

Лабораторная работа № 1

Работа с git

Замбалова Дина Владимировна

Содержание

Цель работы	3
Теоретическое введение	3
Выполнение лабораторной работы.....	3
Подготовка.....	3
Создание проекта	4
Внесение изменений	4
История	6
Получение старых версий	6
Создание тегов версий	7
Отмена локальных изменений (до индексации)	8
Отмена проиндексированных изменений (перед коммитом)	8
Отмена коммитов	9
Удаление коммитов из ветки.....	10
Удаление тега oops	11
Изменение предыдущего коммита.....	11
Перемещение файлов.....	12
Подробнее о структуре.....	12
Git внутри: Каталог .git	13
Работа непосредственно с объектами git	14
Создание ветки	15
Навигация по веткам	15
Изменения в ветке master	16
Слияние	17
Создание конфликта	18
Разрешение конфликтов	18
Сброс ветки style.....	19
Сброс ветки master	20
Перебазирование	21

Слияние в ветку master	22
Клонирование репозитория	22
Что такое origin?	23
Удаленные ветки	23
Изменение оригинального репозитория	23
Слияние извлеченных изменений	24
Добавление ветки наблюдения	25
Создание чистого репозитория	25
Отправка и извлечение изменений	25
Выводы	26
Список литературы	26

Список иллюстраций

Настройка git	4
Создание репозитория	4
Добавление файла в репозиторий	4
Внесение изменений в содержимое репозитория	5
Внесение нескольких изменений в содержимое репозитория	5
Внесение нескольких изменений в содержимое репозитория	6
Просмотр истории	6
Просмотр разных версий репозитория	7
Создание тегов версий	7
Переключение по имени тега и просмотр доступных тегов	8
Отмена локальных изменений (до индексации)	8
Отмена проиндексированных изменений (перед коммитом)	9
Отмена коммитов	10
Удаление коммитов из ветки	10
Ничего никогда не теряется	11
Удаление тега oops	11
Изменение предыдущего коммита	12
Изменение предыдущего коммита	12
Результат открытия index.html	13
Каталог .git	14
Работа непосредственно с объектами git	14
Создание ветки	15
Создание ветки	15
Редактирование файла	15
Редактирование файла	15
Просмотр логов новой ветки	16
Переключение на ветку master	16

Возвращение к ветке style	16
Изменения в ветке master.....	17
Слияние веток	17
Создание конфликта	18
Конфликт	18
Разрешение конфликта	19
Поиск коммита перед слиянием	19
Сброс ветки style	20
Поиск коммита перед конфликтом	20
Сброс ветки master.....	21
Перебазирование.....	21
Слияние style в master	22
Клонирование репозитория.....	22
Клонирование репозитория.....	23
Просмотр имени по умолчанию удаленного репозитория	23
Изменение оригинального репозитория	24
Извлечение изменений	24
Слияние извлеченных изменений	25
Добавление ветки наблюдения	25
Создание чистого репозитория.....	25
Отправка изменений	26
Извлечение изменений	26

Цель работы

Приобрести практические навыки работы с системой управления версиями Git.

Теоретическое введение

Git — распределённая система управления версиями. Проект был создан Линусом Торвальдсом для управления разработкой ядра Linux, первая версия выпущена 7 апреля 2005 года; координатор — Дзюн Хамано [@wiki:bash].

Выполнение лабораторной работы

Подготовка

Сначала настроим `core.autocrlf` с параметрами `true` и `input`, чтобы сделать все переводы строк текстовых файлов в главном репозитории одинаковыми, а затем настроим отображение `unicode`(рис. @fig:001).

```
zamba@LAPTOP-1M90JOT7 MINGW64 ~  
$ git config --global core.autocrlf true  
  
zamba@LAPTOP-1M90JOT7 MINGW64 ~  
$ git config --global core.safecrlf true  
  
zamba@LAPTOP-1M90JOT7 MINGW64 ~  
$ git config --global core.quotePath off
```

Настройка git

Создание проекта

Создадим пустой каталог hello, а в нём файл с именем hello.html. Затем создадим git репозиторий из этого каталога, выполнив команду git init. Добавим файл в репозиторий и проверим статус, который сообщает, что коммитить нечего(рис. @fig:002@fig:003).

```
zamba@LAPTOP-1M90JOT7 MINGW64 ~  
$ mkdir hello  
  
zamba@LAPTOP-1M90JOT7 MINGW64 ~  
$ cd hello  
  
zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello  
$ touch hello.html  
  
zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello  
$ echo "Hello, World!" > hello.html  
  
zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello  
$ git init  
Initialized empty Git repository in C:/Users/zamba/hello/.git/
```

Создание репозитория

```
zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)  
$ git add hello.html  
  
zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)  
$ git commit -m "Initial Commit"  
[master (root-commit) cf7dcdf] Initial Commit  
1 file changed, 1 insertion(+)  
create mode 100644 hello.html  
  
zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)  
$ git status  
On branch master  
nothing to commit, working tree clean
```

Добавление файла в репозиторий

Внесение изменений

Изменим содержимое файла hello.html на:

```
<h1>Hello, World!</h1>
```

Проверив состояние рабочего каталога увидим, что git знает, что файл hello.html был изменен, но при этом эти изменения еще не зафиксированы в репозитории. Теперь проиндексируем изменения и снова посмотрим статус, в нём указано, что изменения пока не записаны в репозиторий. И наконец закоммитим изменения, внеся их в репозиторий и снова посмотрим статус, который теперь показывает, что все изменения внесены в репозиторий (рис. @fig:004).

```

zamba@LAPTOP-1M90J0T7 MINGW64 ~/hello (master)
$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:   hello.html

no changes added to commit (use "git add" and/or "git commit -a")

zamba@LAPTOP-1M90J0T7 MINGW64 ~/hello (master)
$ git add hello.html

zamba@LAPTOP-1M90J0T7 MINGW64 ~/hello (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   hello.html

zamba@LAPTOP-1M90J0T7 MINGW64 ~/hello (master)
$ git commit
hint: Waiting for your editor to close the file...
[master 5bf9aea] «Added h1 tag»
1 file changed, 1 insertion(+), 1 deletion(-)

zamba@LAPTOP-1M90J0T7 MINGW64 ~/hello (master)
$ git status
On branch master
nothing to commit, working tree clean

```

Внесение изменений в содержимое репозитория

Изменим страницу «Hello, World», чтобы она содержала стандартные теги

и

.

```

<html>
  <body>
    <h1>Hello, World!</h1>
  </body>
</html>

```

Теперь добавим это изменение в индекс git и добавим заголовки HTML (секцию

) к странице «Hello, World». Проверив текущий статус увидим, что hello.html указан дважды в состоянии. Первое изменение (добавление стандартных тегов) проиндексировано и готово к коммиту. Второе изменение (добавление заголовков HTML) является непроиндексированным. Произведем коммит проиндексированного изменения, затем проиндексируем оставшееся изменение, посмотрим статус и прокоммитим его(рис. @fig:005@fig:006).

```

zamba@LAPTOP-1M90J0T7 MINGW64 ~/hello (master)
$ git add hello.html

zamba@LAPTOP-1M90J0T7 MINGW64 ~/hello (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   hello.html

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:   hello.html

