

Git i GitHub osnove

Zasto GIT

1. source version control
2. collaboration

1. Imali smo kod pre tri dana I sada je on napredovao sa novim fajlovima tzv. druga faza. Bilo bi super da mozemo da se krećemo kroz te faze e to će nam omogućiti softver git.
2. Jednostavno da možemo na istom fajlu da radimo mi , neki kolega , neki drugi koder koji želi da učestvuje u našem kodu. Kako će se svi ti ljudi čuti, zajedno učestvovati. Možda je taj neko prepravio nešto što nije smeo i tu nastaje zbrka i zbog toga je nastao GIT.

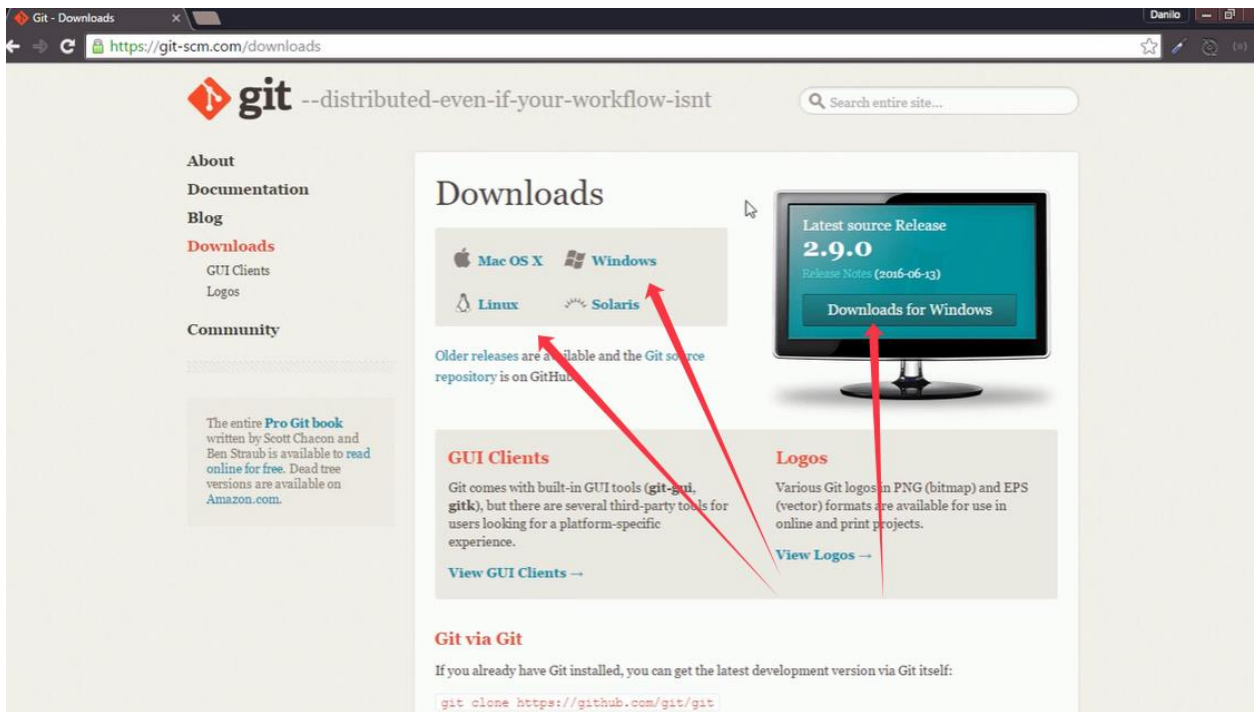
Da razjasnimo

GIT = software
GITHUB = website

repository
commit
add
push
pull

clone
init
remote

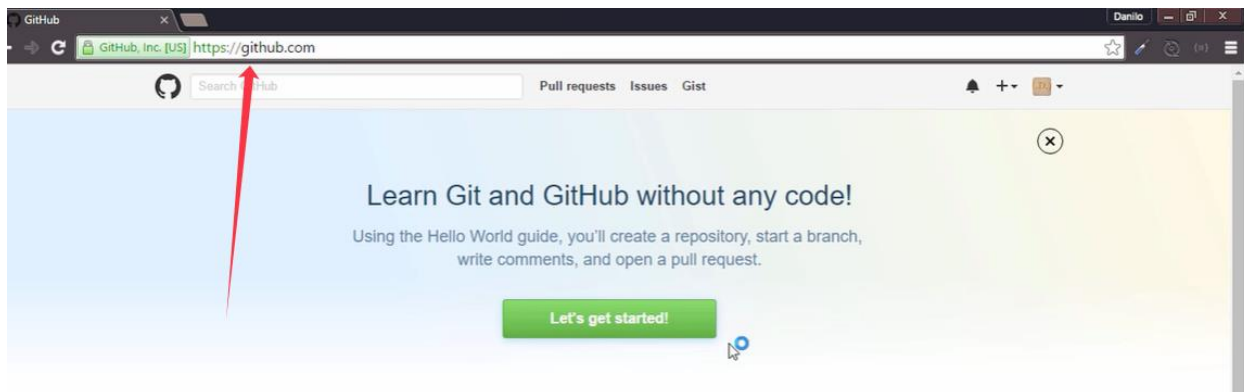
Neki pojmovi



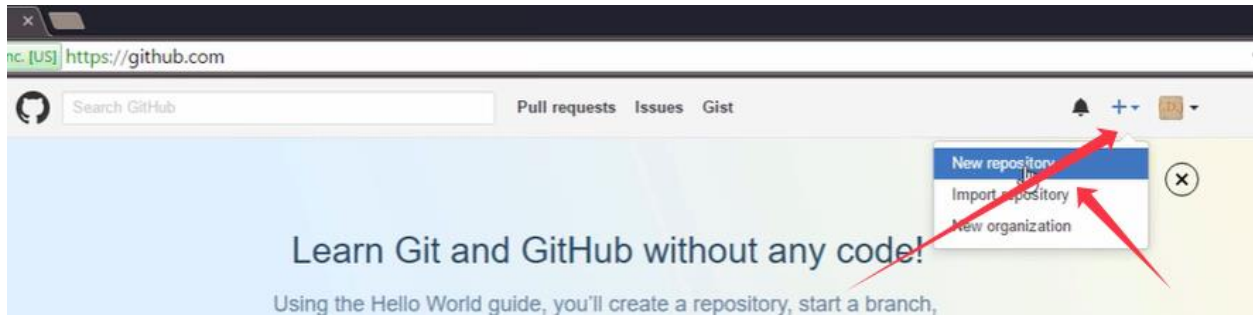
Kada to instaliramo možemo ga probati i u terminalu
Ima i svoj terminala kada se instalira , mi smo pogledali koju trenutnu verziju gita imamo

```
MINGW64:/c/Users/KORISNIK
Korisnik@Djordje80 MINGW64 ~ (master)
$ git --version
git version 2.40.0.windows.1
Korisnik@Djordje80 MINGW64 ~ (master)
$ |
```

