,al
alteraquartusii,altium,amplify,android,androidstudio
angular,anjuta,ansible,apachecordova,apachehadoop
appbuilder,appceleratortitanium,appcode,appcode+all,appcode+iml
appengine,aptanastudio,arcanist,archive,archives
archlinuxpackages,aspnetcore,assembler,ate,atmelstudio
ats,audio,automationstudio,autotools,autotools+strict
awr,azurefunctions,backup,ballerina,basercms
basic,batch,bazaar,bazel,bitrise
bitrix,bittorrent,blackbox,bloop,bluej
bookdown,bower,brickcc,buck,c
c++,cake,cakephp,cakephp2,cakephp3
calabash,carthage,certificates,ceylon,cfwheels
chefcookbook,chocolatey,clean,clion,clion+all
clion+iml,clojure,cloud9,cmake,cocoapods
cocos2dx,cocoscreator,code,code-java,codeblocks
codecomposerstudio,codeigniter,codeio,codekit,codesniffer
coffeescript,commonlisp,compodoc,composer,compressed
compressedarchive,compression,conan,concrete5,coq
cordova,craftcms,crashlytics,crbasic,crossbar
crystal,cs-cart,csharp,cuda,cvs
cypressio,d,dart,darteditor,data
database,datarecovery,dbeaver,defold,delphi
dframe,diff,direnv,diskimage,django
```

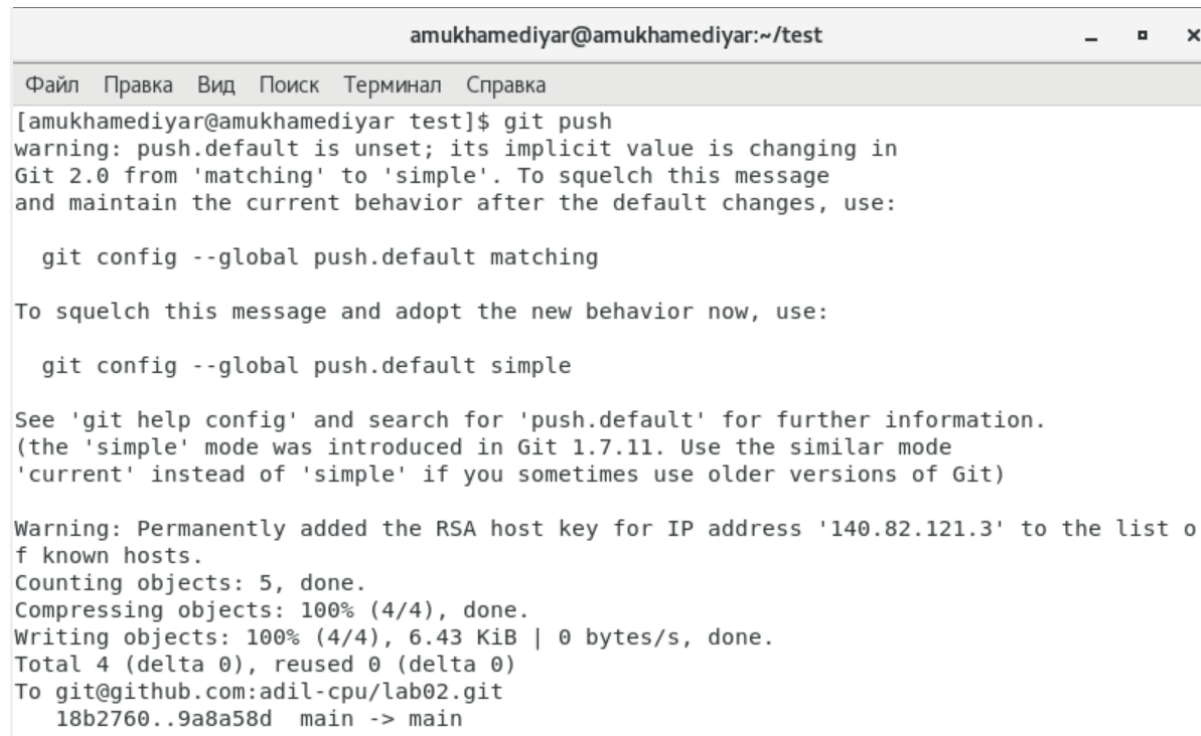
5)Теперь скачиваем шаблон для С