```

Внесение нескольких изменений в содержимое репозитория

```

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ git commit -m "Added standard HTML page tags"
[master 37474f7] Added standard HTML page tags
1 file changed, 5 insertions(+), 1 deletion(-)

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   hello.html

no changes added to commit (use "git add" and/or "git commit -a")

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ git add .

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   hello.html

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ git commit -m "Added HTML header"
[master 7cb8daf] Added HTML header
1 file changed, 2 insertions(+)

```

Внесение нескольких изменений в содержимое репозитория

История

Получим список произведённых изменений в стандартном виде, затем в однострочном, а также с указанием времени и количества (рис. @fig:007).

```

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ git log
commit 7cb8dafa36c825a849386adbc65b250b7a6efe71 (HEAD -> master)
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 19:57:47 2025 +0300

    Added HTML header

commit 37474f7875484b8a4288cb95c7ee9eca00b2e363
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 19:56:45 2025 +0300

    Added standard HTML page tags

commit 5bf9aeab72c67c321b9b75ce212cf8ebaa576c85
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 19:49:26 2025 +0300

    «Added h1 tag»

commit cf7dcdfff6ab8ddf94daf3d55063b0aec1def931
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 19:44:39 2025 +0300

    Initial Commit

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ git log --pretty=oneline
7cb8dafa36c825a849386adbc65b250b7a6efe71 (HEAD -> master) Added HTML header
37474f7875484b8a4288cb95c7ee9eca00b2e363 Added standard HTML page tags
5bf9aeab72c67c321b9b75ce212cf8ebaa576c85 «Added h1 tag»
cf7dcdfff6ab8ddf94daf3d55063b0aec1def931 Initial Commit

```

Просмотр истории

Получение старых версий

Изучим данные лога и найдем там хэш первого коммита, используя его вернемся к первой версии и посмотрим файл hello.html, действительно, увидим первую версию. Затем вернемся к последней версии в ветке master и вновь посмотрим на файл (рис. @fig:008).

```

zamba@LAPTOP-1M90J0T7 MINGW64 ~/hello (master)
$ git checkout cf7dcdf
Note: switching to 'cf7dcdf'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by switching back to a branch.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -c with the switch command. Example:

    git switch -c <new-branch-name>

Or undo this operation with:

    git switch -

Turn off this advice by setting config variable advice.detachedHead to false

HEAD is now at cf7dcdf Initial Commit

zamba@LAPTOP-1M90J0T7 MINGW64 ~/hello ((cf7dcdf...))
$ cat hello.html
Hello, world!

zamba@LAPTOP-1M90J0T7 MINGW64 ~/hello ((cf7dcdf...))
$ git checkout master
Previous HEAD position was cf7dcdf Initial Commit
Switched to branch 'master'

zamba@LAPTOP-1M90J0T7 MINGW64 ~/hello (master)
$ cat hello.html
<html>
  <head>
  </head>
  <body>
    <h1>Hello, world!</h1>
  </body>
</html>

```

Просмотр разных версий репозитория

Создание тегов версий

Назовем текущую версию страницы hello первой (v1). Создадим тег первой версии и используем его для того чтобы вернуться к предыдущей, которой также присвоим тег(рис. @fig:009).

```

zamba@LAPTOP-1M90J0T7 MINGW64 ~/hello (master)
$ git tag v1

zamba@LAPTOP-1M90J0T7 MINGW64 ~/hello (master)
$ git checkout v1^
Note: switching to 'v1^'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by switching back to a branch.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -c with the switch command. Example:

    git switch -c <new-branch-name>

Or undo this operation with:

    git switch -

Turn off this advice by setting config variable advice.detachedHead to false

HEAD is now at 37474F7 Added standard HTML page tags

zamba@LAPTOP-1M90J0T7 MINGW64 ~/hello ((37474F7...))
$ cat hello.html
  <body>
    <h1>Hello, world!</h1>
  </body>
</html>

zamba@LAPTOP-1M90J0T7 MINGW64 ~/hello ((37474F7...))
$ git tag v1-beta

```

Создание тегов версий

Переключимся по тегам между двумя отмеченными версиями. Просмотрим все доступные теги(их два) и посмотрим теги в логе(рис. @fig:010).

```

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello ((v1-beta))
$ git tag
v1
v1-beta

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello ((v1-beta))
$ git log master --all
commit 7cb8d4fe36c825a849386adbc65b250b7a6efe71 (tag: v1, master)
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 19:57:47 2025 +0300

    Added HTML header

commit 37474f7875484b8a4288cb95c7ee9eca00b2e363 (HEAD, tag: v1-beta)
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 19:56:45 2025 +0300

    Added standard HTML page tags

commit 5bf9aeab72c67c321b9b75ce212cf8ebaa576c85
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 19:49:26 2025 +0300

    «Added h1 tag»

commit cf7dcdfff6ab8ddf94daf3d55063b0aec1def931
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 19:44:39 2025 +0300

    Initial Commit

```

Переключение по имени тега и просмотр доступных тегов

Отмена локальных изменений (до индексации)

Убдеимся, что мы находимся на последнем коммите ветки master и внесем изменение в файл hello.html в виде нежелательного комментария. Затем проверим статус, увидим, что изменения ещё не проиндексированы. Используем команду git checkout для переключения версии файла hello.html в репозитории(рис. @fig:011).

```

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello ((v1-beta))
$ git checkout master
Previous HEAD position was 37474f7 Added standard HTML page tags
Switched to branch 'master'

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   hello.html

no changes added to commit (use "git add" and/or "git commit -a")

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ git checkout hello.html
Updated 1 path from the index

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ git status
On branch master
nothing to commit, working tree clean

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ cat hello.html
<html>
  <head>
  </head>
  <body>
    <h1>Hello, world!</h1>
  </body>
</html>

```

Отмена локальных изменений (до индексации)

Отмена проиндексированных изменений (перед коммитом)

Внесем изменение в файл hello.html в виде нежелательного комментария

```

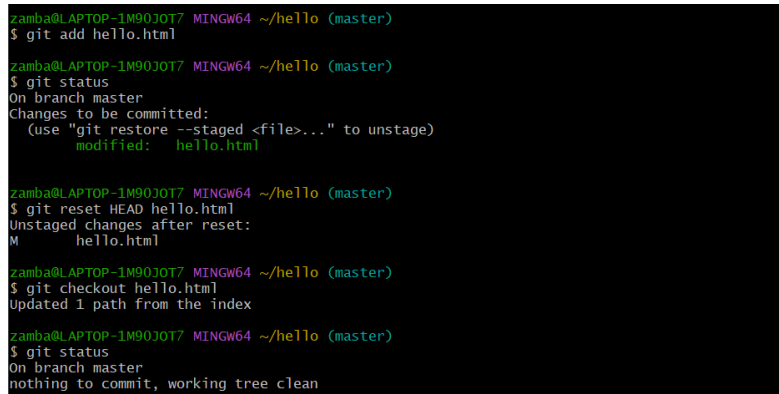
<html>
  <head>
    <!-- This is an unwanted but staged comment -->
  </head>
  <body>
    <h1>Hello, World!</h1>

