

Лабораторная работа №6

Справка:

```
@archlinux ~/vrpo> git help
git [-v | --version] [-h | --help] [-C <path>] [-c <name>=<value>]
  [--exec-path[=<path>]] [--html-path] [--man-path] [--info-path]
  [-p | --paginate | -P | --no-pager] [--no-replace-objects] [--bare]
  [--git-dir=<path>] [--work-tree=<path>] [--namespace=<name>]
  [--config-env=<name>=<envvar>] <command> [<args>]

are common Git commands used in various situations:

a working area (see also: git help tutorial)
ne      Clone a repository into a new directory
t       Create an empty Git repository or reinitialize an existing one

n the current change (see also: git help everyday)
        Add file contents to the index
        Move or rename a file, a directory, or a symlink
tore    Restore working tree files
        Remove files from the working tree and from the index

e the history and state (see also: git help revisions)
ect     Use binary search to find the commit that introduced a bug
f       Show changes between commits, commit and working tree, etc
p       Print lines matching a pattern
        Show commit logs
w       Show various types of objects
tus     Show the working tree status

mark and tweak your common history
nch     List, create, or delete branches
mit     Record changes to the repository
ge      Join two or more development histories together
ase     Reapply commits on top of another base tip
et      Reset current HEAD to the specified state
tch     Switch branches
        Create, list, delete or verify a tag object signed with GPG

orate (see also: git help workflows)
ch      Download objects and refs from another repository
l       Fetch from and integrate with another repository or a local branch
h       Update remote refs along with associated objects

elp -a' and 'git help -g' list available subcommands and some
t guides. See 'git help <command>' or 'git help <concept>'
d about a specific subcommand or concept.
it help git' for an overview of the system.
@archlinux ~/vrpo> []
```

Инициализация локального репозитория:

```
See 'git help git' for an overview of the system.
purpik@archlinux ~/vrpo> git init
hint: Using 'master' as the name for the initial branch. This default
hint: is subject to change. To configure the initial branch name for
hint: of your new repositories, which will suppress this warning,
hint:
hint:   git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this
hint:
hint:   git branch -m <name>
Initialized empty Git repository in /home/purpik/vrpo/.git/
purpik@archlinux ~/vrpo (master)> []
```

Конфигурация:

```
purpik@archlinux ~/vrpo (master)> git config --global user.name "ilya"
purpik@archlinux ~/vrpo (master)> git config --global user.email billylollyisi@gmail.com
purpik@archlinux ~/vrpo (master)> 
```

```
purpik@archlinux ~/vrpo (master)> git config --list
error: cannot run less: No such file or directory
user.name=ilya
user.email=billylollyisi@gmail.com
credential.https://github.com.helper=
credential.https://github.com.helper=!/usr/bin/gh auth git-credential
credential.https://gist.github.com.helper=
credential.https://gist.github.com.helper=!/usr/bin/gh auth git-credential
core.repositoryformatversion=0
core.filemode=true
core.bare=false
core.logallrefupdates=true
purpik@archlinux ~/vrpo (master)> 
```

```
core.logallrefupdates=true
purpik@archlinux ~/vrpo (master)> touch Hello.txt
purpik@archlinux ~/vrpo (master)> git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    Hello.txt

nothing added to commit but untracked files present (use "git add" to track)
purpik@archlinux ~/vrpo (master)> 
```

```
purpik@archlinux ~/vrpo (master) [1]> git commit -m "txt"
[master (root-commit) 6cc7e3c] txt
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 Hello.txt
purpik@archlinux ~/vrpo (master)> 
```

```
purpik@archlinux ~/vrpo (master)> git log
error: cannot run less: No such file or directory
commit 6cc7e3cfeb59414730a451fc80c3832555d598256 (HEAD -> master)
Author: ilya <billylollyisi@gmail.com>
Date:   Wed Dec 6 15:46:42 2023 +0000
```

```
    txt
purpik@archlinux ~/vrpo (master)> \

```

```
purpik@archlinux ~/vrpo (master)> nano Hello.txt
purpik@archlinux ~/vrpo (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.txt

no changes added to commit (use "git add" and/or "git commit -a")
purpik@archlinux ~/vrpo (master)>
purpik@archlinux ~/vrpo (master)> 
```

```

purpik@archlinux ~/vrpo (master) [1]> git add .
purpik@archlinux ~/vrpo (master)> git commit -m "com"
[master e8e306f] com
 1 file changed, 1 insertion(+)
purpik@archlinux ~/vrpo (master)> 

```

```

purpik@archlinux ~/vrpo (master) [128]> git branch test
purpik@archlinux ~/vrpo (master)> git checkout test
Switched to branch 'test'
purpik@archlinux ~/vrpo (test)> 

```

```

purpik@archlinux ~/vrpo (test)> touch Test.html
purpik@archlinux ~/vrpo (test)> git add .
purpik@archlinux ~/vrpo (test)> git commit -m "html"
[test 2758c5b] html
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 Test.html
purpik@archlinux ~/vrpo (test)> git checkout master
Switched to branch 'master'
purpik@archlinux ~/vrpo (master)> git merge test
Updating 9b81fb6..2758c5b
Fast-forward
 Test.html | 0
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 Test.html
purpik@archlinux ~/vrpo (master)> 

```

```

purpik@archlinux ~/git-merge-test (new_branch_to_merge_later)> git checkout master
Switched to branch 'master'
purpik@archlinux ~/git-merge-test (master)> echo "append" > merge.txt
purpik@archlinux ~/git-merge-test (master)> git commit -am "toappend"
master 2b50ffc] toappend
 1 file changed, 1 insertion(+), 1 deletion(-)
purpik@archlinux ~/git-merge-test (master)> git merge new_branch_to_merge_later
Auto-merging merge.txt
CONFLICT (content): Merge conflict in merge.txt
Automatic merge failed; fix conflicts and then commit the result.
purpik@archlinux ~/git-merge-test (master|MERCING) [1]> git status
On branch master
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:   merge.txt

no changes added to commit (use "git add" and/or "git commit -a")
purpik@archlinux ~/git-merge-test (master|MERCING)> cat merge.txt
<<<<<< HEAD
append
=====
conflict
>>>>>> new_branch_to_merge_later
purpik@archlinux ~/git-merge-test (master|MERCING)> nano merge.txt
purpik@archlinux ~/git-merge-test (master|MERCING)> git add merge.txt
purpik@archlinux ~/git-merge-test (master|MERCING)> git commit -m "merged and resolved conflict in merge.txt"
master 4ad1edd] merged and resolved conflict in merge.txt

```

[illegible]

```
Showing 1 changed file with 2 additions and 0 deletions.

1 6 Факультет информационных технологий
2 6 Дисциплина "ВМГО"
3 7 курс 1, группа 4, подгруппа 1
4 8 ИИ
5 9 группа-группа
```