

Основы работы с ветвями в GIT. Основы слияния. Разрешение конфликтов.

Часть 1. Создание ветвей и слияние.

1. Создайте новую ветвь greet_name (где name – Ваша фамилия), используя команду

```
git checkout -b <branchname>
Yurii_Kosakivskyi@EPUAKYIW2659 MINGW64 ~/Desktop/GIT_Task (master)
$ git checkout -b greet_Kosakivskyi
Switched to a new branch 'greet_Kosakivskyi'
```

Добавьте новый каталог (пример lib_name) и файл greeter_name.rb с содержимым:

```
class Greeter
  def initialize(who)
    @who = who
  end
  def greet
    "Hello, #{@who}"
  end
end
```

2. Сделайте 3 commits в ветку greet_name

```
Yurii_Kosakivskyi@EPUAKYIW2659 MINGW64 ~/Desktop/GIT_Task (greet_Kosakivskyi)
$ git add -A

Yurii_Kosakivskyi@EPUAKYIW2659 MINGW64 ~/Desktop/GIT_Task (greet_Kosakivskyi)
$ git commit -m "change file#2"
[greet_Kosakivskyi 3083549] change file#2
 1 file changed, 2 insertions(+)
```

3. Продемонстрируйте переключение на ветку master.

```
$ git checkout master
Switched to branch 'master'
```

4. Создайте файл README и произведите commit в master.

```
$ touch README.md

Yurii_Kosakivskyi@EPUAKYIW2659 MINGW64 ~/Desktop/GIT_Task (master)
$ git add -A

Yurii_Kosakivskyi@EPUAKYIW2659 MINGW64 ~/Desktop/GIT_Task (master)
$ git commit -m "add readme.md"
[master d24d949] add readme.md
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 README.md
```

