

Лабораторная работа №1. Система контроля версий git. Язык разметки Markdown

Дисциплина: Математическое моделирование

Ганина Т. С.

21 февраля 2025

Группа НФИбд-01-22

Российский университет дружбы народов, Москва, Россия

Информация

- Ганина Таисия Сергеевна
- Студентка 3го курса, группа НФИбд-01-22
- Фундаментальная информатика и информационные технологии
- Российский университет дружбы народов
- Ссылка на репозиторий гитхаба `tsganina`

Вводная часть

Целью данной работы является приобретение навыков работы с системой контроля версий git. Работа с репозиториями, внесение изменений, индексация, коммиты.

1. Подготовить пространство для работы.
2. Создать проект.
3. Внести изменения в проект.
4. Провести индексацию изменений
5. Работа с тегами версий.
6. Отменить локальные изменения (до индексации).
7. Отменить проиндексированные изменения (перед коммитом).
8. Отменить коммит.
9. Удалить коммиты из ветки.
10. Удаление тега.
11. Внести изменения в коммиты.
12. Переместить файлы.

13. Рассмотреть Git внутри: Каталог `.git`.
14. Работа непосредственно с объектами `git`.
15. Навигация по веткам. Изменения в ветке `master`.
16. Слияние веток. Создание конфликта.
17. Разрешение конфликтов.
18. Сброс ветки `style`. Сброс ветки `master`.
19. Перебазирование.
20. Слияние в ветку `master`.
21. Клонирование репозитория. Просмотр клонированного репозитория.
22. Что такое `origin`? Удаленные ветки. Изменение оригинального репозитория.
23. Слияние извлеченных изменений.
24. Извлечение общих изменений.

Работа с git

```
taisiya@tsganina:~$ mkdir hello  
cd hello  
touch hello.html  
echo "Hello, World!" > hello.html  
taisiya@tsganina:~/hello$ cat hello.html  
Hello, World!  
taisiya@tsganina:~/hello$
```

Рис. 1: Создание проекта

```
taisiya@tsganina:~/hello$ git status
Текущая ветка: master
нечего коммитить, нет изменений в рабочем каталоге
taisiya@tsganina:~/hello$
```

Рис. 2: Команда git status, чтобы проверить текущее состояние репозитория

```
taisiya@tsganina:~/hello$ gedit hello.html
taisiya@tsganina:~/hello$ git add hello.html
taisiya@tsganina:~/hello$ gedit hello.html
taisiya@tsganina:~/hello$ git status
Текущая ветка: master
Изменения, которые будут включены в коммит:
  (используйте «git restore --staged <файл>...», чтобы убрать из индекса)

    изменено:   hello.html

Изменения, которые не в индексе для коммита:
  (используйте «git add <файл>...», чтобы добавить файл в индекс)
  (используйте «git restore <файл>...», чтобы отменить изменения в рабочем каталоге)

    изменено:   hello.html

taisiya@tsganina:~/hello$
```

Рис. 3: Одно изменение проиндексировано, второе нет

Вместо поиска до хэш,
использовала ^,
обозначающее «родитель
v1». Вместо обозначения v1^
можно использовать v1~1.
Это обозначение можно
определить как «первую
версию предшествующую
v1»

```
tsisyatsganina:~/hello$ git tag v1
tsisyatsganina:~/hello$ git checkout v1^
cat hello.html
Примечание: переключение на «v1^».

Вы сейчас в состоянии «отсоединённого указателя HEAD». Можете осматриваться,
вносить экспериментальные изменения и зафиксировать их, также можете
отменить любые коммиты, созданные в этом состоянии, не затрагивая другие
ветки, переключившись обратно на любую ветку.

Если хотите создать новую ветку для сохранения созданных коммитов, можете
сделать это (сейчас или позже), используя команду switch с параметром «с».
Например:

    git switch -с <новая-ветка>

или отмените эту операцию с помощью:

    git switch -

Отключите этот совет, установив переменную конфигурации
advice.detachedHead в значение false

HEAD сейчас на e758df Added standard HTML page tags
<html>
  <body>
    <h1>Hello, World!</h1>
  </body>
</html>
tsisyatsganina:~/hello$
```

Рис. 4: git tag v1

```
taisiya@tsganina:~/hello$ gedit hello.html
taisiya@tsganina:~/hello$ git status
Текущая ветка: master
Изменения, которые не в индексе для коммита:
  (используйте «git add <файл>...», чтобы добавить файл в индекс)
  (используйте «git restore <файл>...», чтобы отменить изменения в рабочем каталоге)

    изменено:      hello.html

индекс пуст (используйте «git add» и/или «git commit -a»)
taisiya@tsganina:~/hello$ git checkout hello.html
Updated 1 path from the index
taisiya@tsganina:~/hello$ git status
Текущая ветка: master
ничего коммитить, нет изменений в рабочем каталоге
taisiya@tsganina:~/hello$ cat hello.html
<html>
  <head>
  </head>
  <body>
    <h1>Hello, World!</h1>
  </body>
</html>
taisiya@tsganina:~/hello$
```