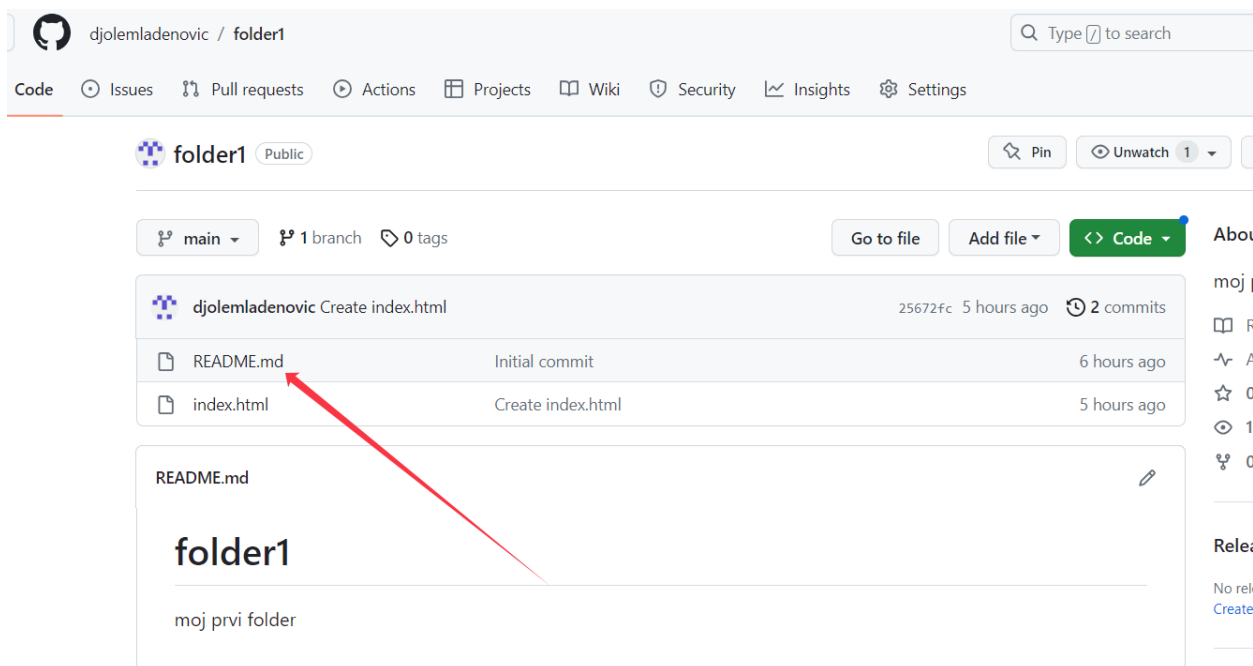
Naravno on može raditi i u drugim terminalima Linux, Windows.
Otiđimo sad na Sajt github



Registrujmo se na sajt i krenimo od početka . Hajde da napravimo naš prvi **repository**



Taj Repository nije ništa drugo već neki folder. Kao da je web sajt hostig našeg foldera , kao da imamo neki naš računar gde stavljamo neke foldere.



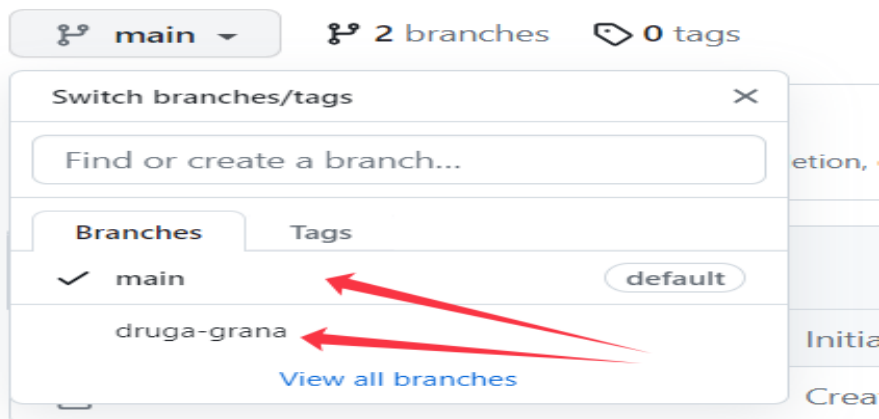
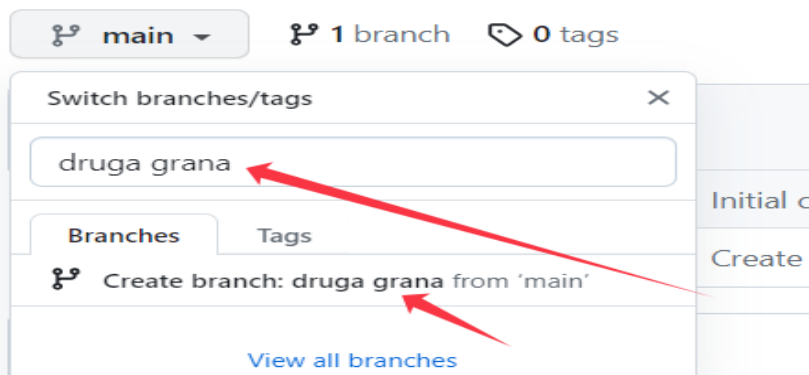
U tom folderu možete napraviti i novi fajl koji sam ja i napravio. U njemu smo napisali npr. ovako nešto



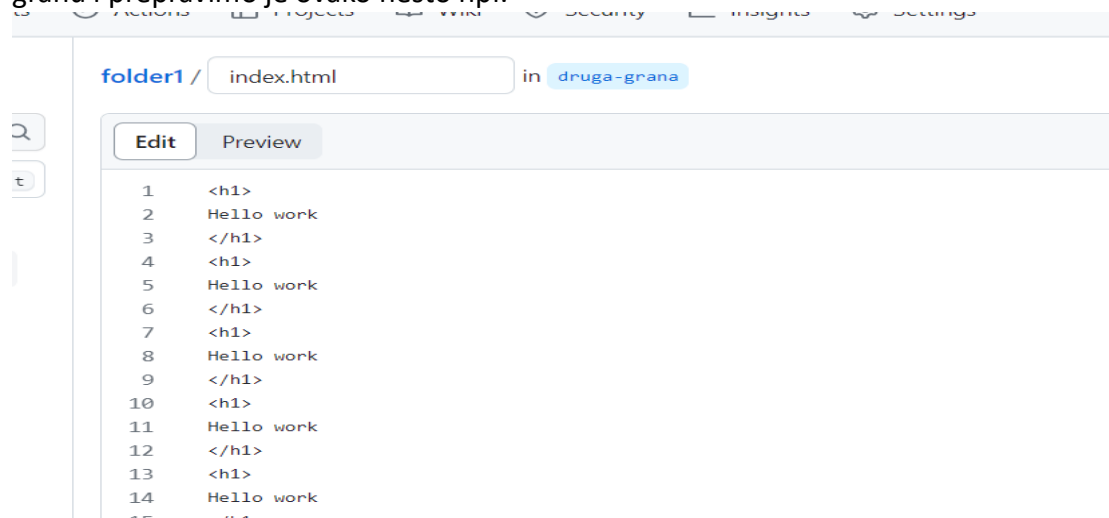
Ovde je mesto gde drugi ljudi mogu pogledati naš fajl, promeniti, učestvovati u radu...