```



```
</body>
</html>
```

Проиндексируем это изменение и проверим состояние. Состояние показывает, что изменение было проиндексировано и готово к коммиту. Используем команду `git reset`, чтобы сбросить буферную зону к HEAD. Это очищает буферную зону от изменений, которые мы только что проиндексировали. И переключимся на последнюю версию коммита, посмотрев статус увидим, что наш каталог опять чист(рис. @fig:012).



```
zamba@LAPTOP-1M90J0T7 MINGW64 ~/hello (master)
$ git add hello.html

zamba@LAPTOP-1M90J0T7 MINGW64 ~/hello (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   hello.html

zamba@LAPTOP-1M90J0T7 MINGW64 ~/hello (master)
$ git reset HEAD hello.html
Unstaged changes after reset:
M       hello.html

zamba@LAPTOP-1M90J0T7 MINGW64 ~/hello (master)
$ git checkout hello.html
Updated 1 path from the index

zamba@LAPTOP-1M90J0T7 MINGW64 ~/hello (master)
$ git status
On branch master
nothing to commit, working tree clean
```

Отмена проиндексированных изменений (перед коммитом)

Отмена коммитов

Изменим файл `hello.html` на следующий.

```
<html>
  <head>
  </head>
  <body>
    <h1>Hello, World!</h1>
    <!-- This is an unwanted but committed change -->
  </body>
</html>
```

Проиндексируем изменения файла и прокоммитим их. Чтобы отменить коммит, нам необходимо сделать коммит, который удаляет изменения, сохраненные нежелательным коммитом. Перейдем в редактор, где изменим нежелательный коммит. Проверим лог. Проверка лога показывает нежелательные и отмененные коммиты в наш репозиторий(рис. @fig:013).

```

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ git add hello.html

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ git commit -m "Oops, we didn't want this commit"
[master 0e2e7c9] Oops, we didn't want this commit
1 file changed, 1 insertion(+)

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ git revert HEAD
[master cc974f9] Revert "Oops, we didn't want this commit"
1 file changed, 1 deletion(-)

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ git log
commit cc974f93684dedc301b531e9c72fc96377e8809 (HEAD -> master)
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 20:12:18 2025 +0300

    Revert "Oops, we didn't want this commit"

    This reverts commit 0e2e7c9c36d596a46e68fc3bfab145341fcbe374.

commit 0e2e7c9c36d596a46e68fc3bfab145341fcbe374
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 20:12:01 2025 +0300

    Oops, we didn't want this commit

```

Отмена коммитов

Удаление коммитов из ветки

Удалим последние два коммита с помощью сброса, сначала отметим последний коммит тегом, чтобы его можно было потом найти. Используем команду `git reset`, чтобы вернуться к версии до этих коммитов. Теперь в логе их нет, но если посмотреть логи с опцией `--all` можно всё ещё их увидеть, но метка HEAD находится на нужной нам версии (рис. @fig:014@fig:015).

```

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ git tag oops

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ git reset --hard v1
HEAD is now at 7cb8daf Added HTML header

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ git log
commit 7cb8daf36c825a849386adbc65b250b7a6efe71 (HEAD -> master, tag: v1)
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 19:57:47 2025 +0300

    Added HTML header

commit 37474f7875484b8a4288cb95c7ee9eca00b2e363 (tag: v1-beta)
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 19:56:45 2025 +0300

    Added standard HTML page tags

commit 5bf9aeab72c67c321b9b75ce212cf8ebaa576c85
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 19:49:26 2025 +0300

    «Added h1 tag»

commit cf7dcdf66ab8ddf94daf3d55063b0aec1def931
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 19:44:39 2025 +0300

    Initial Commit

```

Удаление коммитов из ветки

```

zamba@LAPTOP-1M90J0T7 MINGW64 ~/hello (master)
$ git log --all
commit cc974f93684decd301b531e9c72fc96377e8809 (tag: oops)
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 20:12:18 2025 +0300

    Revert "Oops, we didn't want this commit"

    This reverts commit 0e2e7c9c36d596a46e68fc3bfab145341fcbe374.

commit 0e2e7c9c36d596a46e68fc3bfab145341fcbe374
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 20:12:01 2025 +0300

    Oops, we didn't want this commit

commit 7cb8dafa36c825a849386adb65b250b7a6efe71 (HEAD -> master, tag: v1)
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 19:57:47 2025 +0300

    Added HTML header

commit 37474f7875484b8a4288cb95c7ee9eca00b2e363 (tag: v1-beta)
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 19:56:45 2025 +0300

    Added standard HTML page tags

commit 5bf9aeab72c67c321b9b75ce212cf8ebaa576c85
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 19:49:26 2025 +0300

    «Added h1 tag»

commit cf7dcdfff6ab8ddf94daf3d55063b0aec1def931
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 19:44:39 2025 +0300

    Initial Commit

```

Ничего никогда не теряется

Удаление тега оорс

Удалим тег оорс и коммиты, на которые он ссылался, сборщиком мусора. Теперь этот тег не отображается в репозитории(рис. @fig:016).

```

zamba@LAPTOP-1M90J0T7 MINGW64 ~/hello (master)
$ git tag -d oops
Deleted tag 'oops' (was cc974f9)

zamba@LAPTOP-1M90J0T7 MINGW64 ~/hello (master)
$ git log --all
commit 7cb8dafa36c825a849386adb65b250b7a6efe71 (HEAD -> master, tag: v1)
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 19:57:47 2025 +0300

    Added HTML header

commit 37474f7875484b8a4288cb95c7ee9eca00b2e363 (tag: v1-beta)
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 19:56:45 2025 +0300

    Added standard HTML page tags

commit 5bf9aeab72c67c321b9b75ce212cf8ebaa576c85
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 19:49:26 2025 +0300

    «Added h1 tag»

commit cf7dcdfff6ab8ddf94daf3d55063b0aec1def931
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 19:44:39 2025 +0300

    Initial Commit

```

Удаление тега оорс

Изменение предыдущего коммита

Добавим в страницу комментариев автора (рис. @fig:017).

```
<> hello.html ●
C: > Users > zamba > hello > <> hello.html > ...
1  <!-- Author: Dina V. Zambalova -->
2  <html>
3      <head>
4      </head>
5      <body>
6          <h1>Hello, World!</h1>
7      </body>
8  </html>
```

Изменение предыдущего коммита

Затем добавим их в репозиторий. Теперь мы хотим добавить в комментарий автора почту, обнови страницу hello, включив в неё почту. Чтобы у нас остался один коммит, а не два, изменим последний с помощью опции `--amend`, теперь в логах отображается последняя версия коммита (рис. @fig:018).

```
zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ git add hello.html

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ git commit -m "Add an author comment"
[master c16671a] Add an author comment
1 file changed, 1 insertion(+)

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ git add hello.html

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ git commit -m "Add an author and email comment"
[master 5e89eb6] Add an author and email comment
1 file changed, 1 insertion(+), 1 deletion(-)

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ git reset

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ git commit --amend -m "Add an author/email comment"
[master 3cb49d8] Add an author/email comment
Date: Sat Feb 15 20:19:39 2025 +0300
1 file changed, 1 insertion(+), 1 deletion(-)

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ git log
commit 3cb49d8dac1ed4200b97cc672056195fa68b3999 (HEAD -> master)
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 20:19:39 2025 +0300

    Add an author/email comment

commit c16671a145afe42c3542ffca3f16b2ddff7343d
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 20:17:53 2025 +0300

    Add an author comment
```

Изменение предыдущего коммита

Перемещение файлов

Переместим наш файл в каталог lib. Для этого создадим его и используем команду `git mv`, сделаем коммит этого перемещения.