Рис. 5: Отменить локальные изменения (до индексации)

```
taisiya@tsganina:~/hello$ gedit hello.html
taisiya@tsganina:~/hello$ git add hello.html
taisiya@tsganina:~/hello$ git status
Текущая ветка: master
Изменения, которые будут включены в коммит:
  (используйте «git restore --staged <файл>...», чтобы убрать из индекса)
    изменено:   hello.html

taisiya@tsganina:~/hello$ git reset HEAD hello.html
Непроиндексированные изменения после сброса:
M   hello.html
taisiya@tsganina:~/hello$ git checkout hello.html
Updated 1 path from the index
taisiya@tsganina:~/hello$ git status
Текущая ветка: master
ничего коммитить, нет изменений в рабочем каталоге
taisiya@tsganina:~/hello$
```

Рис. 6: Отменить проиндексированные изменения (перед коммитом)

Работа непосредственно с объектами git.

Посмотрим последний коммит в репозиторий. SHA1 хэш в нашей системе. Мы можем вывести дерево каталогов, ссылка на который идет в коммите. Это должно быть описание файлов (верхнего уровня) в нашем проекте (для конкретного коммита). Вывод каталога lib, вывод файла hello.html.

```
taisiya@tsganina:~/hello$ git log --max-count=1
commit fe9904d086e78396b1c22eb37f9435e5de3af3f9 (HEAD -> master)
Author: tsganina <ta1.sergeevna@yandex.com>
Date: Sat Feb 15 14:41:10 2025 +0300

    Added index.html.

taisiya@tsganina:~/hello$ git cat-file -t fe9904d
commit
taisiya@tsganina:~/hello$ git cat-file -p fe9904d
tree 711ac7a9245185717a3e108fe149fc6b821b4eb
parent 7df96859d482c6cb77042f300441f95af22b6a85
author tsganina <ta1.sergeevna@yandex.com> 1739619670 +0300
committer tsganina <ta1.sergeevna@yandex.com> 1739619670 +0300

Added index.html.
taisiya@tsganina:~/hello$ git cat-file -p 711ac7a
100644 blob cfsbc0ea78289c0be141b578ca407f1159bd25bc    index.html
040800 tree aeae7cd371d664fb2afee8f60e3a2db98f853ce    lib
taisiya@tsganina:~/hello$ git cat-file -p aeae7c
100644 blob b69ab178cf5175103dfb1b4b908b0bd80831119a0    hello.html
taisiya@tsganina:~/hello$ git cat-file -p b69ab17
<!-- Author: Taisia S. Ganina (1132226429@pfur.ru) -->
<html>
  <head>
  </head>
  <body>
    <h1>Hello, World!</h1>
  </body>
</html>
taisiya@tsganina:~/hello$
```

Рис. 7: Работа непосредственно с объектами git

```
commit 64a83a23934628601807fa3be276b9ad3ac9a0cf
Author: tsganina <ta1.sergeevna@yandex.com>
Date: Sat Feb 15 13:54:58 2025 +0300

Initial Commit

taisiya@tsganina:~/hello$ git cat-file -t 64a83a23934628601807fa3be276b9ad3ac9a0cf
a0cf
commit
taisiya@tsganina:~/hello$ git cat-file -p 64a83a23934628601807fa3be276b9ad3ac9a0cf
tree 523cd318fc3b74ad3852cd34a74720ad6f7dd422
author tsganina <ta1.sergeevna@yandex.com> 1739616898 +0300
committer tsganina <ta1.sergeevna@yandex.com> 1739616898 +0300

Initial Commit
taisiya@tsganina:~/hello$ git cat-file -p 523cd31100644 blob 8ab686eafeb1f44702738c8b0f24f2567c36da6d hello.html
taisiya@tsganina:~/hello$ git cat-file -p 8ab686e
Hello, World!
taisiya@tsganina:~/hello$
```

