



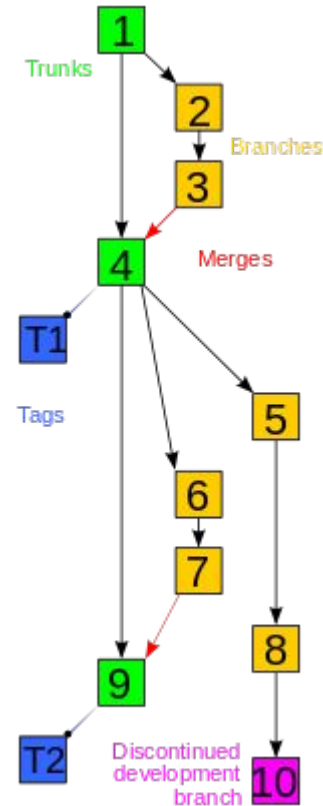
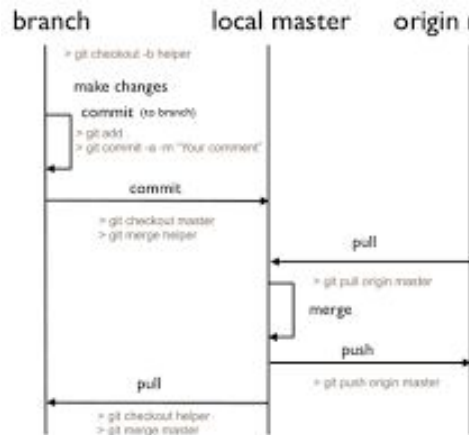
Git introduction



git

Git

- Osmislio ga je Linus Torvalds, razvoj je krenuo 2005
- Version control system (SVN, Mercurial)
 - Kompletna istorija celog programa





Git clone

To ste već uradili :D



Git status

- Prikazuje informacije o tome kako ti trenutno stoji lokalni repository tj koji fajlovi su ti izmenjeni, koji nisu dodati u VCS, koji su spremni za commit



Git add

- Dodaje fajlove u pred commit stanje



Git commit

- Završio si jednu određenu celinu i želiš da je sačuvaš
- Pišeš commit poruku kako bi bilo zabeleženo kakva je to celina



Git push

- Šalješ svoje commitove na remote repository (GitHub, GitLab)



Git pull

- Povlači sve izmene sa repository-ja i ubacuje ih u lokalni repository
- U stvari izvršava 2 komande *git fetch* i *git merge FETCH_HEAD*



Git merge

- Spajanje dve ili više grana u jednu



Git diff

- Prikazuje razliku između grana
- Najviše se koristi pre merge-a i tokom merge-a ako je došlo do conflict-a



Git branch

- Koristi se za pravljenje, listanje ili brisanje branch-eva (grana)
 - `git branch <new branch>`
 - `git branch --list`
 - `git checkout <created branch>`
- Branch-evi mogu biti lokalni ili iz glavnog repository-ja (kada se uradi `git pull` ili `git fetch` povuku se reference na glavne repository-je koji isto mogu da se checkout-uju lokalno)



Git stash

- Koristi se u slučajevima kada radimo na nečemu i treba da se prebacimo na nešto drugo a ovo još nije spremno da bude commit-ovano
 - `git stash push` / `git stash` (stash-ovanje podataka)
 - `git stash list` (da izlistamo stash)
 - `git stash pop` (da izvadimo prvi stash)
 - `git stash apply <stash>` (da izvadimo određeni stash)
 - `git stash show <stash>` (da pregledamo određeni stash)
 - `git stash drop <stash>` (da izbrišemo određeni stash)



Git log

- Prikazuje commit log-ove
 - `git log` prikazuje log-ove za ceo projekat
 - `git log <filename>` pokazuje log-ove za određeni fajl



Git reset

- git reset [<mode>] [<commit>]
 - **soft** Reset-uje branch na određeni commit ali svi fajlovi su sačuvani i stavljeni u stanje *ready to be committed*
 - **mixed** Reset-uje branch na određeni commit ali svi fajlovi su sačuvani i nisu u stanju *ready to be committed*
 - **hard** Reset-uje branch na određeni commit i sve izmene koje su se desile nakon tog commit-a briše
 - **merge**
 - **keep**



Git blame

- `git blame <file>` vraća za svaku liniju koda ko ju je poslednji menjao, jako je korisno iz razloga da možeš da pitaš tu osobu šta je hteo da uradi sa tim kodom ako je taj kod nejasan



Git resources

Ukoliko želite da istražite i nučite više o git-u to možete na ovim linkovim:

- <https://git-scm.com/doc> dokumentacija sa detaljnim opisom svih komandi
- <https://www.atlassian.com/git> git tutorial sa advanced workflow-om i nekim stvarima sa kojima se susrećete u realnim situacijama
- <https://www.pluralsight.com/courses/code-school-git-real-2> Git advanced tutorial

Pitanja?