Подробнее о структуре

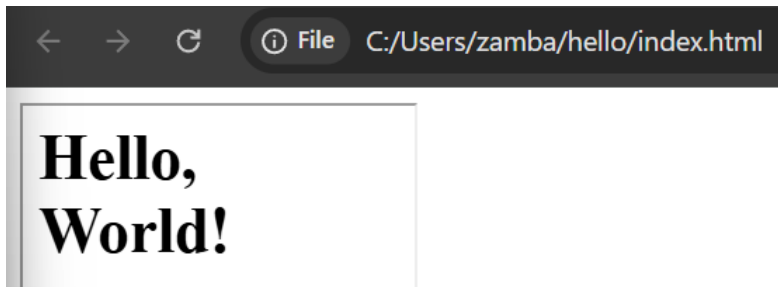
Добавим файл `index.html` в наш репозиторий

```
<html>
  <body>
```

```
<iframe src="lib/hello.html" width="200" height="200" />
</body>
</html>
```

Добавим файл и сделаем коммит.

Теперь при открытии index.html, увидим кусок страницы hello в маленьком окошке(рис.@fig:019).



Результат открытия index.html

Git внутри: Каталог .git

Посмотрим каталог, в котором хранится вся информация git. Затем посмотрим набор каталогов, имена которых состоят из 2 символов. Имена каталогов являются первыми двумя буквами хэша sha1 объекта, хранящегося в git. Посмотрим в один из каталогов с именем из 2 букв. Увидим файлы с именами из 38 символов. Это файлы, содержащие объекты, хранящиеся в git. Посмотрим файл конфигурации, создающийся для каждого конкретного проекта. Затем посмотрим подкаталоги .git/refs/heads и .git/refs/tags, а также содержимое файла v1, в нём хранится хэш коммита, привязанный к тегу. Также посмотрим содержимое файла HEAD, который содержит ссылку на текущую ветку, в данный момент это ветка master(рис. @fig:020).

```

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ ls -C .git
COMMIT_EDITMSG  ORIG_HEAD  description  index  logs/  refs/
HEAD           config     hooks/      info/  objects/

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ ls -C .git/objects
02/ 0e/ 31/ 3c/ 52/ 5a/ 5e/ 6c/ 7c/ 8a/ a6/ bf/ cc/ f7/ pack/
09/ 1f/ 37/ 4b/ 59/ 5b/ 62/ 78/ 84/ 8b/ ae/ c1/ cf/ info/

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ ls -C .git/objects/<dir>
bash: syntax error near unexpected token `newline'

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ ls -C .git/objects/cf
7dcdf6ab8ddf94daf3d55063b0aec1def931

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ cat .git/config
[core]
    repositoryformatversion = 0
    filemode = false
    bare = false
    logallrefupdates = true
    symlinks = false
    ignorecase = true
    autocrlf = input

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ ls .git/refs
heads/  tags/

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ ls .git/refs/heads
master

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ ls .git/refs/tags
v1  v1-beta

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ cat .git/refs/tags/v1
7cb8dafa36c825a849386adbc65b250b7a6efe71

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ cat .git/HEAD
ref: refs/heads/master

```

Каталог .git

Работа непосредственно с объектами git

Найдем последний коммит и выведем его с помощью SHA1 хэша. Затем посмотрим дерево каталогов, ссылка на который идёт в последнем коммите, выведем каталог lib и файл hello.html(рис. @fig:019).

```

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ git log --max-count=1
commit 84cb721c7c96dd409b1a1941fc4ce6b6482712e5 (HEAD -> master)
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 20:26:12 2025 +0300

    Moved hello.html to lib

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ git cat-file -t 84cb721
commit

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ git cat-file -p 84cb721
tree 1f6b6f16f2309b92848ffa72267b7a81b52a7b48
parent 3cb49d8dac1ed4200b97cc672056195fa68b3999
author dina <1132226536@pfur.ru> 1739640372 +0300
committer dina <1132226536@pfur.ru> 1739640372 +0300

Moved hello.html to lib

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ git cat-file -p 1f6b6f1
040000 tree 31b4412bbf8f56ba5c352a027b8650ea3d5191b4    lib

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ git cat-file -p 31b4412
100644 blob 02375877b79a7caad70a95b704ba824976431759    hello.html

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ git cat-file -p 0237587
<!-- Author: Dina V. Zambalova (1132226536@pfur.ru) -->
<html>
  <head>
  </head>
  <body>
    <h1>Hello, world!</h1>
  </body>
</html>

```

Работа непосредственно с объектами git

Исследуем git репозиторий вручную самостоятельно. Используя хэш родительского коммита последовательно дойдем до первой версии файла hello.html и посмотрим его.

Создание ветки

Создадим новую ветку «style» и перейдем в неё. Добавим туда файл стилей style.css и добавим его в репозиторий. Обновим файл hello.html, чтобы использовать стили style.css и index.html, также обновим их в репозиторий(рис. @fig:022@fig:025).

```
zamba@LAPTOP-1M90J0T7 MINGW64 ~/hello (master)
$ git checkout -b style
Switched to a new branch 'style'

zamba@LAPTOP-1M90J0T7 MINGW64 ~/hello (style)
$ git status
On branch style
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    index.html

nothing added to commit but untracked files present (use "git add" to track)

zamba@LAPTOP-1M90J0T7 MINGW64 ~/hello (style)
$ touch lib/style.css
```

Создание ветки

```
zamba@LAPTOP-1M90J0T7 MINGW64 ~/hello (style)
$ git add lib/style.css

zamba@LAPTOP-1M90J0T7 MINGW64 ~/hello (style)
$ git commit -m "Added css stylesheet"
[style fe7d490] Added css stylesheet
1 file changed, 4 insertions(+)
create mode 100644 lib/style.css
```

Создание ветки

```
1  <!-- Author: Dina V. Zambalova (1132226536@pfur.ru) -->
2  <html>
3      <head>
4          <link type="text/css" rel="stylesheet"
5      media="all" href="style.css" />
6      </head>
7      <body>
8          <h1>Hello, World!</h1>
9      </body>
10 </html>
```

Редактирование файла

```
1  <html>
2      <head>
3          <link type="text/css" rel="stylesheet"
4          media="all" href="lib/style.css" />
5      </head>
6      <body>
7          <iframe src="lib/hello.html" width="200" height="200" />
8      </body>
9  </html>
```

Редактирование файла

Навигация по веткам

Посмотрим все логи(рис. @fig:026).

```

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (style)
$ git log --all
commit 74ed3f16a843022b4df178d9aea637cdfc3b6324 (HEAD -> style)
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 20:57:39 2025 +0300

    Updated index.html

commit b58634dec18f15e63a0aa43b9a47845b585f221b
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 20:54:49 2025 +0300

    Hello uses style.css

commit fe7d490e063675ef8acf4ed553b221084122e9c8
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 20:52:07 2025 +0300

    Added css stylesheet

commit 84cb721c7c96dd409b1a1941fc4ce6b6482712e5 (master)
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 20:26:12 2025 +0300

    Moved hello.html to lib