Рис. 8: Самостоятельное исследование 1


```

tatsiya@tsganina:~/hello$ git cat-file -p 7df0805
tree 5e5438c90e1c7ee53de953fad6ee712046cfdb6
parent fbc03d5d6d9f59a8c083142541f8a366d7f5837b0
author tsganina <ta1.sergeevnadyandex.com> 1739619537 +0300
committer tsganina <ta1.sergeevnadyandex.com> 1739619537 +0300

Moved hello.html to lib

tatsiya@tsganina:~/hello$ git cat-file -p fbc83d5
tree aeeae7cd371d664fb2afeef68e3a2d08f053ce
parent 0b677b88f537736bb73fe78c47e844e8bd7c565
author tsganina <ta1.sergeevnadyandex.com> 17396193287 +0300
committer tsganina <ta1.sergeevnadyandex.com> 1739619389 +0300

Add an author/email comment

tatsiya@tsganina:~/hello$ git cat-file -p 0b677b8
tree e8da7e4cc558a041bcdb34ad3682e9e03f26fe
parent e75addf68e315c7312ae7d12304b8a845654f50b
author tsganina <ta1.sergeevnadyandex.com> 1739617813 +0300
committer tsganina <ta1.sergeevnadyandex.com> 1739617813 +0300

Added HTML header

tatsiya@tsganina:~/hello$ git cat-file -p e75addf
tree 5a08fc0510990b3a0d3ef11bda23005ce7676b8
parent d4e8f68cdd1b074db31b424200299ad78400
author tsganina <ta1.sergeevnadyandex.com> 1739617664 +0300
committer tsganina <ta1.sergeevnadyandex.com> 1739617664 +0300

Added standard HTML page tags

tatsiya@tsganina:~/hello$ git cat-file -p d4e80f6
tree 629377ed8eb9afede9fddcb7b48b1e4a1d3dc56
parent 64a83a23934628601807fa3be276b9ad3ac9a0cf
author tsganina <ta1.sergeevnadyandex.com> 1739617339 +0300
committer tsganina <ta1.sergeevnadyandex.com> 1739617339 +0300

Added h1 tag

tatsiya@tsganina:~/hello$ git cat-file -p 64a83a2
tree 523cd18fc3b74d3852cd34a74720a06f7d6422
author tsganina <ta1.sergeevnadyandex.com> 1739616898 +0300
committer tsganina <ta1.sergeevnadyandex.com> 1739616898 +0300

Initial Commit

tatsiya@tsganina:~/hello$ git cat-file -p 523cd31
100644 blob 8ab080aef6b1f44702738c8b0f24f2567c36da6d    hello.html
tatsiya@tsganina:~/hello$ git cat-file -p 8ab080e
Hello, World!
tatsiya@tsganina:~/hello$

```

Рис. 9: Самостоятельное исследование 2

Потом используем команду `git checkout` для переключения между ветками.

```
taisiya@tsganina:~/hello$ git checkout master
cat lib/hello.html
Переключились на ветку «master»
<!-- Author: Taisia S. Ganina (1132226429@pfur.ru) -->
<html>
  <head>
  </head>
  <body>
    <h1>Hello, World!</h1>
  </body>
</html>
taisiya@tsganina:~/hello$ git checkout style
cat lib/hello.html
Переключились на ветку «style»
<!-- Author: Taisia S. Ganina (1132226429@pfur.ru) -->
<html>
  <head>
    <link type="text/css" rel="stylesheet"
media="all" href="style.css" />
  </head>
  <body>
    <h1>Hello, World!</h1>
  </body>
</html>
taisiya@tsganina:~/hello$
```

Рис. 10: Переключения между ветками

```
taisiya@tsganina:~/hello$ git commit -m "Added HTML header"
[master 0b677b8] Added HTML header
 1 file changed, 2 insertions(+)
taisiya@tsganina:~/hello$ git log
commit 0b677b80f537736cbb73fef8c47e844e8bd7c565 (HEAD -> master)
Author: tsganina <tai.sergeevna@yandex.com>
Date: Sat Feb 15 14:10:13 2025 +0300

    Added HTML header

commit e75a8df68e315c7312ae7d12304b8a045654f58b
Author: tsganina <tai.sergeevna@yandex.com>
Date: Sat Feb 15 14:07:44 2025 +0300

    Added standard HTML page tags

commit d4e86f696c9dd1b0f4db31b4242b0296ad278480
Author: tsganina <tai.sergeevna@yandex.com>
Date: Sat Feb 15 14:02:19 2025 +0300

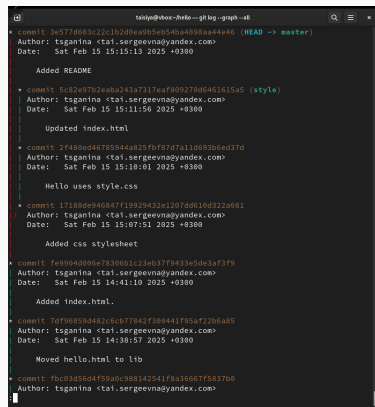
    Added h1 tag

commit 64a83a23934628601807fa3be276b9ad3ac9a0cf
Author: tsganina <tai.sergeevna@yandex.com>
Date: Sat Feb 15 13:54:58 2025 +0300

    Initial Commit
taisiya@tsganina:~/hello$
```

