

МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И МАССОВЫХ
КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ

Ордена Трудового Красного Знамени федеральное государственное
бюджетное образовательное
учреждение высшего образования
«Московский технический университет связи и информатики»

Кафедра "Системное программирование"

Лабораторная работа № 1
Основы работы с системой контроля версий по
дисциплине
“Информационные технологии и программирование”

Выполнил: студент гр. БВТ2201
Шамсутдинов Р.Ф.

Москва, 2023 г

Цель работы: изучение базовых функций и подходов к командной разработке в системе контроля версий Git

Задание: определить проект (состоящий из нескольких файлов), который будет добавлен под версионный контроль, и произвести ряд операций в СКВ, включающих:

1) Создание репозитория или клонирование существующего

- 2) Внесение ряда изменений в файл(ы) и выполнение коммитов (кол-во не меньше 2)
 - 3) Откат изменений до одного из предыдущих коммитов двумя разными способами
 - 4) Создание, переименование и удаление ветки
 - 5) Слияние ветки с веткой мастер с решением конфликта содержания файлов
 - 6) Настройка игнорирования git'ом файлов определенного расширения с использованием glob-шаблонов
 - 7) Вывод лога коммитов в измененном пользовательском формате
 - 8) Настройка удаленного репозитория: получение и отправка изменений в репозиторий
 - 9) Организация совместной работы над проектом: клонирование чужого репозитория, выполнение коммитов, отправка изменений
- Перечень основных используемых команд (\$ git ...):

```

init
add
commit
status
log
diff
branch
checkout
merge
reset
clone
fetch
pull
push

```

Ход работы:

- 1) Создание репозитория или клонирование существующего git init (инициализируем гит репозиторий)
- 2) Внесение ряда изменений в файл(ы) и выполнение коммитов

git status (смотрим статус у отслеживаемых файлов) git

add . (добавляем все отслеживаемые файлы в индекс)

git commit -m "описание коммита" (создаем коммит из всех файлов, содержащихся в индексе) git log (смотрим историю коммитов)

```

PS C:\Users\фввшт\IdeaProjects\lab> git log
commit 448f8e588c055b626033e26ac441622abd5485da (HEAD -> main)
Author: Sypoo1 <h89672142336@yandex.ru>
Date: Sat Sep 9 19:45:02 2023 +0300

    create new method in Main class

commit 6ad63e06e00fe92a36f6adcbac755b3b1bf0b51c
Author: Sypoo1 <h89672142336@yandex.ru>
Date: Sat Sep 9 19:43:16 2023 +0300

    create Logic class

commit c85f031b29965170eea6f91e66ae48336312b446
Author: Sypoo1 <h89672142336@yandex.ru>
Date: Sat Sep 9 19:42:18 2023 +0300

    create Data class

commit 154b4c75c5c30befc2eb37bc8b827f2732e090ca
Author: Sypoo1 <h89672142336@yandex.ru>
Date: Sat Sep 9 18:46:50 2023 +0300

    initial commit
PS C:\Users\фввшт\IdeaProjects\lab>

```

3) Откат изменений до одного из предыдущих коммитов двумя разными способами

а) `git revert 448f8e5` (создает новый коммит, который отменяет изменения, сделанные в указанном коммите)

```
Terminal Local x
448f8e5 (HEAD -> main) create new method in Main class
6ad63e0 create Logic class
c85f031 create Data class
154b4c7 initial commit
PS C:\Users\фввшт\IdeaProjects\lab> git revert 448f8e5
[main df41f2a] Revert "create new method in Main class"
1 file changed, 4 deletions(-)
PS C:\Users\фввшт\IdeaProjects\lab> git log --oneline
df41f2a (HEAD -> main) Revert "create new method in Main class"
448f8e5 create new method in Main class
6ad63e0 create Logic class
c85f031 create Data class
154b4c7 initial commit
PS C:\Users\фввшт\IdeaProjects\lab> 
```

б) `git reset --hard 448f8e5` (удаляем все коммиты до указанного, а также все изменения с помощью флага `--hard`)

```
Terminal Local x + v
PS C:\Users\фввшт\IdeaProjects\lab> git log --oneline
df41f2a (HEAD -> main) Revert "create new method in Main class"
448f8e5 create new method in Main class
6ad63e0 create Logic class
c85f031 create Data class
154b4c7 initial commit
PS C:\Users\фввшт\IdeaProjects\lab> git reset --hard 448f8e5
HEAD is now at 448f8e5 create new method in Main class
PS C:\Users\фввшт\IdeaProjects\lab> git log --oneline
448f8e5 (HEAD -> main) create new method in Main class
6ad63e0 create Logic class
c85f031 create Data class
154b4c7 initial commit
PS C:\Users\фввшт\IdeaProjects\lab> 
```