```

Просмотр логов новой ветки

Переключимся обратно на основную ветку и посмотрим содержимое файла `lib/hello.html`, заметим, что он не использует стили, также посмотрим содержимое этого файла в новой ветке(рис. @fig:027@fig:028).

```

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (style)
$ git checkout master
Switched to branch 'master'

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ cat lib/hello.html
<!-- Author: Dina V. Zambalova (1132226536@pfur.ru) -->
<html>
  <head>
  </head>
  <body>
    <h1>Hello, world!</h1>
  </body>
</html>

```

Переключение на ветку master

```

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ git checkout style
Switched to branch 'style'

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (style)
$ cat lib/hello.html
<!-- Author: Dina V. Zambalova (1132226536@pfur.ru) -->
<html>
  <head>
    <link type="text/css" rel="stylesheet"
media="all" href="style.css" />
  </head>
  <body>
    <h1>Hello, world!</h1>
  </body>
</html>

```

Возвращение к ветке style

Изменения в ветке master

Вернемся в основную ветку и добавим файл `README.md`. Посмотрим ветки и их различия(рис. @fig:029).


```

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (style)
$ git checkout master
Switched to branch 'master'

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ echo "This is the Hello world example from the git tutorial." > README.md

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ git add README.md

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ git commit -m "Added README"
[master b2954c3] Added README
1 file changed, 1 insertion(+)
create mode 100644 README.md

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ git log --graph --all
* commit b2954c34fae578928709eb3c621cded5ae45a380 (HEAD -> master)
  Author: dina <1132226536@pfur.ru>
  Date: Sat Feb 15 21:04:52 2025 +0300

    Added README

* commit 74ed3f16a843022b4df178d9aea637cdfc3b6324 (style)
  Author: dina <1132226536@pfur.ru>
  Date: Sat Feb 15 20:57:39 2025 +0300

    Updated index.html

* commit b58634dec18f15e63a0aa43b9a47845b585f221b
  Author: dina <1132226536@pfur.ru>
  Date: Sat Feb 15 20:54:49 2025 +0300

    Hello uses style.css

* commit fe7d490e063675ef8acf4ed553b221084122e9c8
  Author: dina <1132226536@pfur.ru>
  Date: Sat Feb 15 20:52:07 2025 +0300

    Added css stylesheet

* commit 84cb721c7c96dd409b1a1941fc4ce6b6482712e5
  Author: dina <1132226536@pfur.ru>
  Date: Sat Feb 15 20:26:12 2025 +0300

    Moved hello.html to lib

```

Изменения в ветке master

Слияние

Слияние переносит изменения из двух веток в одну. Вернемся к ветке style и сольем master с style(рис. @fig:030).

```

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ git checkout style
Switched to branch 'style'

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (style)
$ git merge master
Merge made by the 'ort' strategy.
 README.md | 1 +
 1 file changed, 1 insertion(+)
 create mode 100644 README.md

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (style)
$ git log --graph --all
* commit cafceef19ee610acbae233608c99b4a683e75a252 (HEAD -> style)
  \ Merge: 74ed3f1 b2954c3
  | Author: dina <1132226536@pfur.ru>
  | Date: Sat Feb 15 21:06:34 2025 +0300
  |
  | Merge branch 'master' into style
  |
  * commit b2954c34fae578928709eb3c621cded5ae45a380 (master)
    | Author: dina <1132226536@pfur.ru>
    | Date: Sat Feb 15 21:04:52 2025 +0300
    |
    | Added README
    |
  * commit 74ed3f16a843022b4df178d9aea637cdfc3b6324
    | Author: dina <1132226536@pfur.ru>
    | Date: Sat Feb 15 20:57:39 2025 +0300
    |
    | Updated index.html
    |
  * commit b58634dec18f15e63a0aa43b9a47845b585f221b
    | Author: dina <1132226536@pfur.ru>
    | Date: Sat Feb 15 20:54:49 2025 +0300
    |
    | Hello uses style.css
    |
  * commit fe7d490e063675ef8acf4ed553b221084122e9c8
    | Author: dina <1132226536@pfur.ru>
    | Date: Sat Feb 15 20:52:07 2025 +0300
    |
    | Added css stylesheet

```

Слияние веток

Создание конфликта

Вернемся в ветку master и создадим конфликт, внося изменения в файл hello.html. Просмотрим ветки. После коммита «Added README» ветка master была объединена с веткой style, но в настоящее время в master есть дополнительный коммит, который не был слит с style. Последнее изменение в master конфликтует с некоторыми изменениями в style(рис. @fig:031).

```
1 <!-- Author: Dina V. Zambalova (1132226536@pfur.ru) -->
2 <html>
3   <head>
4     <!-- no style -->
5   </head>
6   <body>
7     <h1>Hello, World! Life is great!</h1>
8   </body>
9 </html>
```

Создание конфликта

Разрешение конфликтов

Вернемся к ветке style и попытаемся объединить ее с новой веткой master. В файле lib/hello.html можно увидеть записи с обеих версий этого файла. Первый раздел — версия текущей ветки (style). Второй раздел — версия ветки master. Внесем изменения в lib/hello.html, оставив только необходимую нам запись и добавим этот файл в репозиторий, чтобы вручную разрешить конфликт(рис. @fig:032@fig:033).

```
zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (style)
$ git checkout master
Switched to branch 'master'

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ git add lib/hello.html

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ git commit -m 'Life is great'
[master 3dd0edb] Life is great
1 file changed, 2 insertions(+), 1 deletion(-)

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ git log --graph --all
* commit 3dd0edb77a7b57e00d4b1b65505e3b6b42931693 (HEAD -> master)
  Author: dina <1132226536@pfur.ru>
  Date: Sat Feb 15 21:10:38 2025 +0300

    Life is great

* commit cafcef19ee610acbae233608c99b4a683e75a252 (style)
  Merge: 74ed3f1 b2954c3
  Author: dina <1132226536@pfur.ru>
  Date: Sat Feb 15 21:06:34 2025 +0300

    Merge branch 'master' into style

* commit b2954c34fae578928709eb3c621cded5ae45a380
  Author: dina <1132226536@pfur.ru>
  Date: Sat Feb 15 21:04:52 2025 +0300

    Added README

* commit 74ed3f16a843022b4df178d9aea637cdfc3b6324
  Author: dina <1132226536@pfur.ru>
  Date: Sat Feb 15 20:57:39 2025 +0300

    Updated index.html

* commit b58634dec18f15e63a0aa43b9a47845b585f221b
  Author: dina <1132226536@pfur.ru>
  Date: Sat Feb 15 20:54:49 2025 +0300

    Hello uses style.css
```

Конфликт

```

1  <!-- Author: Dina V. Zambalova (1132226536@pfur.ru) -->
2  <html>
3      <head>
4          <link type="text/css" rel="stylesheet"
5              media="all" href="style.css" />
6      </head>
7      <body>
8          <h1>Hello, World! Life is great!</h1>
9      </body>
10 </html>