Рис. 11: Кратчайший маршрут до разрыва соединения

Создадим файл README в ветке master.
Сделаем коммит изменений README.md в
ветку master. Просмотр отличающихся веток.



```
tai@p@vbox:~$ git log --graph --all
* commit 3e577d603c22c1b2d0ea9b5eb54ba4898aa44e46 (HEAD -> master)
  Author: tsganina <tai.sergeevna@yandex.com>
  Date: Sat Feb 15 15:15:13 2025 +0300

    Added README

* commit 5c82e97b2eaba243a7317eaf809270d6461615a5 (style)
  Author: tsganina <tai.sergeevna@yandex.com>
  Date: Sat Feb 15 15:11:56 2025 +0300

    Updated index.html

* commit 2f480ed46785944a825fbf97d7a11d693b6ed37d
  Author: tsganina <tai.sergeevna@yandex.com>
  Date: Sat Feb 15 15:18:01 2025 +0300

    Hello uses style.css

* commit 17188de946847f19929432e1207dd610d322a081
  Author: tsganina <tai.sergeevna@yandex.com>
  Date: Sat Feb 15 15:07:51 2025 +0300

    Added css stylesheet

* commit fe9904d906e78306b1c23eb37f9433e5de3af3f9
  Author: tsganina <tai.sergeevna@yandex.com>
  Date: Sat Feb 15 14:41:10 2025 +0300

    Added index.html.

* commit 7d196859d482c6cb77042f309441f95af22b6a85
  Author: tsganina <tai.sergeevna@yandex.com>
  Date: Sat Feb 15 14:38:57 2025 +0300

    Moved hello.html to lib

* commit fbc03d56d4f59a0c988142541f8a36667f5837b0
  Author: tsganina <tai.sergeevna@yandex.com>
```

Рис. 12: git log --graph --all

Путем периодического слияния ветки master с веткой style мы можем переносить из master любые изменения и поддерживать совместимость изменений style с изменениями в основной ветке. Но что если изменения в ветке master конфликтуют с изменениями в style?

```
taisiya@tsganina:~/hello$ git checkout style
git merge master
Репозиторий на ветку «style»
Ассоциинные lib/hello.html
КОНФЛИКТ (содержимое): Конфликт слияния в lib/hello.html
Сбой автоматического слияния; исправьте конфликты, затем зафиксируйте результа
t.
taisiya@tsganina:~/hello$ cat lib/hello.html
cat: lib/hello.html: Нет такого файла или каталога
taisiya@tsganina:~/hello$ cat lib/hello.html
<!-- Author: Taisia S. Ganina (1132226429@pfur.ru) -->
<html>
  <head>
<----- HEAD
      <link type="text/css" rel="stylesheet"
media="all" href="style.css" />
-----
      <!-- no style -->
>>>>>> master
      </head>
      <body>
          <h1>Hello, World!</h1>
      </body>
</html>
taisiya@tsganina:~/hello$
```

Рис. 13: Вернемся к ветке style и попытаемся объединить ее с новой веткой master

```

* commit c1db7ef65458c760e1e5e067ad191014cc4c072a (HEAD -> style)
Merge: 0bcf53f 5ba398e
Author: tsganina <ta1.sergeevna@yandex.com>
Date: Sat Feb 15 15:24:55 2025 +0300

    Merged master fixed conflict.

* commit 5ba398e706ac7754f90349fff7a81aef4c5a7be2c (master)
Author: tsganina <ta1.sergeevna@yandex.com>
Date: Sat Feb 15 15:20:37 2025 +0300

    Life is great

* commit 0bcf53f4ba31822402fa0ee975dc017bd9ce077a
Merge: 5c82e97 3e577d6
Author: tsganina <ta1.sergeevna@yandex.com>
Date: Sat Feb 15 15:17:09 2025 +0300

    Merge branch 'master' into style

* commit 3e577d603c22c1b2d0ea9b5eb54ba4898aa44e46
Author: tsganina <ta1.sergeevna@yandex.com>
Date: Sat Feb 15 15:15:13 2025 +0300

    Added README

* commit 5c82e97b2eaba243a7317eaf809270d6461615a5
Author: tsganina <ta1.sergeevna@yandex.com>
Date: Sat Feb 15 15:11:56 2025 +0300

    Updated index.html

* commit 2f480ed46785944a825fbf87d7a11d693b6ed37d
Author: tsganina <ta1.sergeevna@yandex.com>
Date: Sat Feb 15 15:10:01 2025 +0300

    Hello uses style.css