4)Создание, переименование и удаление ветки

`git branch` (смотрим какие есть ветки) `git checkout -b test` (создаем ветку test и переходим на неё) `git branch -m test new_test` (меняем имя ветки test на new_test) `git checkout main` (переходим на ветку main) `git branch -D new_test` (удаляем ветку new_test)

```

PS C:\Users\фввшт\IdeaProjects\lab> git branch
* main
PS C:\Users\фввшт\IdeaProjects\lab> git checkout -b test
Switched to a new branch 'test'
PS C:\Users\фввшт\IdeaProjects\lab> git branch
main
* test
PS C:\Users\фввшт\IdeaProjects\lab> git branch -m test new_test
PS C:\Users\фввшт\IdeaProjects\lab> git branch
main
* new_test
PS C:\Users\фввшт\IdeaProjects\lab> git checkout main
Switched to branch 'main'
PS C:\Users\фввшт\IdeaProjects\lab> git branch -D new_test
Deleted branch new_test (was 448f8e5).
PS C:\Users\фввшт\IdeaProjects\lab> git branch
* main
PS C:\Users\фввшт\IdeaProjects\lab>

```

5) Слияние ветки с веткой мастер с решением конфликта содержания файлов
git merge test (слияние ветки test с веткой main) git status
(видим что есть конфликт в файле src/Main.java)

```

PS C:\Users\фввшт\IdeaProjects\lab> git log --oneline --graph --branches=*
* f47e268 (HEAD -> main) create new method summa
| * d494de2 (test) edit bye method
|/
* 448f8e5 create new method in Main class
* 6ad63e0 create Logic class
* c85f031 create Data class
* 154b4c7 initial commit
PS C:\Users\фввшт\IdeaProjects\lab> git status
On branch main
You have unmerged paths.
  (fix conflicts and run "git commit")
  (use "git merge --abort" to abort the merge)

Unmerged paths:
  (use "git add <file>..." to mark resolution)
        both modified:   src/Main.java

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

```

решаем конфликт путем редактирования нужного файла, делаем коммит

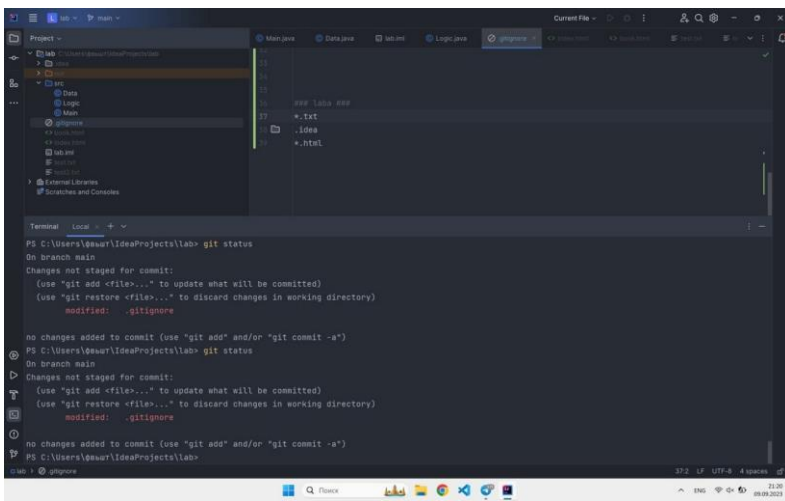
```

PS C:\Users\фввшт\IdeaProjects\lab> git log --oneline --graph --branches=*
* 62dfe33 (HEAD -> main) merge main and test
| \
| * d494de2 (test) edit bye method
* | f47e268 create new method summa
|/
* 448f8e5 create new method in Main class
* 6ad63e0 create Logic class
* c85f031 create Data class
* 154b4c7 initial commit

```

6) Настройка игнорирования git'ом файлов определенного расширения с использованием glob-шаблонов

В файл .gitignore добавляем имена файлов, которые не хотим отслеживать. Для игнорирования файлов с определенным расширением, добавляем * перед .названием расширения