Stavićemo ovde nešto što se zove Branch.

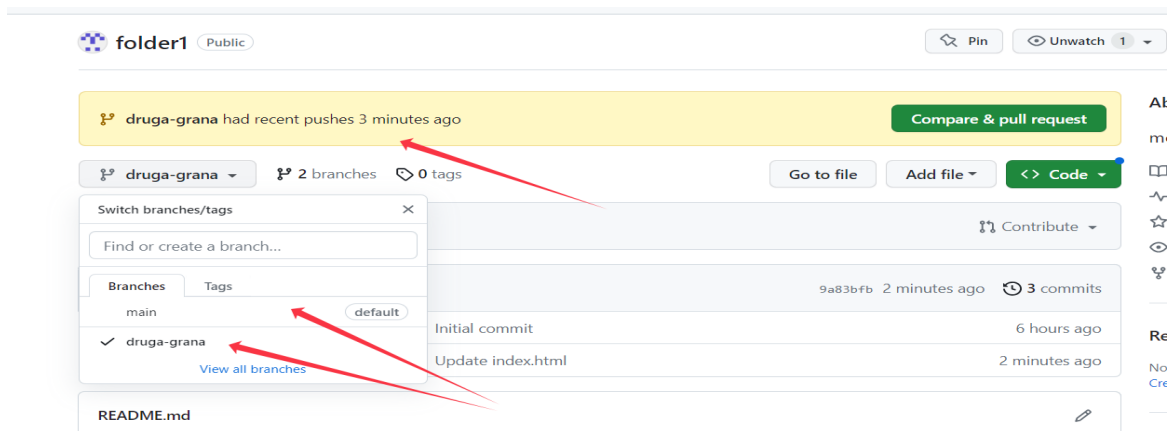
Branch su neke nove grane možemo slobodno reći. Hajmo da napravimo neku novu granu



Sta je sada ta druga grana kada je to kao što vidimo isti folder. Da bi videli razliku ućićemo u tu drugu granu i prepravimo je ovako nešto npr.

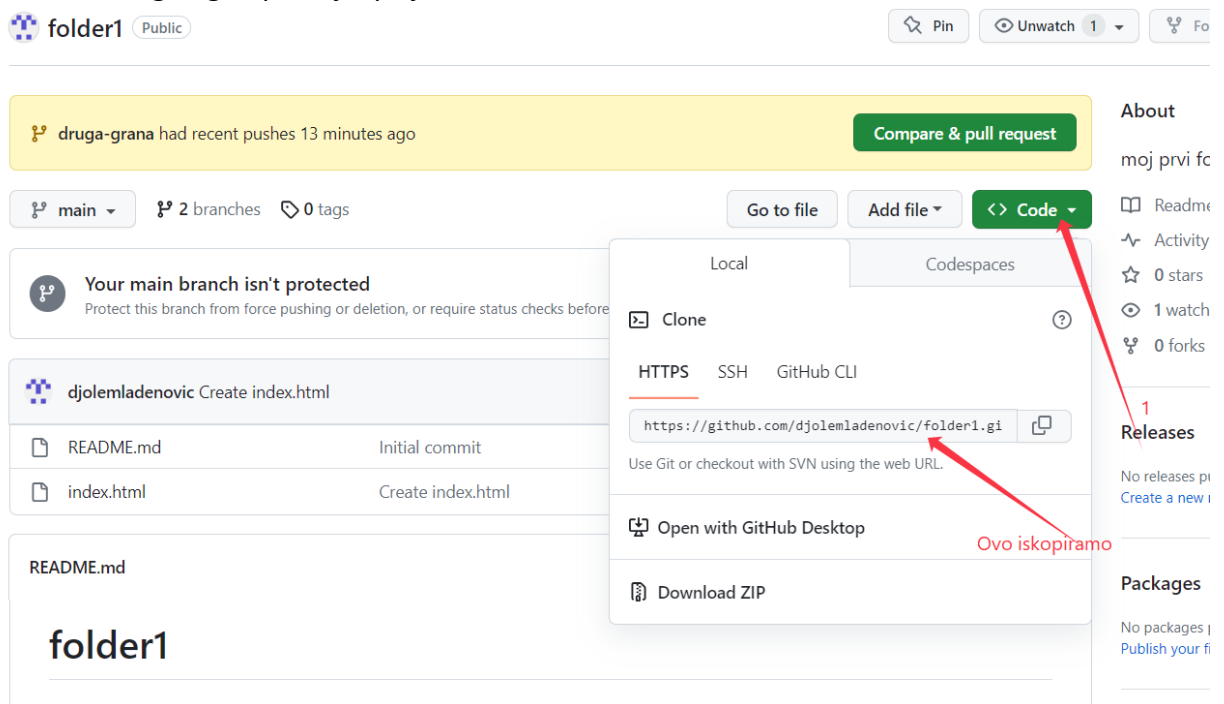


Ako se vratimo na početak videćemo da smo u drugoj grani u našem folderu, tačnije na drugom putu. Naravno ako se vratimo u našem main fajlu možemo proveriti da se ništa nije promenilo.



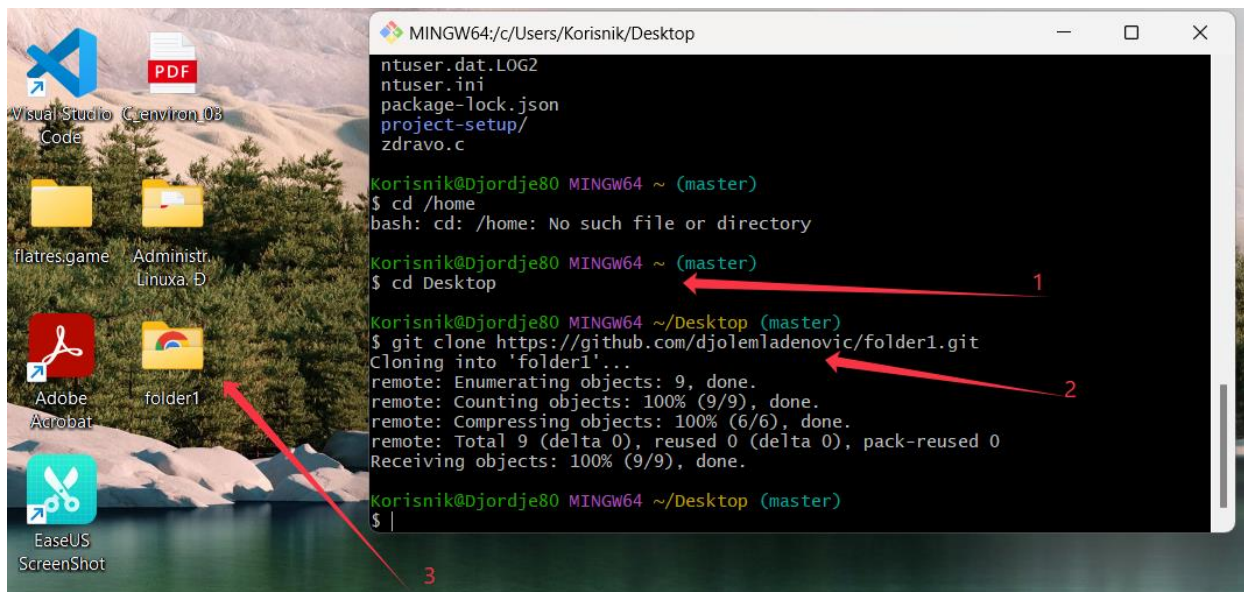
Znači da vidimo mi ovde imamo dve verzije našeg foldera.