```

Рис. 14: Внесём изменения в lib/hello.html

Вернемся на ветке style к точке перед тем, как мы слили ее с веткой master. Мы можем сбросить ветку к любому коммиту. По сути, это изменение указателя ветки на любую точку дерева коммитов. В этом случае мы хотим вернуться в ветке style в точку перед слиянием с master. Нам необходимо найти последний коммит перед слиянием.

```
tsl@tslpyan@nina:~/helios$ git reset --hard 5c22e97
tsl@tslpyan@nina:~/helios$ git log --graph --all
* commit 5c22e97b2d8a2d3a7317eaf00927b06461013a5 (HEAD)
  Author: tsarina <ta1.sergeev@yandex.com>
  Date: Sat Feb 15 15:20:37 2025 +0300

    Life is great

* commit 5c22e97b2d8a2d3a7317eaf00927b06461013a5 (HEAD -> style)
  Author: tsarina <ta1.sergeev@yandex.com>
  Date: Sat Feb 15 15:11:56 2025 +0300

    Added README

* commit 3740b0e4079264e4126f9f8707a1a09316ed176
  Author: tsarina <ta1.sergeev@yandex.com>
  Date: Sat Feb 15 15:19:01 2025 +0300

    Updated index.html

* commit 5c130da9d08e1f2015d32e120740109122d981
  Author: tsarina <ta1.sergeev@yandex.com>
  Date: Sat Feb 15 15:07:51 2025 +0300

    Hello uses style.css

* commit f4b80a080e7a19021c21b0177b031e5d3a370f9
  Author: tsarina <ta1.sergeev@yandex.com>
  Date: Sat Feb 15 14:41:18 2025 +0300

    Added css stylesheet

* commit 5c130da9d08e1f2015d32e120740109122d981
  Author: tsarina <ta1.sergeev@yandex.com>
  Date: Sat Feb 15 15:07:51 2025 +0300

    Added index.html.
```

Рис. 15: Сброс ветки style

```
taisha@vbox:~/hello -- git log --graph --all
Author: tsganina <ta1.sergeevna@yandex.com>
taisiya@tsganina:~/hello$ git reset --hard 3e577d6
Указатель HEAD сейчас на коммите 3e577d6 Added README
taisiya@tsganina:~/hello$ git log --graph --all
* commit 3e577d603c22c1b2d0ea9b5eb54ba4898aa44e46 (HEAD -> master)
  Author: tsganina <ta1.sergeevna@yandex.com>
  Date: Sat Feb 15 15:15:13 2025 +0300

    Added README

* commit 5c82e97b2eaba243a7317eaf809270d6461615a5 (style)
  Author: tsganina <ta1.sergeevna@yandex.com>
  Date: Sat Feb 15 15:11:56 2025 +0300

    Updated index.html

* commit 2f480ed46785944a825fbf87d7a11d693b6ed37d
  Author: tsganina <ta1.sergeevna@yandex.com>
  Date: Sat Feb 15 15:10:01 2025 +0300

    Hello uses style.css

* commit 17188de946847f19929432e1207dd610d322a081
  Author: tsganina <ta1.sergeevna@yandex.com>
  Date: Sat Feb 15 15:07:51 2025 +0300

    Added css stylesheet

* commit fe9904d006e78306b1c23eb37f9433e5de3af3f9
  Author: tsganina <ta1.sergeevna@yandex.com>
  Date: Sat Feb 15 14:41:10 2025 +0300

    Added index.html.

* commit 7df96059d482c6cb77042f300441f95af22b6a85
  Author: tsganina <ta1.sergeevna@yandex.com>
  Date: Sat Feb 15 14:38:57 2025 +0300

    Moved hello.html to lib
```

Рис. 16: Сброс ветки master

Мы поддерживали соответствие ветки style с веткой master (с помощью rebase), теперь давайте сольем изменения style в ветку master.

```
ts@tsiyatstgmlai:~/hello$
ts@tsiyatstgmlai:~/hello$
ts@tsiyatstgmlai:~/hello$
ts@tsiyatstgmlai:~/hello$ git checkout master
git merge style
$openchrome --a --key --master
$openchrome 3a0770b..3a0736d
fast-forward
index.html | 4 +---
lib/hello.html | 2 +-
lib/style.css | 8 +---
3 files changed, 9 insertions(+)
create mode 100644 lib/style.css
ts@tsiyatstgmlai:~/hello$ git log
commit 3a0770b3c22c1b28ba09b5eb5ba09bba4e0e (HEAD -> master, style)
Author: ts@tsiyatstgmlai:~/hello$
Date: Sat Feb 15 15:11:56 2025 +0300

    Updated index.html

commit 01c8b5c5460c3a10a43d8c0b1c07b7f181cf
Author: ts@tsiyatstgmlai:~/hello$
Date: Sat Feb 15 15:10:01 2025 +0300

    Hello user style.css

commit 3a0770b3c22c1b28ba09b5eb5ba09bba4e0e
Author: ts@tsiyatstgmlai:~/hello$
Date: Sat Feb 15 15:07:51 2025 +0300

    Added css stylesheet

commit 3a0770b3c22c1b28ba09b5eb5ba09bba4e0e
Author: ts@tsiyatstgmlai:~/hello$
Date: Sat Feb 15 15:15:13 2025 +0300

    Added README
```