```

Разрешение конфликта

Сброс ветки style

Вернемся на ветке style к точке перед тем, как мы слили ее с веткой master. Мы хотим вернуться в ветке style в точку перед слиянием с master. Нам необходимо найти последний коммит перед слиянием(рис. @fig:034).

```

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (style)
$ git log --graph
* commit ccfce8f302853830e3260b72cbfcf59743b7c30f (HEAD -> style)
  Merge: cafceff1 3dd0edb
  Author: dina <1132226536@pfur.ru>
  Date: Sat Feb 15 21:17:36 2025 +0300

    Merged master fixed conflict.

* commit 3dd0edb77a7b57e00d4b1b65505e3b6b42931693 (master)
  Author: dina <1132226536@pfur.ru>
  Date: Sat Feb 15 21:10:38 2025 +0300

    Life is great

* commit cafceff19ee610acbae233608c99b4a683e75a252
  Merge: 74ed3f1 b2954c3
  Author: dina <1132226536@pfur.ru>
  Date: Sat Feb 15 21:06:34 2025 +0300

    Merge branch 'master' into style

* commit b2954c34fae578928709eb3c621cded5ae45a380
  Author: dina <1132226536@pfur.ru>
  Date: Sat Feb 15 21:04:52 2025 +0300

    Added README

* commit 74ed3f16a843022b4df178d9aea637cdfc3b6324
  Author: dina <1132226536@pfur.ru>
  Date: Sat Feb 15 20:57:39 2025 +0300

    Updated index.html

* commit b58634dec18f15e63a0aa43b9a47845b585f221b
  Author: dina <1132226536@pfur.ru>
  Date: Sat Feb 15 20:54:49 2025 +0300

    Hello uses style.css

* commit fe7d490e063675ef8acf4ed553b221084122e9c8
  Author: dina <1132226536@pfur.ru>
  Date: Sat Feb 15 20:52:07 2025 +0300

    Added css stylesheet

* commit 84cb721c7c96dd409b1a1941fc4ce6b6482712e5
  Author: dina <1132226536@pfur.ru>
  Date: Sat Feb 15 20:26:12 2025 +0300

    Moved hello.html to lib

```

Поиск коммита перед слиянием

Мы видим, что коммит «Updated index.html» был последним на ветке style перед слиянием. Сбросим ветку style к этому коммиту.

Поищем лог ветки style. Увидим, что у нас в истории больше нет коммитов слияний(рис. @fig:035).

```

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (style)
$ git reset --hard 74ed3f1
HEAD is now at 74ed3f1 Updated index.html

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (style)
$ git log --graph --all
* commit 3dd0edb77a7b57e00d4b1b65505e3b6b42931693 (master)
  Author: dina <1132226536@pfur.ru>
  Date: Sat Feb 15 21:10:38 2025 +0300

    Life is great

* commit b2954c34fae578928709eb3c621cded5ae45a380
  Author: dina <1132226536@pfur.ru>
  Date: Sat Feb 15 21:04:52 2025 +0300

    Added README

* commit 74ed3f16a843022b4df178d9aea637cdfc3b6324 (HEAD -> style)
  Author: dina <1132226536@pfur.ru>
  Date: Sat Feb 15 20:57:39 2025 +0300

    Updated index.html

* commit b58634dec18f15e63a0aa43b9a47845b585f221b
  Author: dina <1132226536@pfur.ru>
  Date: Sat Feb 15 20:54:49 2025 +0300

    Hello uses style.css

* commit fe7d490e063675ef8acf4ed553b221084122e9c8
  Author: dina <1132226536@pfur.ru>
  Date: Sat Feb 15 20:52:07 2025 +0300

    Added css stylesheet

* commit 84cb721c7c96dd409b1a1941fc4ce6b6482712e5
  Author: dina <1132226536@pfur.ru>
  Date: Sat Feb 15 20:26:12 2025 +0300

    Moved hello.html to lib

```

Сброс ветки style

Сброс ветки master

Добавив интерактивный режим в ветку master, мы внесли изменения, конфликтующие с изменениями в ветке style. Давайте вернемся в ветке master в точку перед внесением конфликтующих изменений. Это позволяет нам продемонстрировать работу команды git rebase, не беспокоясь о конфликтах. Просмотрим коммиты ветки master(рис. @fig:036).

```

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (style)
$ git checkout master
Switched to branch 'master'

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ git log --graph
* commit 3dd0edb77a7b57e00d4b1b65505e3b6b42931693 (HEAD -> master)
  Author: dina <1132226536@pfur.ru>
  Date: Sat Feb 15 21:10:38 2025 +0300

    Life is great

* commit b2954c34fae578928709eb3c621cded5ae45a380
  Author: dina <1132226536@pfur.ru>
  Date: Sat Feb 15 21:04:52 2025 +0300

    Added README

* commit 84cb721c7c96dd409b1a1941fc4ce6b6482712e5
  Author: dina <1132226536@pfur.ru>
  Date: Sat Feb 15 20:26:12 2025 +0300

    Moved hello.html to lib

* commit 3cb49d8dac1ed4200b97cc672056195fa68b3999
  Author: dina <1132226536@pfur.ru>
  Date: Sat Feb 15 20:19:39 2025 +0300

    Add an author/email comment

* commit c16671a145afef42c3542ffca3f16b2ddff7343d
  Author: dina <1132226536@pfur.ru>
  Date: Sat Feb 15 20:17:53 2025 +0300

    Add an author comment

* commit 7cb8dafa36c825a849386adb65b250b7a6efe71 (tag: v1)
  Author: dina <1132226536@pfur.ru>
  Date: Sat Feb 15 19:57:47 2025 +0300

    Added HTML header

```

Поиск коммита перед конфликтом

Коммит «Added README» идет непосредственно перед коммитом конфликтующего интерактивного режима. Мы сбросим ветку master к коммиту «Added README»(рис. @fig:037).

```
zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ git reset --hard b2954c3
HEAD is now at b2954c3 Added README

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ git log --graph --all
* commit b2954c34fae578928709eb3c621cdd5ae45a380 (HEAD -> master)
  Author: dina <113226536@pfur.ru>
  Date: Sat Feb 15 21:04:52 2025 +0300

    Added README

* commit 74ed3f16a843022b4df178d9aea637cdfc3b6324 (style)
  Author: dina <113226536@pfur.ru>
  Date: Sat Feb 15 20:57:39 2025 +0300

    Updated index.html

* commit b58634dec18f15e63a0aa43b9a47845b585f221b
  Author: dina <113226536@pfur.ru>
  Date: Sat Feb 15 20:54:49 2025 +0300

    Hello uses style.css

* commit fe7d490e063675ef8acf4ed553b221084122e9c8
  Author: dina <113226536@pfur.ru>
  Date: Sat Feb 15 20:52:07 2025 +0300

    Added css stylesheet

* commit 84cb721c7c96dd409b1a1941fc4ce6b6482712e5
  Author: dina <113226536@pfur.ru>
  Date: Sat Feb 15 20:26:12 2025 +0300

    Moved hello.html to lib
```