5. Произведите слияние ветвей greet и master – продемонстрируйте историю commits.

```

$ git merge greet_Kosakivskyi
Merge made by the 'recursive' strategy.
 lib_Kosakivskyi/greeter_Kosakivskyi.rb | 10 ++++++++
 1 file changed, 10 insertions(+)
 create mode 100644 lib_Kosakivskyi/greeter_Kosakivskyi.rb

Revert "Revert "Revert "part 5/5
Kosakivskyi"""

This reverts commit
326729fb096b68d936b04b0de36467a502eb00e6.

commit
326729fb096b68d936b04b0de36467a502eb00e6
Author: Yurii_tmp <example@mail.com>
Date:   Fri May 24 15:52:13 2019 +0300

Revert "Revert "Revert "part 5/5
Kosakivskyi"""

This reverts commit
bdb58a1e55e361a00ac6bf3a7fde52a4ed3f72ff.

commit
bdb58a1e55e361a00ac6bf3a7fde52a4ed3f72ff
Author: Yurii_tmp <example@mail.com>
Date:   Fri May 24 15:51:03 2019 +0300

commit
8c6a3e13d5cb921bcfd98eabb9fb7f365d44ec87
Author: Yurii_tmp <example@mail.com>
Date:   Fri May 24 15:50:54 2019 +0300

part 5/5 Kosakivskyi

commit
a7ea2bb3cbf0d89c27c80fa41d17d320f7f32b5a
Author: Yurii_tmp <example@mail.com>
Date:   Fri May 24 15:33:57 2019 +0300

2bd file changed

commit
7f7dd5b3dea2aea326665c2e73d4af227a3f5bc1
Author: Yurii_tmp <example@mail.com>
Date:   Fri May 24 15:33:30 2019 +0300
:

commit
f6690be828ac18c63ec08c3408767626822e259e

commit
f6690be828ac18c63ec08c3408767626822e259e
(HEAD -> master)
Merge: d24d949 3083549
Author: Yurii_tmp <example@mail.com>
Date:   Fri May 24 16:13:16 2019 +0300

Merge branch 'greet_Kosakivskyi'

commit
d24d9498ab5dc22a110af27aa8a39fea653c561e
Author: Yurii_tmp <example@mail.com>
Date:   Fri May 24 16:12:41 2019 +0300

add readme.md

commit
30835491197b9cbad1ac6e18e7f5f007a0b1de46
(greet_Kosakivskyi)
Author: Yurii_tmp <example@mail.com>
Date:   Fri May 24 16:09:47 2019 +0300

change file#2

commit
04c63d27c7ae8d460fdf3265e90e11fd5b34559f
Author: Yurii_tmp <example@mail.com>
Date:   Fri May 24 16:09:27 2019 +0300

change file

commit
4d26966d3d6894a64ea8fdf660b162d7e84a7b8a
Author: Yurii_tmp <example@mail.com>
Date:   Fri May 24 16:04:00 2019 +0300

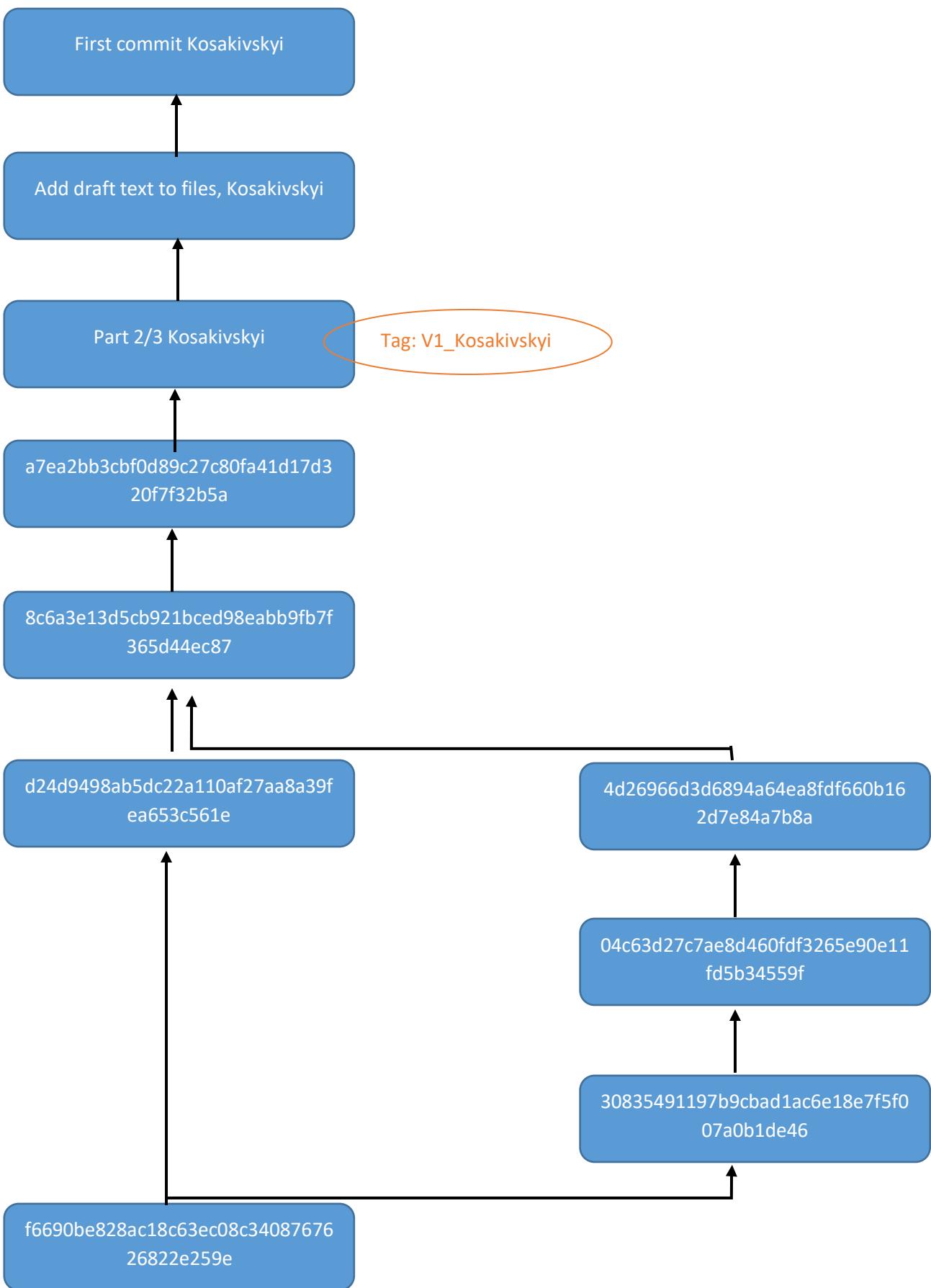
first commit on new branch

commit
8f054fdf51708ba352d9ab4216cbde6a41242bae
Author: Yurii_tmp <example@mail.com>
Date:   Fri May 24 15:53:11 2019 +0300

:

```

6. Изобразите в удобном для Вас графическом редакторе граф коммитов, где вершинами графа будут коды коммитов.