Рис. 17: Слияние в ветку master

```

taisiya@tsganina:~/hello$ cd ..
pwd
ls
/home/taisiya
  backup          Видео
  bin             'Внешний курс инфобез'
  hello          Документы
  lab07.sh~      Загрузки
  pandoc-2.19.2  Изображения
  pandoc-2.19.2-linux-amd64.tar.gz Музыка
  pandoc-crossref  Общедоступные
  pandoc-crossref.1 'Рабочий стол'
  pandoc-crossref-Linux.tar.xz Шаблоны
  work
taisiya@tsganina:~$ git clone hello cloned_hello
ls
Клонирование в «cloned_hello»...
готово.
  backup          work
  bin             Видео
  cloned_hello    'Внешний курс инфобез'
  hello          Документы
  lab07.sh~      Загрузки
  pandoc-2.19.2  Изображения
  pandoc-2.19.2-linux-amd64.tar.gz Музыка
  pandoc-crossref  Общедоступные
  pandoc-crossref.1 'Рабочий стол'
  pandoc-crossref-Linux.tar.xz Шаблоны
taisiya@tsganina:~$

```

Рис. 18: Создадим клон репозитория hello

```
taisiya@tsganina:~/cloned_hello$ ls  
index.html  lib  README.md  
taisiya@tsganina:~/cloned_hello$
```

Рис. 19: Просмотр клонированного репозитория

Мы видим, что клонированный репозиторий знает об имени по умолчанию удаленного репозитория. Давайте посмотрим, можем ли мы получить более подробную информацию об имени по умолчанию.

```
taisiya@tsganina:~/cloned_hello$ git remote
origin
taisiya@tsganina:~/cloned_hello$ git remote show origin
* внешний репозиторий origin
  URL для извлечения: /home/taisiya/hello
  URL для отправки: /home/taisiya/hello
  HEAD ветка: master
  Внешние ветки:
    master отслеживается
    style отслеживается
  Локальная ветка, настроенная для «git pull»:
    master будет слита с внешней веткой master
  Локальная ссылка, настроенная для «git push»:
    master будет отправлена в master (уже актуальна)
taisiya@tsganina:~/cloned_hello$
```

Рис. 20: Более подробная информация об имени по умолчанию

```
taisiya@tszanina:~/hello$ cd ../cloned_hello
git fetch
git log --all
commit 3a8f38d41447f3178f8572fe33f8e41453f408d9 (HEAD -> master, origin/style,
origin/master, origin/HEAD)
Author: tsganina <ta1.sergeevna@yandex.com>
Date: Sat Feb 15 15:11:56 2025 +0300

    Updated index.html

commit d1cdb5c5b66cda1bb8b2ed8c6b02cb7fbf33c9cf
Author: tsganina <ta1.sergeevna@yandex.com>
Date: Sat Feb 15 15:10:01 2025 +0300

    Hello uses style.css

commit 393a90e1024ddeb6b4a4f790caf34ad234f15094
Author: tsganina <ta1.sergeevna@yandex.com>
Date: Sat Feb 15 15:07:51 2025 +0300

    Added css stylesheet

commit 3e577d603c22c1b2d0ea9b5eb54ba4098aa44e46
Author: tsganina <ta1.sergeevna@yandex.com>
Date: Sat Feb 15 15:15:13 2025 +0300

    Added README

commit fe9904d006e78306b1c23eb37f9433e5de3af3f9
Author: tsganina <ta1.sergeevna@yandex.com>
Date: Sat Feb 15 14:41:10 2025 +0300

    Added index.html.
```

Рис. 21: Научиться извлекать изменения из удаленного репозитория

```
taisiya@tsganina:~/cloned_hello$ cat README.md
This is the Hello World example from the git tutorial.
taisiya@tsganina:~/cloned_hello$ cd ../hello
taisiya@tsganina:~/hello$ cat README.md
This is the Hello World example from the git tutorial.
taisiya@tsganina:~/hello$ cd ../cloned_hello
taisiya@tsganina:~/cloned_hello$ git merge origin/master
Уже актуально.
taisiya@tsganina:~/cloned_hello$ □
```

