

Homework_Lesson10_OS_Repositories

Индивидуальная часть: На созданном аккаунте github.com создать репозиторий и клонировать его себе локально. Настроить себе доступы для того, чтобы можно было пушить изменения в ветки. Init commit может быть и локальный, это на ваше усмотрение. Далее реализовать следующие сценарии

1. Создать 5-10 коммитов. (Пример: создать файлы, поменять их содержимое). Вывести их лог экран, сделать скрин и добавить в отчёт.

2. С помощью reflog перейти к предыдущему коммиту (на ваше усмотрение). Output, screen, report. (Вывести их лог экран, сделать скрин и добавить в отчёт.) <https://youtu.be/XRV9kai-3mc> (небольшая помощь, если запутаетесь)

3. Создайте ветку с названием на ваше усмотрение (можно develop). OSR. (Output, screen, report.)

4. Создайте коммит и добавьте туда ещё дополнительные изменения (добавьте, например в изменяемый файл точку, пробел и тд) с помощью amend. OSR.

5. Сделайте коммит в main. Но не делайте git push (это важно!), сделайте изменения локально.

6. Сделайте так, чтобы этот коммит оказался в новой ветке с помощью git reset --hard. OSR

7. Сделайте изменения в файле локально. Сделайте коммит для этого изменения. OSR

8. Через git checkout отмените изменения в файле через откат по сохранённому хэшу. OSR

9. Начните всё заново (это важно чтобы вы делали локально то, что я указал сделать локально). Можно использовать любой из подходов. OSR Ваш основной гайд: https://pikabu.ru/story/git_shit_10252570

Оформленный отчёт запустите на ваш ПРИВАТНЫЙ репозиторий, куда вы меня уже добавили или добавьте (кто этого ещё не сделал).

Выполнение индивидуального задания:

На аккаунте SeriousMak в github.com, создан репозиторий под названием Homework_Lesson10_OS_9. Также локально клонирован к себе на wsl.

1. Создал 5 коммитов. (Пример: создать файлы, поменять их содержимое). Вывести их лог экран, сделать скрин и добавить в отчёт:

```
commit dda8f8327d58ba601227995758438e0f15dab67c (HEAD -> main)
Author: SeriousMak <sashamkr@mail.ru>
Date: Sat Nov 16 23:49:39 2024 +0300

    add echo

commit 73e97321b44290f736474d033f10e6583dce5168
Author: SeriousMak <sashamkr@mail.ru>
Date: Sat Nov 16 23:48:17 2024 +0300

    add copy file with new name

commit 27e0beaea544b88e9a983813a5911191ade7bc56
Author: SeriousMak <sashamkr@mail.ru>
Date: Sat Nov 16 23:46:02 2024 +0300

    add new file name generation

commit 2fdc3db38d180d92a5c5725eb010cf38a2be1d55
Author: SeriousMak <sashamkr@mail.ru>
Date: Sat Nov 16 23:44:36 2024 +0300

    add file extanstion reading

commit 53afca4a1371d0ec6b12a007c642fdff3d81035c
Author: SeriousMak <sashamkr@mail.ru>
Date: Sat Nov 16 23:43:18 2024 +0300

    add file path reading

commit 2eafe0c130cef6a11b5ae1f4b2da6e719188d110
Author: SeriousMak <sashamkr@mail.ru>
Date: Sat Nov 16 23:41:25 2024 +0300

    add test.sh

commit 2b28cb81a206929f76fb3a9277c6822e2d7467b3 (origin/main, origin/HEAD)
Author: SeriousMak <sashamkr@mail.ru>
Date: Sat Nov 16 23:21:55 2024 +0300

    :
```

2. С помощью reflog перешел к 3-ему коммиту.

Вводим команду git reflog. Она показывает все сделанные коммиты с их HEAD{ }.

```
● makarov@DESKTOP-UG6J7T7:~/git/Homework_Lesson10_OS_9/Homework_Lesson10_OS_9$ git reflog
dda8f83 (HEAD -> main) HEAD@{0}: commit: add echo
73e9732 HEAD@{1}: commit: add copy file with new name
27e0bea HEAD@{2}: commit: add new file name generation
2fdc3db HEAD@{3}: commit: add file extanstion reading
53afca4 HEAD@{4}: commit: add file path reading
2eafe0c HEAD@{5}: commit: add test.sh
2b28cb8 (origin/main, origin/HEAD) HEAD@{6}: clone: from github.com:SeriousMak/Homework_Lesson10_OS_9.git
○ makarov@DESKTOP-UG6J7T7:~/git/Homework_Lesson10_OS_9/Homework_Lesson10_OS_9$
```