Часть 2.

Смоделируйте искусственно конфликт, при слиянии описанном в предыдущей части.

Конфликт можно легко создать изменив один и тот же файл на разных ветвях.

```
Yurii_Kosakivskyi@EPUAKYIW2659 MINGW64 ~/Desktop/GIT_Task - Copy (master)
$ git checkout
greet_Kosakivskyi    master          V1_Kosakivskyi
HEAD                ORIG_HEAD

Yurii_Kosakivskyi@EPUAKYIW2659 MINGW64 ~/Desktop/GIT_Task - Copy (master)
$ git checkout greet_Kosakivskyi
Switched to branch 'greet_Kosakivskyi'

Yurii_Kosakivskyi@EPUAKYIW2659 MINGW64 ~/Desktop/GIT_Task - Copy (greet_Kosakivskyi)
$ git add -A

Yurii_Kosakivskyi@EPUAKYIW2659 MINGW64 ~/Desktop/GIT_Task - Copy (greet_Kosakivskyi)
$ git commit -m "ds"
[greet_Kosakivskyi 2be1929] ds
 1 file changed, 2 insertions(+), 1 deletion(-)

Yurii_Kosakivskyi@EPUAKYIW2659 MINGW64 ~/Desktop/GIT_Task - Copy (greet_Kosakivskyi)
$ git checkout master
Switched to branch 'master'

Yurii_Kosakivskyi@EPUAKYIW2659 MINGW64 ~/Desktop/GIT_Task - Copy (master)
$ git add -A

Yurii_Kosakivskyi@EPUAKYIW2659 MINGW64 ~/Desktop/GIT_Task - Copy (master)
$ git commit -m"sdf"
[master c0147a1] sdf
 1 file changed, 2 insertions(+)

Yurii_Kosakivskyi@EPUAKYIW2659 MINGW64 ~/Desktop/GIT_Task - Copy (master)
$ git merge greet_Kosakivskyi
Auto-merging student_Kosakivskiy/hello_Yuri2.rb
CONFLICT (content): Merge conflict in student_Kosakivskiy/hello_Yuri2.rb
Automatic merge failed; fix conflicts and then commit the result.

Yurii_Kosakivskyi@EPUAKYIW2659 MINGW64 ~/Desktop/GIT_Task - Copy (master|MERGING)
|
$ |
```

Часть 3.

Используя команду reset верните указатель на состояние ветвей до их слияния.

```
Yurii_Kosakivskyi@EPUAKYIW2659 MINGW64 ~/Desktop/GIT_Task - Copy (2) (master)
$ git reset greet_Kosakivskyi~4
Unstaged changes after reset:
D     student_Kosakivskiy/New folder/hello_Yurii.rb

Yurii_Kosakivskyi@EPUAKYIW2659 MINGW64 ~/Desktop/GIT_Task - Copy (2) (master)
$ git reset master~2
Unstaged changes after reset:
D     student_Kosakivskiy/New folder/hello_Yurii.rb
```

Часть 4.

Проделайте шаги 3-5 в части 1. Выполните слияние изменений командой rebase. Изобразите в удобном для Вас графическом редакторе граф коммитов, где вершинами графа будут коды коммитов.

Объясните разницу между merge и rebase. На примере Ваших графов.

График на следующей странице,

Насколько я понимаю после команды merge будет новый комит с изменениями с двух веток, и обе ветки будут ссылаться на этот комит. А rebase – копирует комиты с одной ветки и вставляет их на другую. Будет выглядеть так как будто бы комиты были сделаны последовательно.



Для начала работы создайте/зарегистрируйтесь (в случае отсутствия на текущий момент) на удаленном сервисе (типа *github*) , т.е. инициализируйте удаленный репозиторий.

Часть 5.

1. Разбейтесь на пары.
2. Выберите в качестве общего репозитория один из двух удаленных.
3. Student1 заливает все содержимое своих папок в удаленный репозиторий.

```
$ git init
Initialized empty Git repository in C:/Users/Yuri_Kosakivskyi/Desktop/GIT/Tasks/GIT-2/New folder/.git

Yuri_Kosakivskyi@EPUAKYIW2659 MINGW64 ~/Desktop/GIT/Tasks/GIT-2/New folder (master)
$ git clone https://github.com/ezzvarych/LabJavaGithubTask5.git
Cloning into 'LabJavaGithubTask5'...
remote: Enumerating objects: 61, done.
remote: Counting objects: 100% (61/61), done.
remote: Compressing objects: 100% (33/33), done.
remote: Total 61 (delta 1), reused 61 (delta 1), pack-reused 0
Unpacking objects: 100% (61/61), done.
```

4.