Рис. 22: cat README.md

```
taisiya@tsganina:~/cloned_hello$ git branch --track style origin/style
git branch -a
git log --max-count=2
branch 'style' set up to track 'origin/style'.
* master
  style
remotes/origin/HEAD -> origin/master
remotes/origin/master
remotes/origin/style
commit 3a8f38d41447f3178f8572fe33f8e41453f408d9 (HEAD -> master, origin/style,
origin/master, origin/HEAD, style)
Author: tsganina <ta1.sergeevna@yandex.com>
Date: Sat Feb 15 15:11:56 2025 +0300

    Updated index.html

commit d1cdb5c5b66cdalbb8b2ed8c6b02cb7fbf33c9cf
Author: tsganina <ta1.sergeevna@yandex.com>
Date: Sat Feb 15 15:10:01 2025 +0300

    Hello uses style.css
taisiya@tsganina:~/cloned_hello$
```

Рис. 23: Добавим локальную ветку, которая отслеживает удаленную ветку

```
taisiya@tsganina:~/hello$ git checkout master
git add README.md
git commit -m "Added shared comment to readme"
M    README.md
Уже на «master»
[master 5c399d3] Added shared comment to readme
 1 file changed, 1 insertion(+)
taisiya@tsganina:~/hello$ git push shared master
Перечисление объектов: 5, готово.
Подсчет объектов: 100% (5/5), готово.
При сжатии изменений используется до 4 потоков
Сжатие объектов: 100% (3/3), готово.
Запись объектов: 100% (3/3), 405 байтов | 405.00 КиБ/с, готово.
Всего 3 (изменений 0), повторно использовано 0 (изменений 0), повторно использо-
вано пакетов 0
To ../hello.git
   3a8f38d..5c399d3  master -> master
taisiya@tsganina:~/hello$
```

Рис. 24: Отправка изменений


```
taisiya@tsganina:~/hello$ cd ../cloned_hello
taisiya@tsganina:~/cloned_hello$ git remote add shared ../hello.git
git branch --track shared master
git pull shared master
cat README.md
branch 'shared' set up to track 'master'.
remote: Перечисление объектов: 5, готово.
remote: Подсчет объектов: 100% (5/5), готово.
remote: Сжатие объектов: 100% (3/3), готово.
remote: Всего 3 (изменений 0), повторно использовано 0 (изменений 0), повторно
использовано пакетов 0
Распаковка объектов: 100% (3/3), 385 байтов | 385.00 Киб/с, готово.
Из ../hello
* branch          master    -> FETCH_HEAD
* [новая ветка]    master    -> shared/master
Обновление 3a8f38d..5c399d3
Fast-forward
 README.md | 1 +
 1 file changed, 1 insertion(+)
This is the Hello World example from the git tutorial.
(Changed in the original and pushed to shared)
taisiya@tsganina:~/cloned_hello$
```

Рис. 25: Извлечение общих изменений

Написание отчета по лабораторной
работе в формате Markdown.

```
73 # Задание
74
75 1. Подготовить пространство для работы.
76 2. Создать проект.
77 3. Внести изменения в проект.
78 4. Провести индексацию изменений
79 5. Работа с тегами версий.
80 6. Отменить локальные изменения (до индексации).
81 7. Отменить проиндексированные изменения (перед коммитом).
82 8. Отменить коммит.
83 9. Удалить коммиты из ветки.
84 10. Удаление тега.
85 11. Внести изменения в коммиты.
86 12. Переместить файл.
87 13. Рассмотреть git внутри: Каталог .git.
88 14. Работа непосредственно с объектами git.
89 15. Навигация по веткам. Изменения в ветке master.
90 16. Слияние веток. Создание конфликта.
91 17. Разрешение конфликтов.
92 18. Сброс ветки style. Сброс ветки master.
93 19. Перебазирование.
94 20. Слияние в ветку master.
95 21. Клонирование репозитория. Просмотр клонированного репозитория.
96 22. Что такое origin? Удаление ветки. Изменение оригинального репозитория.
97 23. Слияние извлеченных изменений.
98 24. Извлечение общих изменений.
99 25. Написание отчета по лабораторной работе в формате Markdown.
```

Рис. 26: Заголовок и нумерованный список

```
106
107 Существует три типа систем контроля версий:
108 |
109 - Локальные системы контроля версий
110 - Централизованные системы контроля версий
111 - Распределенные системы контроля версий
112
```

Рис. 27: Маркированный список

разных этапах, чтобы и они сами, и их коллеги могли их увидеть позже.