В нашей примере, мы перешли к 3-ему коммиту вводя команду `git reset HEAD@{3}`

```
makarov@DESKTOP-UG6J7T7:~/git/Homework_Lesson10_OS_9/Homework_Lesson10_OS_9$ git reset HEAD@{3}
Unstaged changes after reset:
M   test.sh
makarov@DESKTOP-UG6J7T7:~/git/Homework_Lesson10_OS_9/Homework_Lesson10_OS_9$ git log
commit 2fdc3db38d180d92a5c5725eb010cf38a2be1d55 (HEAD -> main)
Author: SeriousMak <sashamkr@mail.ru>
Date:   Sat Nov 16 23:44:36 2024 +0300

    add file extanstion reading

commit 53afca4a1371d0ec6b12a007c642fdff3d81035c
Author: SeriousMak <sashamkr@mail.ru>
Date:   Sat Nov 16 23:43:18 2024 +0300

    add file path reading

commit 2eafe0c130cef6a11b5ae1f4b2da6e719188d110
...skipping...
commit 2fdc3db38d180d92a5c5725eb010cf38a2be1d55 (HEAD -> main)
Author: SeriousMak <sashamkr@mail.ru>
Date:   Sat Nov 16 23:44:36 2024 +0300
```

3. Создаем ветку develop.

Вводим команды:

`$git branch develop` //создаем ветку develop

`$git checkout develop` //переходим в новую созданную ветку develop

```
makarov@DESKTOP-UG6J7T7:~/git/Homework_Lesson10_OS_9/Homework_Lesson10_OS_9$ git checkout develop
M   test.sh
Switched to branch 'develop'
makarov@DESKTOP-UG6J7T7:~/git/Homework_Lesson10_OS_9/Homework_Lesson10_OS_9$
```

4. Создали коммит и изменили описание комментария с помощью:

`$git commit --amend -m "rewrite echo comment"` // данная команда позволяет править название последнего коммита.

Также через `git log` видно, что последний коммит исправлен на новое название.

```
makarov@DESKTOP-UG6J7T7:~/git/Homework_Lesson10_OS_9/Homework_Lesson10_OS_9$ git commit -m "update echo comment"
[main c6b1bb1] update echo comment
1 file changed, 10 insertions(+), 1 deletion(-)
makarov@DESKTOP-UG6J7T7:~/git/Homework_Lesson10_OS_9/Homework_Lesson10_OS_9$ git add .
makarov@DESKTOP-UG6J7T7:~/git/Homework_Lesson10_OS_9/Homework_Lesson10_OS_9$ git commit --amend -m "rewrite echo comment"
[main 1dddb89] rewrite echo comment
1 file changed, 10 insertions(+), 1 deletion(-)
makarov@DESKTOP-UG6J7T7:~/git/Homework_Lesson10_OS_9/Homework_Lesson10_OS_9$ git log
commit 1dddb89347a8ee7629f82dc4ff4050d41e2327b9 (HEAD -> main)
Author: SeriousMak <sashamkr@mail.ru>
Date:   Sun Nov 17 00:18:43 2024 +0300

    rewrite echo comment

commit 2fdc3db38d180d92a5c5725eb010cf38a2be1d55 (develop)
Author: SeriousMak <sashamkr@mail.ru>
Date:   Sat Nov 16 23:44:36 2024 +0300

    add file extanstion reading

commit 53afca4a1371d0ec6b12a007c642fdff3d81035c
Author: SeriousMak <sashamkr@mail.ru>
```

5. Сделан коммит в main. Не делали `git push`.

6. Переводим последний commit в новую ветку develop. (Случайно не туда закомитили).

```
$git reset HEAD~ // отменяет последний коммит  
$git stash
```

```
$git checkout develop//переходим в новую ветку  
$git commit -m "rewrite echo comment"
```

7. Сделаны локальные изменения в файле и сделан коммит.

```
$git commit -m "delete echo"
```

```
● makarov@DESKTOP-UG6J7T7:~/git/Homework_Lesson10_OS_9/Homework_Lesson10_OS_9$ git reset HEAD~ --hard  
HEAD is now at 1dddb89 rewrite echo comment  
● makarov@DESKTOP-UG6J7T7:~/git/Homework_Lesson10_OS_9/Homework_Lesson10_OS_9$ git checkout develop  
Switched to branch 'develop'  
○ makarov@DESKTOP-UG6J7T7:~/git/Homework_Lesson10_OS_9/Homework_Lesson10_OS_9$  
  
● makarov@DESKTOP-UG6J7T7:~/git/Homework_Lesson10_OS_9/Homework_Lesson10_OS_9$ git commit -m "delete echo"  
[develop 0d7d07d] delete echo  
1 file changed, 2 deletions(-)  
○ makarov@DESKTOP-UG6J7T7:~/git/Homework_Lesson10_OS_9/Homework_Lesson10_OS_9$
```

