

Glossary

[value]	Optional value
value	User-defined value
<ref>	SHA-1 reference to commit or tag
<paths>	List of paths/wildcards to match files
git --help glossary	Git vocabulary

Init Repository

Create new git repository in current folder

```
git init
```

Create a copy of git repository by given url

```
git clone https://repository.git
```

List remote repositories

```
git remote -v
```

Create alias *origin* for remote repository

```
git remote add origin https://repository.git
```

Change url of remote repository with alias *origin*

```
git remote set-url origin https://repository-2.git
```

Manage remote repositories

```
git repository --help
```

Branching

Create new branch

```
git checkout -b branch-name [<commit>] // -B to
override existing branch
```

Switch to existing branch

```
git checkout branch-name
```

Remove branch

```
git branch -d branch-name // Use -D or -df to force
```

Rename current branch

Branching (cont)

```
git branch -m new-branch-name
```

List branches

```
git branch --list [pattern] // -r (remote) -a (all) -v
(show last commit)
// --contains <commit> (contains specified commit)
// --merged (can be safely removed) --no-merged (may
need attention)
```

Create branch without commits history

```
git checkout --orphan -b branch-name [<commit>]
```

More actions with branches

```
git branch --help
```

```
git checkout --help
```

Utilities

Tricks

GUI

Commit Changes

Add selected files to staging (prepare to commit)

```
git add <paths> // '.' will match all unstaged files
```

Describe current state of local repository

```
git status
```

Make commit with message

```
git commit -m "message"
```

Make commit and append it to previous one

```
git commit --amend
```

Make commit and append it to previous one with new message

```
git commit --amend -m "message"
```



By **Andrei Zhalezniczenka**
(a.zhalezniczenka)
[cheatography.com/a-
zhalezniczenka/](https://cheatography.com/a-zhalezniczenka/)

www.facebook.com/zhalezniczenka

Not published yet.

Last updated 13th September, 2017.

Page 1 of 2.

Sponsored by **Readability-Score.com**

Measure your website readability!

<https://readability-score.com>

↶ Undo Changes

Get the previous version of a given files

```
git checkout <commit> <paths> // HEAD is used implicitly if <commit> omitted
```

Revert certain commit by applying another one (to persist history)

```
git revert <commit>
```

Unstage files

```
git reset <paths>
```

Unstage all files

```
git reset
```

Move state to specified commit

```
git reset <commit>
// --hard (to remove changes) --soft (to leave changes staged)
```

Set specified files to state of certain revision

```
git checkout <commit> <paths>
```

⇄ Exchange with Remote

Fetch

```
git fetch
```

Pull

```
git pull
```

Push

```
git push
```

▢ Difference and Merge

☑ Hide Changes

☰ Watch History

⚙ General Configs

↑ Revisions

HEAD	Reference to last commit in current branch
<commit>^	First parent of selected commit
<commit>^^	Second (and so on) parent of specified commit
<commit>~n	N-th parent of selected commit (e.g. ^2)
<commit1>..<commit2>	
git --help	More git revision syntax here
revisions	

⚙ Advanced Configs

🔗 Links

Tips

Use Cases

Remove all the local updates

```
git reset --hard
git clean -f
```



By **Andrei Zhalezniczenka**
(a.zhalezniczenka)
cheatography.com/a-zhalezniczenka/

www.facebook.com/zhalezniczenka

Not published yet.
Last updated 13th September, 2017.
Page 2 of 2.

Sponsored by **Readability-Score.com**
Measure your website readability!
<https://readability-score.com>