```
[amukhamediyar@amukhamediyar test]$ curl -L -s https://www.gitignore.io/api/c >> .gitignore
[amukhamediyar@amukhamediyar test]$ ls
legalcode.txt  README.md
```

6)Добавил новые файлы и выполнил КОММИТ

```
[amukhamediyar@amukhamediyar test]$ git add .
[amukhamediyar@amukhamediyar test]$ git commit -m "push"
[main 9a8a58d] push
2 files changed, 455 insertions(+)
create mode 100644 .gitignore
create mode 100644 legalcode.txt
```

7)Отправим на github



```
amukhamediyar@amukhamediyar:~/test
Файл  Правка  Вид  Поиск  Терминал  Справка
[amukhamediyar@amukhamediyar test]$ git push
warning: push.default is unset; its implicit value is changing in
Git 2.0 from 'matching' to 'simple'. To squelch this message
and maintain the current behavior after the default changes, use:

    git config --global push.default matching

To squelch this message and adopt the new behavior now, use:

    git config --global push.default simple

See 'git help config' and search for 'push.default' for further information.
(the 'simple' mode was introduced in Git 1.7.11. Use the similar mode
'current' instead of 'simple' if you sometimes use older versions of Git)

Warning: Permanently added the RSA host key for IP address '140.82.121.3' to the list o
f known hosts.
Counting objects: 5, done.
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 6.43 KiB | 0 bytes/s, done.
Total 4 (delta 0), reused 0 (delta 0)
To git@github.com:adil-cpu/lab02.git
18b2760..9a8a58d  main -> main
```


8) Инициализировал git-flow, установив префикс для ярлыков в v.

```
[amukhamediyar@amukhamediyar test]$ git flow init
```

```
Which branch should be used for bringing forth production releases?
```

```
- main
```

```
Branch name for production releases: [main]
```

```
Branch name for "next release" development: [develop]
```

```
How to name your supporting branch prefixes?
```

```
Feature branches? [feature/]
```

```
Release branches? [release/]
```

```
Hotfix branches? [hotfix/]
```

```
Support branches? [support/]
```

```
Version tag prefix? [] v
```

9) Проверяю, что я нахожусь на ветке develop.

```
[amukhamediyar@amukhamediyar test]$ git branch
```

```
* develop
```

```
main
```

11) Записал версию

```
[amukhamediyar@amukhamediyar test]$ echo "1.0.0" >> VERSION
```

```
[amukhamediyar@amukhamediyar test]$ ls
```

```
legalcode.txt  README.md  VERSION
```

10) Создаю релиз с версией 1.0.0

```
[amukhamediyar@amukhamediyar test]$ git flow release start 1.0.0
```

```
Switched to a new branch 'release/1.0.0'
```

```
Summary of actions:
```

- A new branch 'release/1.0.0' was created, based on 'develop'
- You are now on branch 'release/1.0.0'

```
Follow-up actions:
```

- Bump the version number now!
- Start committing last-minute fixes in preparing your release
- When done, run:

```
git flow release finish '1.0.0'
```

12) Добавил в индекс

```
[amukhamediyar@amukhamediyar test]$ git add *
```

```
[amukhamediyar@amukhamediyar test]$ git commit -m "first version"
```

```
[release/1.0.0 b96c270] first version
```

```
1 file changed, 1 insertion(+)
```

```
create mode 100644 VERSION
```

13) Залил релизную ветку в основную ветку

```
[amukhamediyar@amukhamediyar test]$ git flow release finish 1.0.0
Switched to branch 'main'
Merge made by the 'recursive' strategy.
 VERSION | 1 +
1 file changed, 1 insertion(+)
 create mode 100644 VERSION
fatal: no tag message?
Tagging failed. Please run finish again to retry.
```

14) Отправил данные на github

```
[amukhamediyar@amukhamediyar test]$ git push --all
Counting objects: 5, done.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 368 bytes | 0 bytes/s, done.
Total 4 (delta 2), reused 0 (delta 0)
remote: Resolving deltas: 100% (2/2), completed with 1 local object.
To git@github.com:adil-cpu/lab02.git
  9a8a58d..b545d61  main -> main
* [new branch]      develop -> develop
* [new branch]      release/1.0.0 -> release/1.0.0
[amukhamediyar@amukhamediyar test]$ git push --tags
Everything up-to-date
```

Вывод:

Изучил и понял как работать с системой контроля версий, с помощью командной строки, а именно с Git. Разобрал команды