A da vidimo kako to izgleda kada neko želi da radni na našem tom fajlu. Ša ćemo uraditi, pogledaćemo sa desne strane ili gde god postoji opcija CLONE



Uđimo u sad u neki od terminala i kucamo komandu

`git clone https://github.com/djolemladenovic/folder1.git` (ovo je neka moja adresa, vi kopirate svoju)



Šta ja radim, ja kloniram taj folder 1 sa githuba na svoj destop računar. Tačnije ja sam mu pokazao da on to iskopira na moj Dekstop. Što se može viditi iz priloženog.

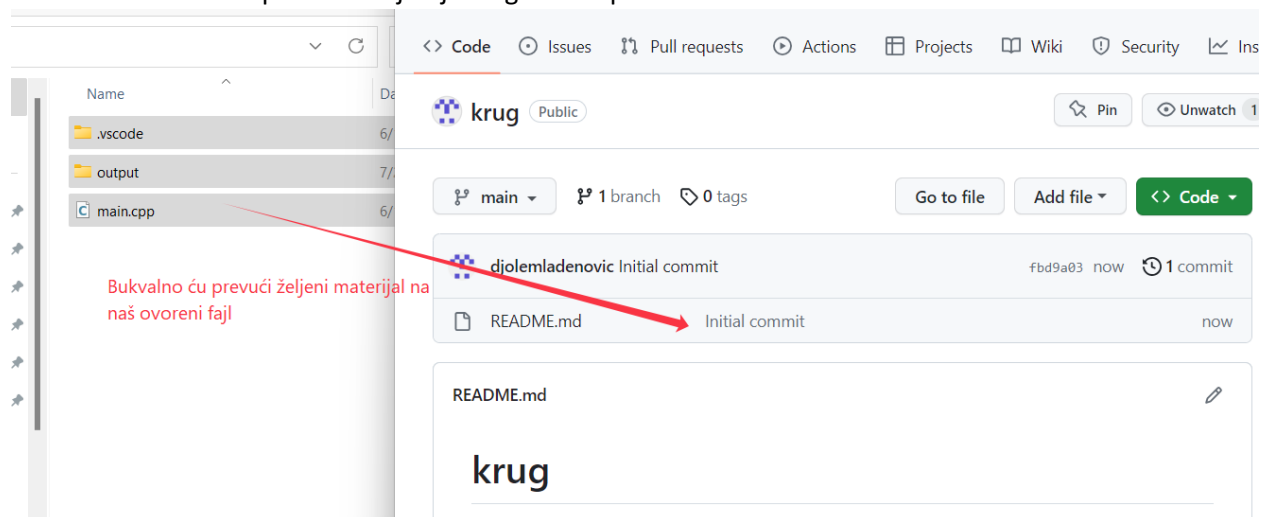
Kao što možemo videti kada uđemo u taj folder videćemo isti tekst (kod mene se otvorio u VSCodeu) koji mi možemo sada dorađivati ako želimo i kasnije vratiti na Git Hub.

Ajde da vidimo kako najprostije da postavimo neki program na GitHub.

Napravimo novi Reposotory ja ću staviti neki moj program pisan u C-u vezan za računanje prečnika kruga.

Sada ćemo bukvalno da uzmemo željeni fajl i da prenesemo na naš novi fajl na githubu.

Ne moramo čak ni kopirati dovoljno je da ga samo prenesemo.



Potvrdite samo da to želite i vaš fajl je ispunjen željenim sadržajem.

Evo našeg programa

```
krug / main.cpp.c  
djolemladenovic Add files via upload a209bcc · now History  
9 lines (9 loc) · 162 Bytes  
Code Blame Raw Copy Download Edit  
1 #include <stdio.h>  
2 int main()  
3 {  
4 float r, P;  
5 printf("Uneti poluprecnik kruga:");  
6 scanf_s("%f", &r);  
7 P = r * r*3.14;  
8 printf("Povrsina kruga je %.3f.", P);  
9 }
```

Otvorimo sada na nekom drugom nalogu GitHub da vidimo šta možemo.

Ulogovao sam se sada sa drugog naloga i u pretrazi sam pronašao moj git nalog na malopredašnjem nalogu. Naravno ja mogu da kloniram ili skinem neki od programa koji mi se sviđa ali postoji još jedna zanimljiva opcija. Ta opcija se zove Fork (viljuška) ona upravo to i radi . Dovoljno je da ,, bocnem,, ono od ponuđenih stvari koji mi se sviđa i oni će se automatski naći na mom nalogu.

github.com/djolemladenovic/krug

Code Issues Pull requests Actions Projects Security Insights

krug Public Watch 1 Fork 0 Star 0

main 1 branch 0 tags Go to file Add file Code

File	Commit	Time
README.md	Initial commit	11 minutes ago
main.cpp.c	Add files via upload	8 minutes ago

README.md

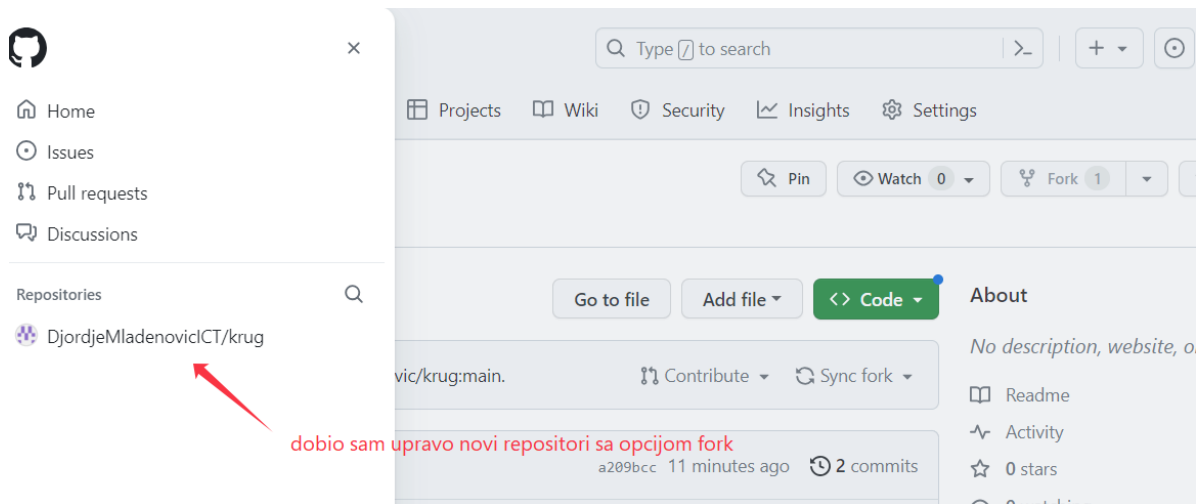
krug

About
No description, website, or topics provided.