Сброс ветки master

Перебазирование

Используем команду rebase вместо команды merge. Мы вернулись в точку до первого слияния и хотим перенести изменения из ветки master в нашу ветку style. На этот раз для переноса изменений из ветки master мы будем использовать команду git rebase вместо слияния(рис. @fig:038).

```
zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (master)
$ git checkout style
Switched to branch 'style'

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (style)
$ git rebase master
Successfully rebased and updated refs/heads/style.

zamba@LAPTOP-1M90JOT7 MINGW64 ~/hello (style)
$ git log --graph
* commit 77c529194578880085eee3392f99e505bc2520cf (HEAD -> style)
  Author: dina <113226536@pfur.ru>
  Date: Sat Feb 15 20:57:39 2025 +0300

    Updated index.html

* commit 3c8eef665633ac400eecaabdbec253368b5cf87dd
  Author: dina <113226536@pfur.ru>
  Date: Sat Feb 15 20:54:49 2025 +0300

    Hello uses style.css

* commit 51e28697339b4e53911b6516932861a8ac7a7a6a
  Author: dina <113226536@pfur.ru>
  Date: Sat Feb 15 20:52:07 2025 +0300

    Added css stylesheet

* commit b2954c34fae578928709eb3c621cdd5ae45a380 (master)
  Author: dina <113226536@pfur.ru>
  Date: Sat Feb 15 21:04:52 2025 +0300

    Added README

* commit 84cb721c7c96dd409b1a1941fc4ce6b6482712e5
  Author: dina <113226536@pfur.ru>
  Date: Sat Feb 15 20:26:12 2025 +0300

    Moved hello.html to lib
```

Перебазирование

Слияние в ветку master

Вернемся в ветку master и сольем ветку style в неё с помощью команды git merge(рис. @fig:039).

```
zamba@LAPTOP-1M90J0T7 MINGW64 ~/hello (style)
$ git checkout master
Switched to branch 'master'

zamba@LAPTOP-1M90J0T7 MINGW64 ~/hello (master)
$ git merge style
Updating b2954c3..77c5291
Fast-forward
 index.html | 9 +++++++
 lib/hello.html | 2 ++
 lib/style.css | 4 +++
 3 files changed, 15 insertions(+)
 create mode 100644 index.html
 create mode 100644 lib/style.css

zamba@LAPTOP-1M90J0T7 MINGW64 ~/hello (master)
$ git log
commit 77c529194578880085eee3392f99e505bc2520cf (HEAD -> master, style)
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 20:57:39 2025 +0300

    Updated index.html

commit 3c8eef665633ac400eecabdbec253368b5cf87dd
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 20:54:49 2025 +0300

    Hello uses style.css

commit 51e28697339b4e53911b6516932861a8ac7a7a6a
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 20:52:07 2025 +0300

    Added css stylesheet

commit b2954c34fae578928709eb3c621cded5ae45a380
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 21:04:52 2025 +0300

    Added README

commit 84cb721c7c96dd409b1a1941fc4ce6b6482712e5
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 20:26:12 2025 +0300

    Moved hello.html to lib
```

Слияние style в master

Клонирование репозитория

Перейдем в наш рабочий каталог и сделаем клон репозитория hello, затем создадим клон репозитория. Просмотрев его увидим список всех файлов на верхнем уровне оригинального репозитория README.md, index.html и lib. Затем посмотрим историю репозитория и увидим список всех коммитов в новый репозиторий, и он совпадает с историей коммитов в оригинальном репозитории. Единствен в названиях веток(рис. @fig:040@fig:041).

```
zamba@LAPTOP-1M90J0T7 MINGW64 ~
$ cd work

zamba@LAPTOP-1M90J0T7 MINGW64 ~/work
$ pwd
/c/Users/zamba/work

zamba@LAPTOP-1M90J0T7 MINGW64 ~/work
$ ls
hello/

zamba@LAPTOP-1M90J0T7 MINGW64 ~/work
$ git clone hello cloned_hello
Cloning into 'cloned_hello'...
done.

zamba@LAPTOP-1M90J0T7 MINGW64 ~/work
$ ls
cloned_hello/  hello/
```

Клонирование репозитория

```

zamba@LAPTOP-1M90JOT7 MINGW64 ~/work
$ cd cloned_hello

zamba@LAPTOP-1M90JOT7 MINGW64 ~/work/cloned_hello (master)
$ ls
README.md  index.html  lib/

zamba@LAPTOP-1M90JOT7 MINGW64 ~/work/cloned_hello (master)
$ git log --all
commit 77c529194578880085eee3392f99e505bc2520cf (HEAD -> master, origin/master, origin/HEAD)
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 20:57:39 2025 +0300

    Updated index.html

commit 3c8eef665633ac400eacabdbec253368b5cf87dd
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 20:54:49 2025 +0300

    Hello uses style.css

commit 51e28697339b4e53911b6516932861a8ac7a7a6a
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 20:52:07 2025 +0300

    Added css stylesheet

commit b2954c34fae578928709eb3c621cded5ae45a380
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 21:04:52 2025 +0300

    Added README

commit 84cb721c7c96dd409b1a1941fc4ce6b6482712e5
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 20:26:12 2025 +0300

    Moved hello.html to lib

```

Клонирование репозитория

Что такое origin?

Клонированный репозиторий знает об имени по умолчанию удаленного репозитория. Посмотрим, подробную информацию об имени по умолчанию. Для того, чтобы увидеть все ветки используем опцию -a (рис. @fig:042).

```

zamba@LAPTOP-1M90JOT7 MINGW64 ~/work/cloned_hello (master)
$ git remote
origin

zamba@LAPTOP-1M90JOT7 MINGW64 ~/work/cloned_hello (master)
$ git remote show origin
* remote origin
  Fetch URL: C:/Users/zamba/work/hello
  Push URL: C:/Users/zamba/work/hello
  HEAD branch: master
  Remote branches:
    master tracked
    style tracked
  Local branch configured for 'git pull':
    master merges with remote master
  Local ref configured for 'git push':
    master pushes to master (up to date)

```

Просмотр имени по умолчанию удаленного репозитория

Удаленные ветки

Посмотрим на ветки, доступные в нашем клонированном репозитории. Можно увидеть, что в списке только ветка master.

Изменение оригинального репозитория

Перейдем в репозиторий hello. Внесем изменения в файл README.md. Затем добавим их в репозиторий (рис. @fig:043).

```

zamba@LAPTOP-1M90J0T7 MINGW64 ~/work/cloned_hello (master)
$ git branch
* master

zamba@LAPTOP-1M90J0T7 MINGW64 ~/work/cloned_hello (master)
$ git branch -a
* master
  remotes/origin/HEAD -> origin/master
  remotes/origin/master
  remotes/origin/style

zamba@LAPTOP-1M90J0T7 MINGW64 ~/work/cloned_hello (master)
$ cd ../hello

zamba@LAPTOP-1M90J0T7 MINGW64 ~/work/hello (master)
$ git add README
fatal: pathspec 'README' did not match any files

zamba@LAPTOP-1M90J0T7 MINGW64 ~/work/hello (master)
$ git add README.md

zamba@LAPTOP-1M90J0T7 MINGW64 ~/work/hello (master)
$ git commit -m "Changed README in original repo"
[master 5684cdf] Changed README in original repo
1 file changed, 1 insertion(+), 1 deletion(-)