8. Через git checkout отменил изменения в файле через откат по сохранённому хэшу.

```
$ git checkout 2fdc3db38d180d92a5c5725eb010cf38a2be1d55
```

```
● makarov@DESKTOP-UG6J7T7:~/git/Homework_Lesson10_OS_9/Homework_Lesson10_OS_9$ git relog  
0d7d07d (HEAD -> develop) HEAD@{0}: commit: delete echo  
2fdc3db HEAD@{1}: checkout: moving from main to develop  
1dddb89 (main) HEAD@{2}: reset: moving to HEAD~  
1626c86 HEAD@{3}: commit: update extension UI  
1dddb89 (main) HEAD@{4}: commit (amend): rewrite echo comment  
c6b1bb1 HEAD@{5}: commit: update echo comment  
2fdc3db HEAD@{6}: checkout: moving from develop to main  
2fdc3db HEAD@{7}: checkout: moving from main to develop  
2fdc3db HEAD@{8}: reset: moving to HEAD@{3}  
dda8f83 HEAD@{9}: commit: add echo  
73e9732 HEAD@{10}: commit: add copy file with new name  
27e0bea HEAD@{11}: commit: add new file name generation  
2fdc3db HEAD@{12}: commit: add file extanstion reading  
53afca4 HEAD@{13}: commit: add file path reading  
2eafe0c HEAD@{14}: commit: add test.sh  
2b28cb8 (origin/main, origin/HEAD) HEAD@{15}: clone: from github.com:SeriousMak/Homework_Lesson10_OS_9.git  
● makarov@DESKTOP-UG6J7T7:~/git/Homework_Lesson10_OS_9/Homework_Lesson10_OS_9$ git checkout 2fdc3db38d180d92a5c5725eb010cf38a2be1d55  
Note: switching to '2fdc3db38d180d92a5c5725eb010cf38a2be1d55'.  
  
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 2fdc3db add file extanstion reading  
○ makarov@DESKTOP-UG6J7T7:~/git/Homework_Lesson10_OS_9/Homework_Lesson10_OS_9$
```

Выполнение задания по командам.

Командный репозиторий: https://github.com/SeriousMak/DOS24_HW10_group4

В команде каждый выполнял свой кусочек фичи. Мой был 3-ий. Добавление изменения пароля пользователя.


Скрин изменений, выполненные мной:

```
* bf4da4e (origin/develop) Merge pull request #6 from SeriousMak/FixedReadMe
14e5283 (origin/FixedReadMe) Fixed case
3e57a38 Fixed


* ba2d263 Merge pull request #5 from SeriousMak/change-password


b148866 (HEAD -> change-password, origin/change-password) Resolved Conflict

//
* a449800 (develop) Merge pull request #1 from SeriousMak/CreateUser_MedvedevaAnna
* 405ef67 (origin/CreateUser_MedvedevaAnna) Merge branch 'develop' into CreateUser_MedvedevaAnna
//
* 339f25b Merge pull request #4 from SeriousMak/CreateUser_SergeySelivonchik
* f5d9ff1 (origin/CreateUser_SergeySelivonchik) Create conflict
* 17d6080 Add functional block: List of users
* 2416778 Create User
//
* cc7667e feature/add change password
* 8b6ee76 Update README
//
* 35cdabe Return back develop branch
//
* 0847f01 (origin/main, origin/HEAD, main) Update README.md
* e6ede7e Create user_management.sh
* 2127cff Initial commit
~
```


 SeriousMak commented last week Owner ...

No description provided.


 SeriousMak added 2 commits last week




 Update README 8b6ee76




 feature/add change password cc7667e



 SeriousMak requested a review from annette-medvedeva last week



 Resolved Conflict b148866



 annette-medvedeva merged commit ba2d263 into develop last week Revert

 Pull request successfully merged and closed Delete branch

You're all set—the change-password branch can be safely deleted.

После разрешения конфликтов скрипт выглядит таким образом:

```
$ user_management.sh
1  #!/bin/bash
2  case "$1" in
3      case "$1" in
4          1)
5              read -p "Enter username to add: " username
6              sudo useradd "$username"
7              echo "User $username added."
8              ;;
9          2)
10             echo "smth"
11             3)
12             read -p "Enter username to change password: " username
13             sudo passwd "$username"
14             echo "Password changed for user $username."
15             ;;
16         4)
17             echo "List of users:"
18             cut -d: -f1 /etc/passwd
19             ;;
20         *)
21             echo "Invalid option. Please use 1-4."
22     esac
```

Коммиты:

```
● makarov@DESKTOP-UG6J7T7:~/DOS24_Hw10_group4$ git reflog
b148866 (HEAD -> change-password, origin/change-password) HEAD@{0}: commit (merge): Resolved Conflict
cc7667e HEAD@{1}: checkout: moving from develop to change-password
a449800 (develop) HEAD@{2}: pull origin develop: Fast-forward
35cdabe HEAD@{3}: checkout: moving from change-password to develop
cc7667e HEAD@{4}: commit: feature/add change password
8b6ee76 HEAD@{5}: commit: Update README
35cdabe HEAD@{6}: checkout: moving from develop to change-password
35cdabe HEAD@{7}: checkout: moving from main to develop
0847f01 (origin/main, origin/HEAD, main) HEAD@{8}: clone: from github.com:SeriousMak/DOS24_Hw10_group4.git
○ makarov@DESKTOP-UG6J7T7:~/DOS24_Hw10_group4$
```