- Readme
- Activity
- 0 stars
- 1 watching
- 0 forks
- Report repository

Releases

bocnem
drugi nalog

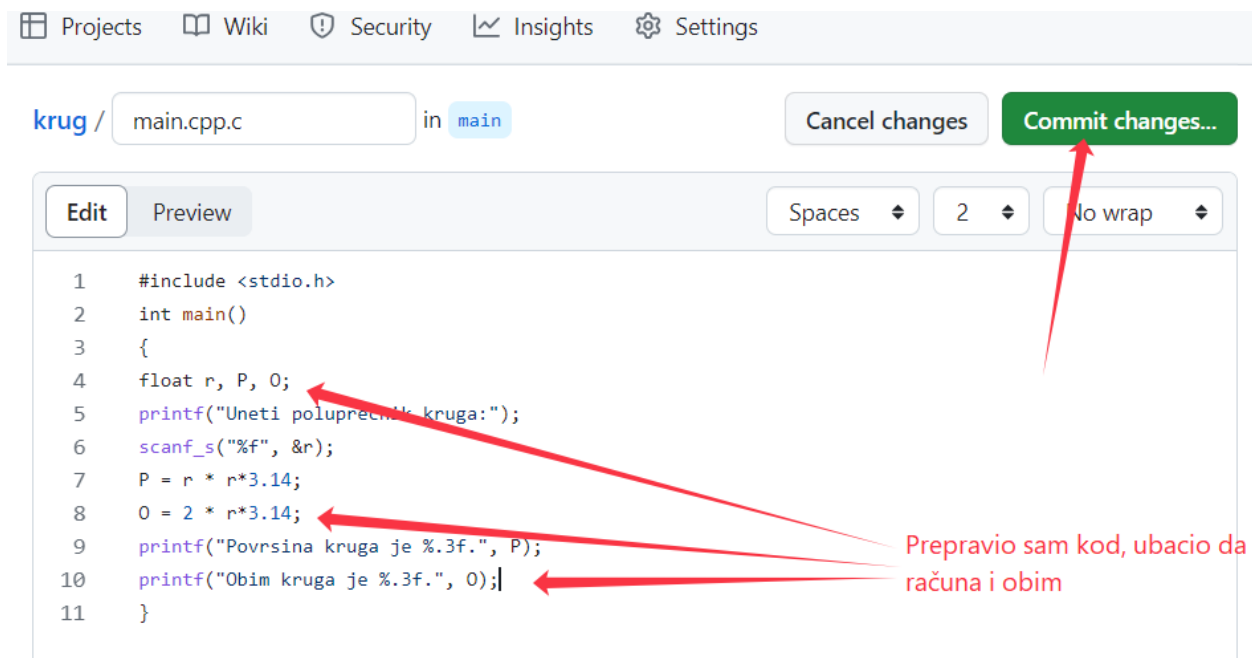


Sada ja na mom drugom nalogu gde sam fork-ovao mogu raditi sa tim programom, materijalom šta god želim i to neće uticati na moj originalni nalog gde se program nalazi. Ali šta ako bi želeo da nešto prepravi i ukažem da je ovako bolje. I to je moguće...

Bitno je skrenuti pažnju da možemo videti ko je sve pogledao moj materijal , ko je forkovao i sl.

Ako želimo da nešto prepravimo i ukažemo na učinjeno. Mi pravimo **novi pull request** .

Ajde da prepravimo nešto u kodu



Ja mislim da je ovo mnogo dobra promena i želim da predložim te promene.

Sada to radimo preko opcije **New pull request**-

Showing 1 changed file with 4 additions and 2 deletions.

Split Unified

```
6 main.cpp.c
... @@ -1,9 +1,11 @@
1 1 #include <stdio.h>
2 2 int main()
3 3 {
4 4 - float r, P;
5 5 + float r, P, 0;
6 6 printf("Uneti poluprecnik kruga:");
7 7 scanf_s("%f", &r);
8 8 P = r * r*3.14;
9 9 + 0 = 2 * r*3.14;
10 10 - }
11 11 + printf("Obim kruga je %.3f.", 0);
11 11 + }
```

ovde se vide tačno stvari koje sam dodao, oduzeo sa opcijama +- (with arrows pointing to lines 4, 8, and 10)

0 comments on commit ffaa3e1

Lock conversation

Onda idemo na opciju

Comparing changes

Choose two branches to see what's changed or to start a new pull request. If you need to, you can also [compare across forks](#).

base repository: djolemladenovic/krug base: main head repository: DjordjeMladenovicICT/krug compare: main

✓ Able to merge. These branches can be automatically merged.

Discuss and review the changes in this comparison with others. [Learn about pull requests](#) **Create pull request**

1 commit 1 file changed 1 contributor

Commits on Aug 12, 2023

Update main.cpp.c Verified ffaa3e1

DjordjeMladenovicICT committed 9 minutes ago

base repository: djolemladenovic/krug base: main head repository: DjordjeMladenovicCT/krug compare: i

✓ **Able to merge.** These branches can be automatically merged.

Update main.cpp.c

Write Preview H B I \equiv <> \equiv \equiv @ \equiv \equiv \equiv \equiv

Računaj površinu i obim kruga, mislim da je tako bolje.

napišemo zašto smatramo da je to bolje kako sam ja uradio

Attach files by dragging & dropping, selecting or pasting them.

Allow edits by maintainers ? Create pull request

Remember, contributions to this repository should follow our [GitHub Community Guidelines](#).

I on će se automatski proslediti na nalog sa kog sam skinuo program i ukazati mu na učinjenu izmenu.

Ako se vlasniku sviđa moja izmena on će je prihvatiti i u njegovom originalnom kodu ostaće izmenjena izmena.

