

Лабораторная работа №1

Управление версиями

Румянцева Александра Сергеевна

17 сентября, 2022

Цели и задание на лабораторную работу

Цель: Изучить идеологию и применение средств контроля версий. Освоить умения по работе в git.

Задание: Лабораторная работа подразумевает изучение основных команд в git.

Выполнение лабораторной работы

1. Изучила теорию и указание к лабораторной работе.
2. Создадим учётную запись на <https://github.com> (рис. 1).

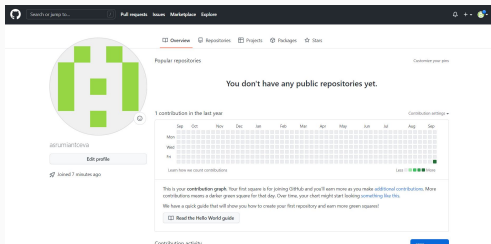


Figure 1: рис. 1. Учетная запись на Github.

Проверила установку git (рис. 2)

```
Honor@LAPTOP-L4D3GQ04 MINGW64 /d/Магистратура К/инф без
$ git --version
git version 2.37.3.windows.1
```

```
Honor@LAPTOP-L4D3GQ04 MINGW64 /d/Магистратура К/инф без
$ |
```

3. Сделала предварительную конфигурацию, указав имя email владельца репозитория (рис 3.).

```
Honor@LAPTOP-L4D3GQ04 MINGW64 /d/Магистатура К/инф без
$ git config --global user.name "asrumiantceva"

Honor@LAPTOP-L4D3GQ04 MINGW64 /d/Магистатура К/инф без
$ git config --global user.name "watercarrier@bk.ru"

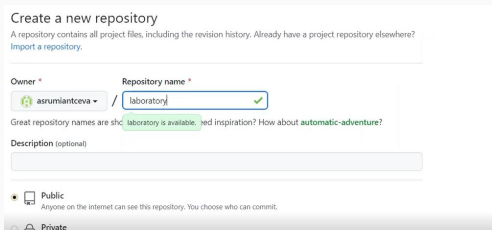
Honor@LAPTOP-L4D3GQ04 MINGW64 /d/Магистатура К/инф без
$ git config --list
diff.astextplain.textconv=astextplain
filter.lfs.clean=git-lfs clean -- %f
filter.lfs.smudge=git-lfs smudge -- %f
filter.lfs.process=git-lfs filter-process
filter.lfs.required=true
http.sslbackend=openssl
http.sslcainfo=C:/Program Files/Git/mingw64/ssl/certs/ca-bundle.crt
core.autocrlf=true
core.fscache=true
core.symlinks=true
pull.rebase=false
credential.helper=manager-core
credential.https://dev.azure.com.usehttppath=true
init.defaultbranch=master
user.email=watercarrier@bk.ru
user.name=watercarrier@bk.ru

Honor@LAPTOP-L4D3GQ04 MINGW64 /d/Магистатура К/инф без
$ |
```

Figure 3: рис. 3. Указание владельца репозитория.

4. Следующим шагом будет создание и подключение репозитория к GitHub.

В GitHub заходим в «repository» и создаём новый репозиторий (имя «laboratory», а заголовок для файла README). Копируем в консоль ссылку на репозиторий (для дальнейшей работы с файлами):



The screenshot shows the 'Create a new repository' page on GitHub. At the top, it says 'Create a new repository' and provides a brief explanation: 'A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)'. Below this, there are two main input fields: 'Owner' and 'Repository name'. The 'Owner' field is a dropdown menu showing 'asrumiantceva' with a green checkmark. The 'Repository name' field is a text input containing 'laboratory' with a green checkmark. Below these fields, there is a hint: 'Great repository names are short, lowercase, and no spaces. repository is available. Need inspiration? How about automatic-adventure?'. There is also a 'Description (optional)' text area. At the bottom, there are two radio buttons for 'Public' and 'Private'. The 'Public' option is selected, and a note below it says 'Anyone on the internet can see this repository. You choose who can commit.'