7) Вывод логов коммитов в измененном пользовательском формате

`git log --oneline --graph` (выводим все коммиты, флаг `--oneline` выводит информацию о каждом коммите в одну строку, а также сокращает хеш коммита до 7 символов, флаг `--graph` отображает граф с ветвлениями и слияниями)

```
PS C:\Users\фввшт\IdeaProjects\lab> git log --oneline --graph
* 62dfe33 (HEAD -> main) merge main and test
| \
| * d494de2 (test) edit bye method
| * | f47e268 create new method summa
| /
* 448f8e5 create new method in Main class
* 6ad63e0 create Logic class
* c85f031 create Data class
* 154b4c7 initial commit
PS C:\Users\фввшт\IdeaProjects\lab>
```

8) Настройка удаленного репозитория: получение и отправка изменений в репозиторий

`git remote add origin https://github.com/Sypoo1/mtuci-java-labs.git` (добавляем удаленный репозиторий с сокращенным именем `origin`)

`git remote -v` (выводим доступные удаленные репозитории)

`git push origin main` (отправляем последний коммит ветки `main` на удаленный

репозиторий) `git remote show origin` (выводим информацию об удаленном репозитории)

`git fetch origin` (забирает последнюю версию с удаленного репозитория)

`git pull origin main` (делает то же самое, что и `git fetch origin`, но также бытается слить изменения в ветку `main`)

```
PS C:\Users\фввшт\IdeaProjects\lab> git remote add origin https://github.com/Sypoo1/mtuci-java-labs.git
PS C:\Users\фввшт\IdeaProjects\lab> git remote -v
origin https://github.com/Sypoo1/mtuci-java-labs.git (fetch)
origin https://github.com/Sypoo1/mtuci-java-labs.git (push)
PS C:\Users\фввшт\IdeaProjects\lab> git push origin main
Enumerating objects: 30, done.
Counting objects: 100% (30/30), done.
Delta compression using up to 12 threads
Compressing objects: 100% (27/27), done.
Writing objects: 100% (30/30), 2.63 KiB | 897.00 KiB/s, done.
Total 30 (delta 14), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (14/14), done.
To https://github.com/Sypoo1/mtuci-java-labs.git
* [new branch] main -> main
PS C:\Users\фввшт\IdeaProjects\lab>
```

```
PS C:\Users\фввшт\IdeaProjects\lab> git remote show origin
* remote origin
Fetch URL: https://github.com/Sypoo1/mtuci-java-labs.git
Push URL: https://github.com/Sypoo1/mtuci-java-labs.git
HEAD branch: main
Remote branch:
  main tracked
Local ref configured for 'git push':
  main pushes to main (up to date)
PS C:\Users\фввшт\IdeaProjects\lab>
```

```
PS C:\Users\фввшт\IdeaProjects\lab> git fetch origin
PS C:\Users\фввшт\IdeaProjects\lab> git pull origin main
From https://github.com/Sypoo1/mtuci-java-labs
  * branch          main          -> FETCH_HEAD
Already up to date.
PS C:\Users\фввшт\IdeaProjects\lab>
```

9) Организация совместной работы над проектом: клонирование чужого репозитория, выполнение коммитов, отправка изменений

git clone https://github.com/Nazhik123/inf_laba.git (клонировем проект из удаленного репозитория) вносим изменения, создаем коммит, отправляем изменения на удаленный репозиторий

Commits

master

Commits on Sep 12, 2023

- rustams commit
Sypoo1 committed last week [3a58cc1](#) <>

Commits on Sep 11, 2023

- add
Nazhik123 committed 2 weeks ago [4061bf1](#) <>
- bm
Nazhik123 committed 2 weeks ago [6798e54](#) <>
- master_edit
Nazhik123 committed 2 weeks ago [f33d680](#) <>
- V4edit
Nazhik123 committed 2 weeks ago [7da5318](#) <>
- V2
Nazhik123 committed 2 weeks ago [018c264](#) <>
- add_file2
Nazhik123 committed 2 weeks ago [61fce65](#) <>
- add_file
Nazhik123 committed 2 weeks ago [33e8bf1](#) <>

Newer Older

Вывод: проделав данную работу, я познакомился с базовыми командами контроля версий Git, научился отслеживать версии проекта, а также взаимодействовать с удаленными репозиториями на GitHub.