Latest changes nova notifikacija

Open DjordjeMladenov... wants to merge 1 commit into djo1emladenovic:main from DjordjeMladenovicICT:main

Conversation 0 Commits 1 Checks 0 Files changed 1 +4 -2

Changes from all commits File filter Conversations Jump to 0 / 1 files viewed Review in codespace Review changes

```
main.cpp.c
... @@ -1,9 +1,11 @@
1 #include <stdio.h>
2 int main()
3 {
4 - float r, P;
5   printf("Uneti poluprecnik kruga:");
6   scanf_s("%f", &r);
7   P = r * r*3.14;
8   printf("Povrsina kruga je %.3f.", P);
9 - }
10
11 + }
12 + float r, P, 0;
13 + printf("Uneti poluprecnik kruga:");
14 + scanf_s("%f", &r);
15 + P = * r*3.14;
16 + P = 2 * r*3.14;
17 + printf("Povrsina kruga je %.3f.", P);
18 + printf("Obim kruga je %.3f.", 0);
19 + }
```

možemo videti i uporediti šta je promenjeno

Update main.cpp.c #1

Open DjordjeMladenov... wants to merge 1 commit into djo1emladenovic:main from DjordjeMladenovicICT:main

Conversation 0 Commits 1 Checks 0 Files changed 1

Commits on Aug 12, 2023

Update main.cpp.c ... Verified

DjordjeMladenovicICT committed 14 minutes ago

Open

Update main.cpp.c #1

DjordjeMladenovicICT wants to merge 1 commit into `djolemladenovic:main` from Djord:



Update main.cpp.c

Add more commits by pushing to the `main` branch on `DjordjeMladenovicICT/krug`



Require approval from specific reviewers before merging

Branch protection rules ensure specific people approve pull requests before they're merged.



This branch has no conflicts with the base branch

Merging can be performed automatically.

Merge pull request

You can also [open this in GitHub Desktop](#) or view [command line instructions](#).



Write

Preview

H

B

I

≡

<>

🔗

☰

☰

☰

Leave a comment

ovde možemo prihvatiti izmena ako ih želimo

I sada ako uđemo u naš fajl `krug` možemo videtu izmenu koju smo izveli.

Sve ovo možemo naravno uraditi I preko git ili bilo kog terminala koji smo konfigurisali.

Pokažimo sada kako bi to radilo iz terminala.

Koristićemo par komandi

Git status (da vidimo gde se nalazi)

Git add pa neki fal ili git add . (dodaj mi sve promene koje sam napravio)

Git commit -m " neki teks koji zelim"

Git push (to stavlja pushuje materijal na nas reposotorij)

Git pull (ako se neka promena desila na samom serveru)

Da vidimo sve ovo kroz jedan primer


Napravimo novi repository na samom GitHub-u.

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Required fields are marked with an asterisk (*).

Owner *

 djolemladenovic ▾

Repository name *

openProject

✓ openProject is available

Great repository names are short and memorable. Need inspiration? How about [studious-octo-goggles](#) ?

Description (optional)

 Public

Anyone on the internet can see this repository. You choose who can commit.

 Private

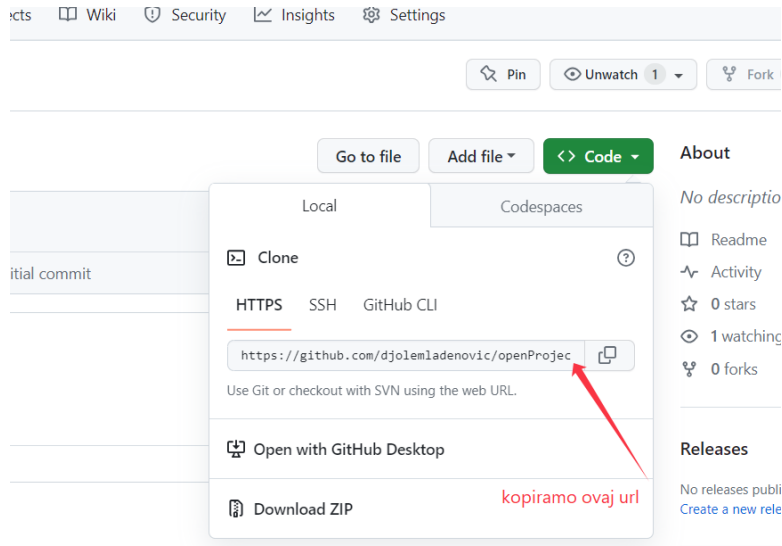
You choose who can see and commit to this repository.

Initialize this repository with:

Add a README file

This is where you can write a long description for your project. [Learn more about READMEs.](#)

Ajmo sada da cloniramo na nas racunar sa GitHuba preko terminara.



Otvorićemo sada naš terminal i kucamo mesto gde želimo da iskopiramo željeni sadržaj.

Ja ću ovako

cd Desktop

Git clone (ovde stavimo kopiranu adresu) npr. <https://github.com/djolemladenovic/openProject.git>

```
Korisnik@Djordje80 MINGW64 ~/Desktop (master)
$ git clone https://github.com/djolemladenovic/openProject.git
Cloning into 'openProject'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (3/3), done.
```

```
Korisnik@Djordje80 MINGW64 ~/Desktop (master)
$ ls
'Administr. Linuxa. D'/'
'Azure DevOps knjiga'/'
'C-vežbe Kg'/'
'C_environ_03.pdf
'DJordje - Chrome.lnk'*
'Docker Curriculum.lnk'*
'Git i GitHub'/'
'II5 - Opsti uspeh i vladanje.docx'
'Instalacija Apache2, MariaDB i PHP (LAMP).docx'
'Kontejneri i Docker'/'
'Kubernetes_Administrator_Roadmap_.pdf
'Linux -kg.docx'
'Virtualizacija/'
'Visual Studio Code.lnk'*
cfs.template.doc
desktop.ini
flatres.game/'
folder1/'
'linux etf niš'/'
openProject/'
predavanj1.ppt
proba/'
putty.exe*
'sheel skripta.docx'
'~$Docker.docx'
'~$nux -kg.docx'
'~$vops 1.docx'
'~WRL0003.tmp'
šablon.pptx
```

