

GIT TUTORIAL

change name: `git config --global user.name "ime"`

change email: `git config --global user.email "email_prijatelj"`

`git config --list` daje nam trenutnu konfiguraciju

`git help commit` 126001 nam help od commit

`pwd` daje nam gdje se nalazimo

`cd ~` ide u home folder

`git init` napravi/inicijalizira git projekat

`git add .` dodajemo sve što smo napravili u tom folderu. u master, znači tag foldera sve foldere

`git commit -m "Message"` kako komitujemo i dodajemo poruku šta smo komitali

`git log` tko je šta radio i kada dodao

`git log --author "Name of author"` daje nam šta je sve tag autor dodao ili izmijenio

`git status` gdje se nalazimo / na kojem branchu, da li se razlikuju tj. da li možemo nešto dodati

`git reset HEAD` vraća sve u working

`git reset HEAD name_file` vraća samo taj file u working a ostalo ostaje u Staging

`git checkout -- name_of_file` sve što smo promijenili vraćamo u verziju od master

`git diff` pokazuje nam šta smo promijenili u file-u
crvena boja stara verzija
zelena boja nova verzija

GIT TUTORIAL

change name: `git config --global user.name "ime"`

change email: `git config --global user.email "email_prijatelj"`

`git config --list` daje nam trenutnu konfiguraciju

`git help commit` 1260ci nam help od commit

`pwd` daje nam gdje se nalazimo

`cd ~` idemo u home folder

`git init` napravi/inicijalizira git projekat

`git add .` dodajemo sve što smo napravili u tom folderu. u master, znači tag foldera sve foldere

`git commit -m "Message"` kako komitamo i dodajemo poruku šta smo napravili

`git log` tko je šta radio i kada dodao

`git log --author "Name of author"` daje nam šta je sve tag autor dodao ili izmijenio

`git status` gdje se nalazimo / na kojem branchu, da li se razlikuju tj. da li možemo nešto dodati

`git reset HEAD` vraća sve u working

`git reset HEAD name_file` vraća samo taj file u working a ostalo ostaje u staging

`git checkout -- name_of_file` sve što smo promijenili vraćamo u verziju od master

`git diff` pokazuje nam šta smo promijenili u file-u
crvena boja starija verzija
zelena boja nova verzija

GIT TUTORIAL

```
git add some_file  
git diff
```

} onda ne samo vidjeti razliku
jer je file u stagingu nije
više u working folder

```
git diff --staged
```

onda koristimo ovu naredbu da
vidimo razliku između staging area
i repository

```
git mv name_of_file
```

ovako obrišemo file iz
working i staging ali nismo
iz repository

move/rename file

```
git add new_name_of_file  
git mv old_name_of_file
```

} ove dvije komande uradimo
da promijenimo ime i
onda radimo commit

```
git mv new_name_of_file old_name_of_file
```

promijena
na isti način promijenimo
ime

```
git mv name_of_file folder/new_name
```

stavljam file

u neki folder, ako bude prazno
new_name onda ostaje isto
ime

```
git commit -m "Message to commit"
```

ovdje radimo
obaje i adding i commit
odmah, koriste kod sve stavljam

GIT TUTORIAL

kako se vratiti na neki commit???

git checkout broj-commita -- index.html

vratimo neki

file iz commita

repositorija

poslije toga

radimo commit

git remote add name_of_repo adresa_na_github

sa ovom komandom pravimo repo i
povezujemo sa github

git remote radimo provjeru koje repo imamo

git push name_of_repo master
|
koji branch

git pull name_of_repo

git branch ime_novog_brancha - ovako pravimo novi branch

git branch - listu svi branchova

git checkout ime_brancha - kako se prebacimo na novi
branch

git merge name_of_branch - kako vučemo u master
neki branch