

Zadatak za LAB-0 i rješenje

1. U nekom direktoriju s kodom („radnoj inačici“ git repozitorija inače pohranjenog na poslužitelju, na adresi **git@kodovi.hr:program.git**), kojeg Pero Perić već duže koristi (ne treba ga inicijalizirati i sl.), napravljene su nove promjene:

- promijenjene su datoteke **main.c**, **lib.c**, **lib.h**
- dodane su nove datoteke **lib2.c** i **lib2.h**

Kojim naredbama će se te promjene dodati u repozitorij i postaviti na poslužitelj?

```
git add *
git commit -m "azurirano, dodano"
git push
```

2. Kasnije, Pero je napravio dodatnu promjenu u **main.c**, ali sad želi i obrisati datoteke **lib.c** i **lib.h** (maknuti ih iz repozitorija, lokalnog i na poslužitelju). Kojim će naredbama to napraviti?

```
git rm lib.c lib.h      (ili samo rm, ali onda iduća naredba mora biti git add *)
git add main.c
git commit -m "jos neke promjene"
git push
```

3. (Još kasnije) Pero je podijelio pristup repozitoriju Ivi Ivić – dao joj prava pristupa i promjene koda preko sučelja git poslužitelja, te mailom joj poslao adresu git repozitorija. Što sve treba Iva napraviti da bi dohvatila repozitorij te u njega dodala nove datoteke **lib3.c** i **lib3.h**, uz dodatnu promjenu postojeće datoteke **main.c** (i naravno postavila te promjene lokalno i na udaljeni repozitorij)? Pretpostaviti da je globalno Iva već postavila sve ime i mail te ne treba to ponovno postavljati za taj repozitorij (ne trebaju naredbe **git config user.name** i **.email**).

```
git clone git@kodovi.hr:program.git
#napravi/kopiraj datoteke lab3.* u program
cd program
git add *
git commit -m "Nadogradnja na v3"
git push
```

4. Nakon što je Iva promjenu napravila i postavila u repozitorij, što treba Pero napraviti da bi ta promjena bila vidljiva i kod njega (dohvatiti nove i ažurirane datoteke)?

```
git pull
```

5. (Bonus pitanje) Kako Pero može vidjeti promjene koje je Iva napravila (što je i gdje promijenila)? Nakon što je Pero napravio **git pull** Ivine promjene su ugrađene i u radnu inačicu. Stoga naredba **git diff** neće pokazati te razlike. Potrebno je s **git log** prikazati povijest promjena te za predzadnju (koju je napravio Pero) kopirati **<id>** promjene i onda napraviti **git diff <id>**. Npr.:

```
$ git log
commit 11493daeca08df228639e518404c2cac267c1309 (HEAD -> master, origin/master, origin/HEAD)
Author: Iva Ivic <iva.ivic@...
    Nadogradnja na v3
commit c501c31fae3eb180a3b47fa218d1fbcf200f2af6
Author: Pero Peric <pero.peric@...
    jos neke promjene
```

```
$ git diff c501c31fae3eb180a3b47fa218d1fbcf200f2af6
```

Uspoređuje se trenutno stanje (master) koje uključuje Ivine promjene i ono što je zadnje Pero napravio.

Uz naprednije poznavanje gita (koje se ne traži): **git diff HEAD~1** (~1 => jedan korak u povijest prije sada=HEAD) ili nekom drugom složenijom metodom.