evo našeg fajla

```
Korisnik@Djordje80 MINGW64 ~/Desktop (master)
$ cd openProject

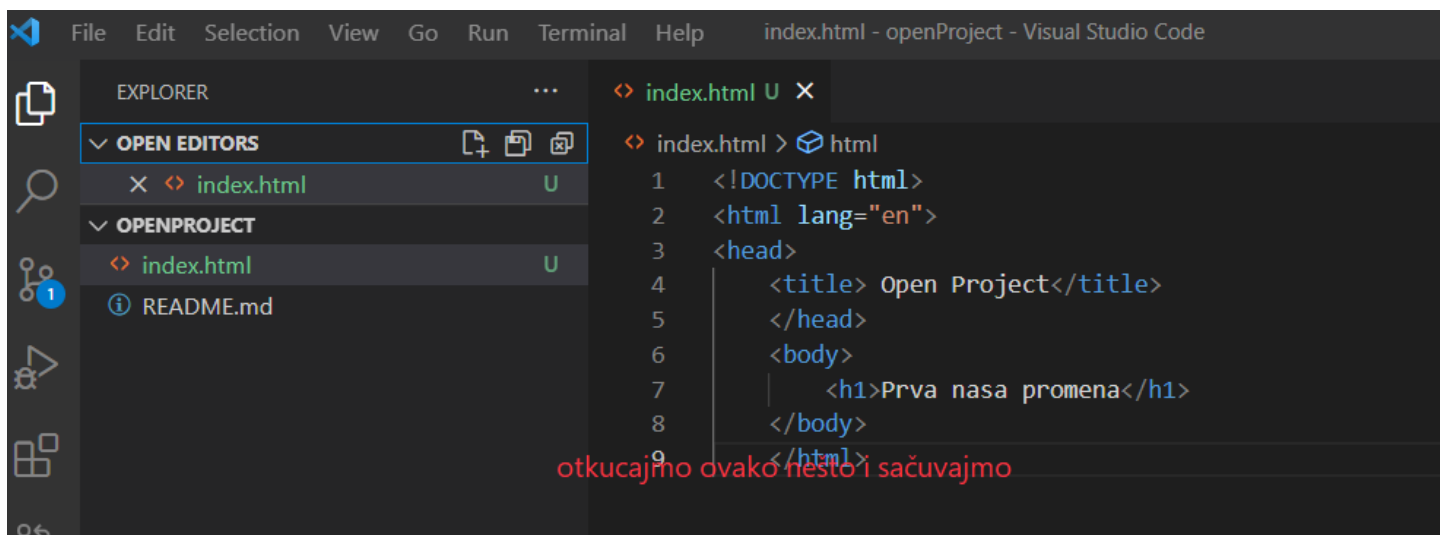
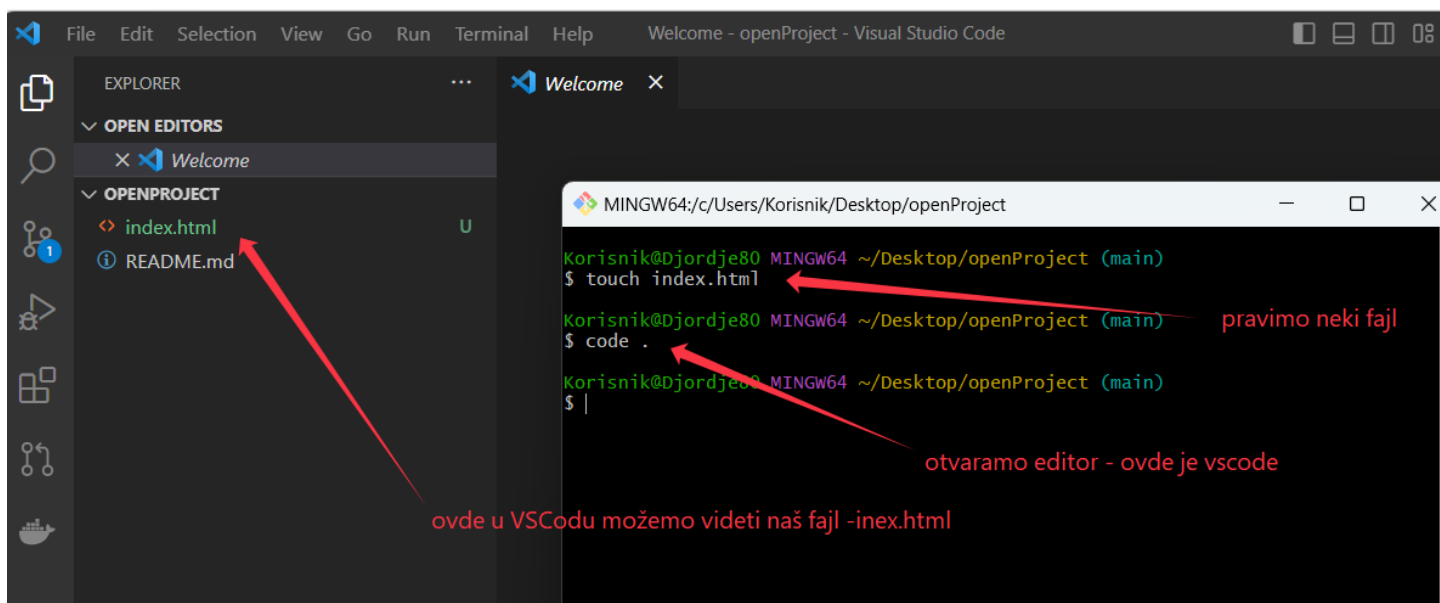
Korisnik@Djordje80 MINGW64 ~/Desktop/openProject (main)
$ ls
README.md

Korisnik@Djordje80 MINGW64 ~/Desktop/openProject (main)
$ ls -al
total 13
drwxr-xr-x 1 Korisnik 197121  0 Aug 13 17:22 ./
drwxr-xr-x 1 Korisnik 197121  0 Aug 13 17:22 ../
drwxr-xr-x 1 Korisnik 197121  0 Aug 13 17:22 .git/
-rw-r--r-- 1 Korisnik 197121 13 Aug 13 17:22 README.md

Korisnik@Djordje80 MINGW64 ~/Desktop/openProject (main)
$ |
```

Da vidimo šta je u tom folderu , možemo videti da je tu sakriven .git fajl . Tu se prate i snimaju sve naše promene

Ajde na napravimo jedan fajl , kucamo **touch index.html**



Ako se sada vratimo na naš openProject na GitHubu možemo videti da nemam nikakve promene i dalje na njemu .

Tu je samo README.md fajl. To je zato što nismo sinhronizovali fajlove.

E sada krećemo da kucamo malopredjašnje komande. Da vidimo prvo status

git status – (na master grani-brencu smo i to je ok ali vidimo da imamo neku promenu a to je upravo napravljeni index.html fajl) moramo da sinhronizujemo lokalni project i isti taj na GitHub-u.

To radimo sa **git add** index.html i na taj način ga uključujem i na naš projekat na GH-u

```
Korisnik@Djordje80 MINGW64 ~/Desktop/openProject (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

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

nothing added to commit but untracked files present (use "git add" to track)

Korisnik@Djordje80 MINGW64 ~/Desktop/openProject (main)
$ git add index.html

Korisnik@Djordje80 MINGW64 ~/Desktop/openProject (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
  new file:   index.html

Korisnik@Djordje80 MINGW64 ~/Desktop/openProject (main)
$ |
```

Sada vidimo da je taj fajl dodat, vidimo da je fajl sada zelene boje.

Idemo sada da potvrdimo promene tačnije commit

Kucamo **git commit -m " prva promena"** (stavite -m da bi napisali sta smo promenili) i videćemo sta je uradeno.