106

107 Существует три типа систем контроля версий: – Локальные системы контроля версий –
Централизованные системы контроля версий – Распределенные системы контроля версий

108

109 ****Что такое локальная система контроля версий (LVCS)?****

110

111 Этот тип системы контроля версий очень распространен и прост в использовании. Однако этот
метод может выдавать ошибки и подвержен атакам, потому что файлы хранятся в вашей
локальной системе.

112

113 Это означает, что вы можете потерять системный файл или случайно забыть каталог, с
которым вы работаете (и затем записать в другой каталог). Что такое централизованная
система контроля версий (CVCS)?

114

115 В этом типе контроля версий сервер работает как общее хранилище, в котором находятся все
версии кода. CVCS помогает разработчикам работать совместно.

116

117 Однако, несмотря на то, что такой метод позволяет разработчикам сотрудничать, если сервер
отключится на несколько секунд или будет поврежден, то есть шанс, что вы потеряете все
файлы. Это является серьезной проблемой при работе с CVCS.

118

119 В CVCS только несколько разработчиков могут работать совместно над проектом. Что такое
распределенная система контроля версий (DVCS)?

120

121 В настоящее время это новый и наиболее часто используемый тип системы контроля версий.

122

123 В DVCS у каждого разработчика есть полная резервная копия всех данных на сервере. Это

Рис. 28: Жирный текст

```

179
180 Это свойство GitHub очень важно. GitHub -- это веб-интерфейс не только
181 для разработчиков. Его также могут использовать дизайнеры, писатели и
182 все, кто хочет отслеживать историю своих проектов.
183
184 # Выполнение лабораторной работы
185
186 1. Подготовить пространство для работы.
187
188 Я уже ранее использовала гит, поэтому сейчас выполнила несколько команд
189 для того, чтобы вывести на экран имя и электронную почту, а после
190 поработала с параметрами установки окончаний строк и с установкой
191 отображения unicode. (рис. @fig:001).
192
193 ![Подготовить пространство для работы](image/1.png){#fig:001 width=70%}
194
195 2. Создание проекта.
196
197 Я начала работу в пустом рабочем каталоге с создания пустого каталога с
198 именем hello, затем вошла в него и создала там файл с именем hello.html.
199
200 Потом я создала git репозиторий из этого каталога. Добавила файл в
201 репозиторий, проверила его состояние. (рис. @fig:002,
202 @fig:003, @fig:004, @fig:005).
203
204 ![Создание проекта](image/2.png){#fig:002 width=70%}
205
206 ![Создать git репозиторий из этого каталога](image/3.png){#fig:003 width=70%}
207

```

Рис. 29: Работа с изображениями

```
343
344 13. Рассмотрим Git внутри: Каталог .git. Посмотрим каталог, в котором
345 хранится вся информация git.
346
347 Выполним:
348
349 ```
350 ls -C .git/objects
351 ```
352
353 Мы должны увидеть набор каталогов, имена которых состоят из 2 символов.
354 Имена каталогов являются первыми двумя буквами hexa sha1 объекта,
355 хранящегося в git. После углубимся в базу данных объектов, посмотрим
356 файл конфигурации, создающийся для каждого конкретного проекта.
357
358 Выполним:
359
360 ```
361 ls .git/refs
362 ls .git/refs/heads
363 ls .git/refs/tags
364 cat .git/refs/tags/v1
365 ```
366
367 Мы узнаём файлы в подкаталоге тегов. Каждый файл соответствует тегу,
368 ранее созданному с помощью команды git tag. Его содержание --- это всего
369 лишь hexa коммита, привязанный к тегу. Посмотрим файл HEAD содержит
370 ссылку на текущую ветку, в данный момент это должна быть ветка master.
371 (рис. @fig:033, @fig:034, @fig:035).
372
```

Рис. 30: Вставка кода

```
506
507 # 5 Выводы
508
509 В ходе данной работы я приобрела практические навыки работы с системой
510 контроля версий git. Работа с репозиториями, внесение изменений,
511 индексация, коммиты.
512
513 # Список литературы
514
515 1. [Руководство к лабораторной работе по git] (https://esystem.rudn.ru/pluginfile.php/2584275/mod\_folder/content/0/git.pdf)
516 2. [Руководство к лабораторной работе по Markdown] (https://esystem.rudn.ru/pluginfile.php/2584275/mod\_folder/content/0/markdown.pdf)
```

Рис. 31: Ссылки

Результаты

В ходе данной работы я приобрела практические навыки работы с системой контроля версий git. Работа с репозиториями, внесение изменений, индексация, коммиты.