Figure 4: рис. 4. Создание репозитория.


```

asrumb@LAPTOP-4035Q04: ~/bin$ git commit -m 'first commit'
[master root-commit] first commit
22 files changed, 261 insertions(+)
create mode 100644 CHANGELOG.md
create mode 100644 README.md
create mode 100644 lab1/lab1.rar
create mode 100644 lab1/presentation/Makefile
create mode 100644 lab1/presentation/images/1.JPG
create mode 100644 lab1/presentation/images/2.JPG
create mode 100644 lab1/presentation/images/3.JPG
create mode 100644 lab1/presentation/images/4.JPG
create mode 100644 lab1/presentation/images/5.JPG
create mode 100644 lab1/presentation/pandoc-crossref.exe
create mode 100644 lab1/presentation/presentation.md
create mode 100644 lab1/presentation/presentation.pdf
create mode 100644 lab1/report/Makefile
create mode 100644 lab1/report/images/1.JPG
create mode 100644 lab1/report/images/2.JPG
create mode 100644 lab1/report/images/3.JPG
create mode 100644 lab1/report/images/4.JPG
create mode 100644 lab1/report/images/5.JPG
create mode 100644 lab1/report/pandoc-crossref.exe
create mode 100644 lab1/report/report.docx
create mode 100644 lab1/report/report.md
create mode 100644 lab1/report/report.pdf
asrumb@LAPTOP-4035Q04: ~/bin$ git push
error: src refspec main does not match any
error: failed to push some refs to 'https://github.com/asrumbantcev/laboratory.git'

```

Figure 6: рис. 6. Добавление файлов в репозиторий 2.

```
$ git push
Enumerating objects: 43, done.
Counting objects: 100% (43/43), done.
Delta compression using up to 4 threads
Compressing objects: 100% (42/42), done.
Writing objects: 95% (40/42), 16.38 MiB | 106.00 KiB/s
```

Figure 7: рис. 7. Добавление файлов в репозиторий 3.

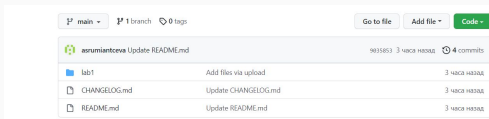


Figure 8: рис. 8. Добавленные файлы в репозиторий.

1. Что такое системы контроля версий (VCS) и для решения каких задач они предназначены?

Системы контроля версий (Version Control System, VCS) применяются при работе нескольких человек над одним проектом.

2. Объясните следующие понятия VCS и их отношения: хранилище, commit, история, рабочая копия.

Хранилище (repository), или репозиторий — место хранения всех версий и служебной информации.

Commit («[трудовой] вклад», не переводится) — синоним версии; процесс создания новой версии. Рабочая копия (working copy) — текущее состояние файлов проекта, основанное на версии, загруженной из хранилища (обычно на последней).

5. Опишите порядок работы с общим хранилищем VCS.

Для последующей идентификации пользователя на сервере репозитория необходимо сгенерировать пару ключей (приватный и открытый):

```
ssh-keygen -C "Имя Фамилия work@mail"
```

Ключи сохраняются в каталоге ~/.ssh/.

Скопировав из локальной консоли ключ в буфер обмена

```
cat ~/.ssh/id_rsa.pub | xclip -sel clip
```

вставляем ключ в появившееся на сайте поле.

9. Что такое и зачем могут быть нужны ветви (branches)?

Ветки нужны для того, чтобы программисты могли вести совместную работу над проектом и не мешать друг другу при этом. Кроме того, с помощью branches решаются следующие проблемы: нужно постоянно создавать архивы с рабочим кодом, сложно “переключаться” между архивами, сложно перетаскивать изменения между архивами, легко что-то напутать или потерять.

*более подробно на все контрольные вопросы описаны ответы в отчете

Я изучила идеологию и применение средств контроля версий. Освоила умения по работе в git.