Samo da potvrdimo sve ovo je do sada uradjeno „LOKALNO,,

Ajde sada da to prosledimo na naš GH. Tj da push-ujemo.

```
$ git push
fatal: helper error (-1): User cancelled dialog.
error: unable to read askpass response from 'C:/Program Files/Git/mingw64/bin/git-askpass.exe'
Password for 'https://github.com':
petremote: No anonymous write access.
fatal: Authentication failed for 'https://github.com/djoemladenovic/djoemladenovic.git/'

Korisnik@Djordje80 MINGW64 ~/Desktop/openProject (main)
$ git config --global user.name "djoemladenovic"

Korisnik@Djordje80 MINGW64 ~/Desktop/openProject (main)
$ git config --global user.email djoemladenovic80@gmail.com
```

git push

ajmo da to odradimo preko terminala

`git config --global user.name "nas user na GitHubu"`

`git config --global user.email nasa.mejl.adresa.sa.kojom.smo.registr.na.gitHub-u.`

```
Korisnik@Djordje80 MINGW64 ~/Desktop/openProject (main)
$ git push
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 381 bytes | 190.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/djolemladenovic/openProject.git
 52ad774..08b1380 main -> main

Korisnik@Djordje80 MINGW64 ~/Desktop/openProject (main)
$ |
```

sada je sa push kom uradjena sinhronizacija

openProject Public Pin Unwatch 1

main 1 branch 0 tags Go to file Add file Code

djordje80 Prva promena		08b1380 14 minutes ago	2 commits
README.md	Initial commit		1 hour ago
index.html	Prva promena		14 minutes ago

README.md možemo videti da je uradjena sinhronizacija

openProject

Sa komandom `git config --list` možemo videti u silnim vezama koje imamo da smo i ovo povezali

```
Korisnik@Djordje80 MINGW64 ~/Desktop/openProject (main)
$ git config --list
diff.astextplain.textconv=astextplain
filter.lfs.clean=git-lfs clean -- %f
filter.lfs.smudge=git-lfs smudge -- %f
filter.lfs.process=git-lfs filter-process
filter.lfs.required=true
http.sslbackend=openssl
http.sslcainfo=C:/Program Files/Git/mingw64/etc/ssl/certs/ca-bundle.crt
core.autocrlf=true
core.fscache=true
core.symlinks=false
pull.rebase=false
credential.helper=manager
credential.https://dev.azure.com.usehttppath=true
init.defaultbranch=master
user.email=djordjemenovic80@gmail.com
user.name=djolemladenovic
core.repositoryformatversion=0
core.filemode=false
core.bare=false
core.logallrefupdates=true
core.symlinks=false
core.ignorecase=true
remote.origin.url=https://github.com/djolemladenovic/openProject.git
remote.origin.fetch=+refs/heads/*:refs/remotes/origin/*
branch.main.remote=origin
branch.main.merge=refs/heads/main
diff.astextplain.textconv=astextplain
filter.lfs.clean=git-lfs clean -- %f
filter.lfs.smudge=git-lfs smudge -- %f
filter.lfs.process=git-lfs filter-process
filter.lfs.required=true
http.sslbackend=openssl
http.sslcainfo=C:/Program Files/Git/mingw64/etc/ssl/certs/ca-bundle.crt
core.autocrlf=true
core.fscache=true
core.symlinks=false
pull.rebase=false
credential.helper=manager
credential.https://dev.azure.com.usehttppath=true
init.defaultbranch=master
user.email=djordjemenovic80@gmail.com
user.name=djolemladenovic
core.repositoryformatversion=0
core.filemode=false
core.bare=false
core.logallrefupdates=true
core.symlinks=false
core.ignorecase=true
remote.origin.url=https://github.com/djolemladenovic/openProject.git
remote.origin.fetch=+refs/heads/*:refs/remotes/origin/*
branch.main.remote=origin
branch.main.merge=refs/heads/main
```

Znači sada sa `git push` guram sve naše promene sa mog lokalnog računara na naš server.

Ajde da vidimo sledeću situaciju, neko je pogledao naš projekat i **Fork**-ovao .

Uradio je neku izmenu i predložio mi je da je izmenim u mom originalnom projektu.

Kao što vidimo i dalje nije sinhronizovana promena u lokala kod nas

ovde smo nešto promenili

Vratimo se u terminal i pogledajmo status

Git status i videćemo da tu nema promena, tačnije lokalno nema promena.

Zato ćemo koristiti komandu za „POVLAČENJE SVEGA SA SERVERA,, tačnije **git pull**

I da vidim koje su to promene učinjene.

```
MINGW64:/c/Users/Korisnik/Desktop/openProject
Korisnik@Djordje80 MINGW64 ~/Desktop/openProject (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

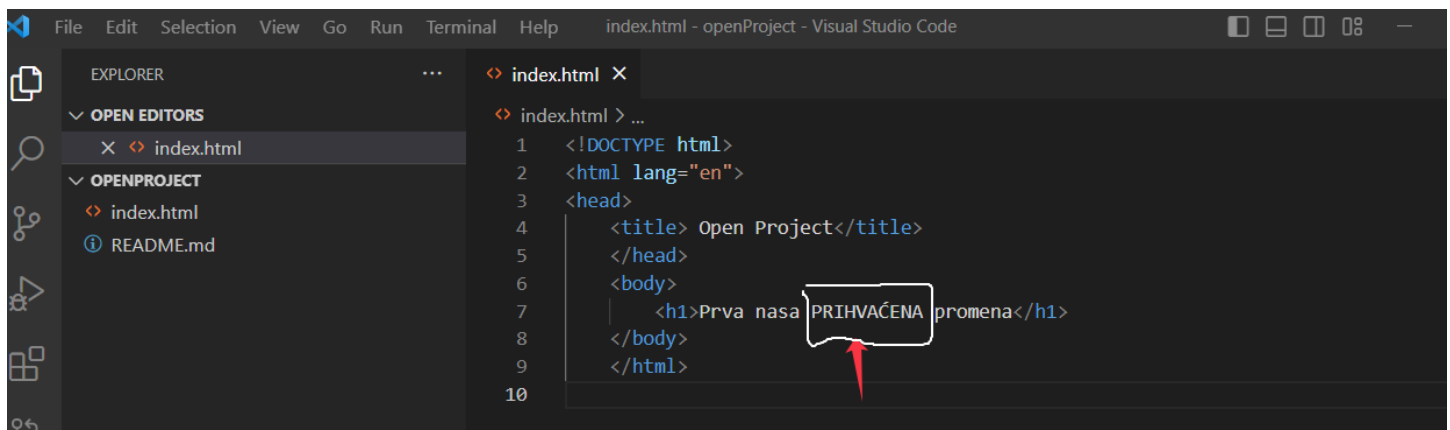
nothing to commit, working tree clean

Korisnik@Djordje80 MINGW64 ~/Desktop/openProject (main)
$ git pull
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 734 bytes | 104.00 KiB/s, done.
From https://github.com/djolemladenovic/openProject
   08b1380..e667891  main       -> origin/main
Updating 08b1380..e667891
Fast-forward
 index.html | 4 ++--
 1 file changed, 2 insertions(+), 2 deletions(-)

Korisnik@Djordje80 MINGW64 ~/Desktop/openProject (main)
$ |
```

ovde vidimo promene

Ako sada pogledamo naš VSCode videćemo da je prihvaćena promena.



The screenshot shows the Visual Studio Code interface with the Explorer view on the left and the Editor view on the right. The Explorer view shows the project structure with 'index.html' selected. The Editor view shows the content of 'index.html' with the following code:

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <title> Open Project</title>
5 </head>
6 <body>
7   <h1>Prva nasa PRIHVAĆENA promena</h1>
8 </body>
9 </html>
10
```

The word 'PRIHVAĆENA' is highlighted with a white box, and a red arrow points to it from below. The Explorer view shows the project structure with 'index.html' selected.