Student 2 «забирает» проект к себе в локальный репозиторий.

```
$ git pull https://github.com/ezzvarych/LabJavaGithubTask5.git master
remote: Enumerating objects: 61, done.
remote: Counting objects: 100% (61/61), done.
remote: Compressing objects: 100% (33/33), done.
remote: Total 61 (delta 1), reused 61 (delta 1), pack-reused 0
Unpacking objects: 100% (61/61), done.
From https://github.com/ezzvarych/LabJavaGithubTask5
 * branch           master      -> FETCH_HEAD
```

5. Выполните команду `git status`.

```
Yuri_Kosakivskyi@EPUAKYIW2659 MINGW64 ~/Desktop/GIT/Tasks/GIT-2/New folder (master)
$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)

    LabJavaGithubTask5/

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

6. Проведите каждый по три коммита в локальный репозиторий и «залейте изменения» в удаленный.

```
[ter] $ touch somefile.txt
ation Yurii_Kosakivskyi@EPUAKYIW2659 MINGW64 ~/Desktop/GIT/Tasks/GIT-2/New folder (mas
cument ter)
current $ echo somefile.txt >> "hello world"
Page Yurii_Kosakivskyi@EPUAKYIW2659 MINGW64 ~/Desktop/GIT/Tasks/GIT-2/New folder (mas
ter)
$ git add -A
warning: adding embedded git repository: LabJavaGithubTask5
$ git commit -m "fitst commit Kosakivskyi"
[master 0c7322a] fitst commit Kosakivskyi
 3 files changed, 2 insertions(+)
   create mode 160000 LabJavaGithubTask5
   create mode 100644 hello world
   create mode 100644 somefile.txt
$ git push https://github.com/ezzvarych/LabJavaGithubTask5.git master
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (4/4), 458 bytes | 458.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0)
To https://github.com/ezzvarych/LabJavaGithubTask5.git
  e939b47..0c7322a  master -> master
```

```
[ter] $ echo somefile.txt >> "hello world 2"
Yurii_Kosakivskyi@EPUAKYIW2659 MINGW64 ~/Desktop/GIT/Tasks/GIT-2/New folder (master)
$ git add -A
warning: LF will be replaced by CRLF in hello world 2.
The file will have its original line endings in your working directory
Yurii_Kosakivskyi@EPUAKYIW2659 MINGW64 ~/Desktop/GIT/Tasks/GIT-2/New folder (master)
$ git commit -m "2nd commit Kosakivskyi"
[master ddcd4c9] 2nd commit Kosakivskyi
 1 file changed, 1 insertion(+)
   create mode 100644 hello world 2
Yurii_Kosakivskyi@EPUAKYIW2659 MINGW64 ~/Desktop/GIT/Tasks/GIT-2/New folder (master)
$ git push https://github.com/ezzvarych/LabJavaGithubTask5.git master
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 253 bytes | 253.00 KiB/s, done.
Total 2 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/ezzvarych/LabJavaGithubTask5.git
  0c7322a..ddcd4c9  master -> master
```

```
Yuriy_Kosakivskyi@EPUAKYIW2659 MINGW64 ~/Desktop/GIT/Tasks/GIT-2/New folder (master)
$ echo somefile.txt >> "hello world 3"

Yuriy_Kosakivskyi@EPUAKYIW2659 MINGW64 ~/Desktop/GIT/Tasks/GIT-2/New folder (master)
$ git add -A
warning: LF will be replaced by CRLF in hello world 3.
The file will have its original line endings in your working directory

Yuriy_Kosakivskyi@EPUAKYIW2659 MINGW64 ~/Desktop/GIT/Tasks/GIT-2/New folder (master)
$ git commit -m "3nd commit Kosakivskyi"
[master 00a86d7] 3nd commit Kosakivskyi
 1 file changed, 1 insertion(+)
 create mode 100644 hello world 3

Yuriy_Kosakivskyi@EPUAKYIW2659 MINGW64 ~/Desktop/GIT/Tasks/GIT-2/New folder (master)
$ git push https://github.com/ezzvarych/LabJavaGithubTask5.git master
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 236 bytes | 236.00 KiB/s, done.
Total 2 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/ezzvarych/LabJavaGithubTask5.git
 ddcd4c9..00a86d7  master -> master
```