```

Изменение оригинального репозитория

Перейдём в клон репозитория и используем команду `git fetch`, которая будет извлекать новые коммиты из удаленного репозитория, но не будет сливать их с наработками в локальных ветках(рис. @fig:044).

```

zamba@LAPTOP-1M90J0T7 MINGW64 ~/work/hello (master)
$ cd ../cloned_hello

zamba@LAPTOP-1M90J0T7 MINGW64 ~/work/cloned_hello (master)
$ git fetch
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 3 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (3/3), 323 bytes | 26.00 KiB/s, done.
From C:/Users/zamba/work/hello
   77c5291..5684cdf  master    -> origin/master

zamba@LAPTOP-1M90J0T7 MINGW64 ~/work/cloned_hello (master)
$ git log --all
commit 5684cdfca0fa01c5d788f676e59239172ac98677 (origin/master, origin/HEAD)
Author: dina <1132226536@pfur.ru>
Date:   Sat Feb 15 21:42:28 2025 +0300

    Changed README in original repo

commit 77c529194578880085eee3392f99e505bc2520cf (HEAD -> master, origin/style)
Author: dina <1132226536@pfur.ru>
Date:   Sat Feb 15 20:57:39 2025 +0300

    Updated index.html

commit 3c8eef665633ac400eeca9dbec253368b5cf87dd
Author: dina <1132226536@pfur.ru>
Date:   Sat Feb 15 20:54:49 2025 +0300

    Hello uses style.css

commit 51e28697339b4e53911b6516932861a8ac7a7a6a
Author: dina <1132226536@pfur.ru>
Date:   Sat Feb 15 20:52:07 2025 +0300

    Added css stylesheet

commit b2954c34fae578928709eb3c621cded5ae45a380
Author: dina <1132226536@pfur.ru>
Date:   Sat Feb 15 21:04:52 2025 +0300

    Added README

```

Извлечение изменений

Слияние извлеченных изменений

Сольём внесённые изменения в главную ветку. Также можно было бы использовать команду `git pull`, которая является объединением `fetch` и `merge` в одну команду (рис. @fig:045).


```

zamba@LAPTOP-1M90JOT7 MINGW64 ~/work/cloned_hello (master)
$ cat README.md
This is the Hello World example from the git tutorial.

zamba@LAPTOP-1M90JOT7 MINGW64 ~/work/cloned_hello (master)
$ git merge origin/master
Updating 77c5291..5684cdf
Fast-Forward
 README.md | 2 +-
 1 file changed, 1 insertion(+), 1 deletion(-)

zamba@LAPTOP-1M90JOT7 MINGW64 ~/work/cloned_hello (master)
$ cat README.md
This is the Hello World example from the git tutorial.
zamba@LAPTOP-1M90JOT7 MINGW64 ~/work/cloned_hello (master)
$ git pull
Already up to date.

```

Слияние извлеченных изменений

Добавление ветки наблюдения

Добавим локальную ветку, которая отслеживает удаленную ветку, теперь мы можем видеть ветку style в списке веток и логе(рис. @fig:046).

```

zamba@LAPTOP-1M90JOT7 MINGW64 ~/work/cloned_hello (master)
$ git branch --track style origin/style
branch 'style' set up to track 'origin/style'.

zamba@LAPTOP-1M90JOT7 MINGW64 ~/work/cloned_hello (master)
$ git branch -a
* master
  style
  remotes/origin/HEAD -> origin/master
  remotes/origin/master
  remotes/origin/style

zamba@LAPTOP-1M90JOT7 MINGW64 ~/work/cloned_hello (master)
$ git log --max-count=2
commit 5684cdfca0fa01c5d788f676e59239172ac98677 (HEAD -> master, origin/master, origin/HEAD)
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 21:42:28 2025 +0300

    Changed README in original repo

commit 77c529194578880085eee3392f09e505bc2520cf (origin/style, style)
Author: dina <1132226536@pfur.ru>
Date: Sat Feb 15 20:57:39 2025 +0300

    Updated index.html

```

Добавление ветки наблюдения

Создание чистого репозитория

Как правило, репозитории, оканчивающиеся на .git являются чистыми репозиториями. Создадим такой в рабочем каталоге. Затем добавим репозиторий hello.git к нашему оригинальному репозиторию(рис. @fig:047).

```

zamba@LAPTOP-1M90JOT7 MINGW64 ~/work/cloned_hello (master)
$ cd ..

zamba@LAPTOP-1M90JOT7 MINGW64 ~/work
$ git clone --bare hello hello.git
Cloning into bare repository 'hello.git'...
done.

zamba@LAPTOP-1M90JOT7 MINGW64 ~/work
$ ls hello.git
HEAD config description hooks/ info/ objects/ packed-refs refs/

zamba@LAPTOP-1M90JOT7 MINGW64 ~/work
$ cd hello

zamba@LAPTOP-1M90JOT7 MINGW64 ~/work/hello (master)
$ git remote add shared ../hello.git

```

Создание чистого репозитория

Отправка и извлечение изменений

Так как чистые репозитории, как правило, расшариваются на каком-нибудь сетевом сервере, нам необходимо отправить наши изменения в другие репозитории. Начнем с

создания изменения для отправки. Отредактируем файл README.md и сделаем коммит, затем отправим изменения в общий репозиторий. Затем извлечем изменения из общего репозитория(рис. @fig:048, @fig:049).

```
zamba@LAPTOP-1W90J0T7 MINGW64 ~/work/hello (master)
$ git checkout master
Already on 'master'
M   README.md

zamba@LAPTOP-1W90J0T7 MINGW64 ~/work/hello (master)
$ git add README.md

zamba@LAPTOP-1W90J0T7 MINGW64 ~/work/hello (master)
$ git commit -m "Added shared comment to readme"
[master 4d12a91] Added shared comment to readme
1 file changed, 2 insertions(+), 1 deletion(-)

zamba@LAPTOP-1W90J0T7 MINGW64 ~/work/hello (master)
$ git push shared master
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 12 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 395 bytes | 395.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To ../hello.git
5684cdf..4d12a91 master -> master
```

Отправка изменений

```
zamba@LAPTOP-1W90J0T7 MINGW64 ~/work/hello (master)
$ cd ../cloned_hello

zamba@LAPTOP-1W90J0T7 MINGW64 ~/work/cloned_hello (master)
$ git remote add shared ../hello.git

zamba@LAPTOP-1W90J0T7 MINGW64 ~/work/cloned_hello (master)
$ git branch --track shared master
branch 'shared' set up to track 'master'.

zamba@LAPTOP-1W90J0T7 MINGW64 ~/work/cloned_hello (master)
$ git pull shared master
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (3/3), 375 bytes | 25.00 KiB/s, done.
From ../hello
 * branch          master       -> FETCH_HEAD
 * [new branch]    master       -> shared/master
Updating 5684cdf..4d12a91
Fast-forward
 README.md | 3 ++-
 1 file changed, 2 insertions(+), 1 deletion(-)

zamba@LAPTOP-1W90J0T7 MINGW64 ~/work/cloned_hello (master)
$ cat README.md
This is the Hello World example from the git tutorial.
(Changed in the original and pushed to shared)
```

Извлечение изменений

Выводы

В процессе выполнения данной лабораторной работы я приобрела практические навыки работы с Git.

Список литературы