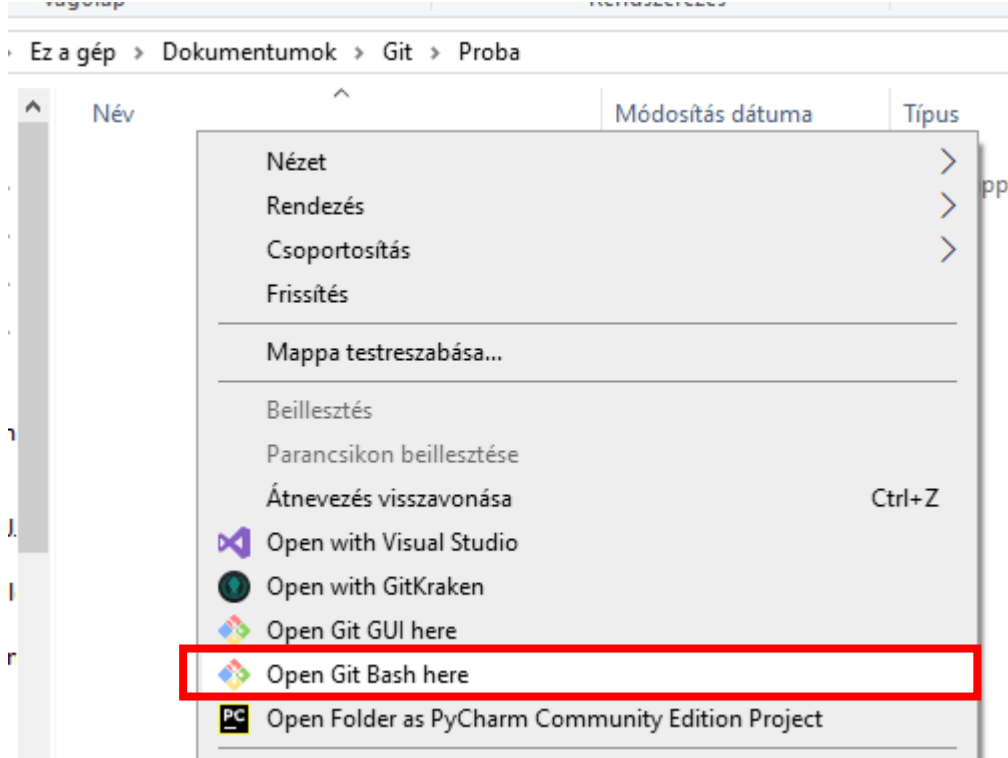


# GITHUB

Első lépések:

- Letöltjük a GIT-et - <https://git-scm.com/download/win>
- Telepítjük alapbeállításokkal, nem kell semmin sem módosítani első körben
- Hozzuk létre az első olyan mappánkat, amit LOKÁLIS REPOSITORY-nak fogunk használni. Ez a saját gépünkön van és innen másolódnak majd az internetre a fájlok. A mappa létrehozása után állítsuk is be lokális repository-nak. Ehhez menjünk a mappába, kattintsunk az egér jobb gombjával, majd a megjelenő menüből válasszuk ki a Git Bash lehetőséget



- A megjelenő ablakba adjuk ki a következő parancsot: `git init` mostantól ez a mappa egy lokális repository. Az itt végzett módosításokat figyelni fogja a Git!
- Első használatkor MINDENKÉPP meg kell adni a USER nevet és E-MAIL címet, hogy tudja a rendszer, kik is vagyunk és milyen e-mail címmel is szeretnénk majd feltölteni. Ennek a megadásnak különböző szintjei vannak, mi adjuk meg lokálisan, csak erre a mappára vonatkozóan a következő módon: `git config --local user.name "username"`, és az e-mail cím beállítása: `git config --local user.email username@gmail.com`
- Készen áll a mappánk a Git használatára. Hozunk benne létre egy fájlt, aminek a változásait majd feltöltjük a netre, követjük a verzióit.
- A `git status` paranccsal kérhetjük le a mappa jelenlegi állapotát. Mint látható, itt van egy -NEM TRACKELT fájl, őt szeretnénk feltölteni a netre és követni a változásait.

```
MINGW64:/c/Users/user/Documents/Git/Proba
user@DESKTOP-M8K550L MINGW64 ~/Documents/Git/Proba (master)
$ git status
On branch master

No commits yet

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

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

- A feltöltéshez először a STAGING-ba kell tölteni az ADD paranccsal. Itt a fájlok kiválasztásának különböző lehetőségei vannak. Egyik a fájl teljes neve pl.: „proba.txt”, a másik pl, az összes txt a „\*.txt” segítségével, végezetül az összes fájlra hivatkozhatunk a pont szombólummal: „”  
`git add elso.txt` vagy `git add *.txt` vagy `git add .`
- A Stagingból a lokális repositoryba a COMMIT paranccsal tölthetjük fel a fájlt. A commit parancs után megadhatjuk a feltöltendő fájl nevét vagy paraméter nélkül minden trackelt fájlt feltölt. A -m kapcsoló mögött megadott szöveg vonatkozik az aktuális verzióra, ebből a szövegből fogjuk tudni később, hogy ott hol is tartottunk a programunk fejlesztése során!  
`git commit -m "elso feltoltes"`

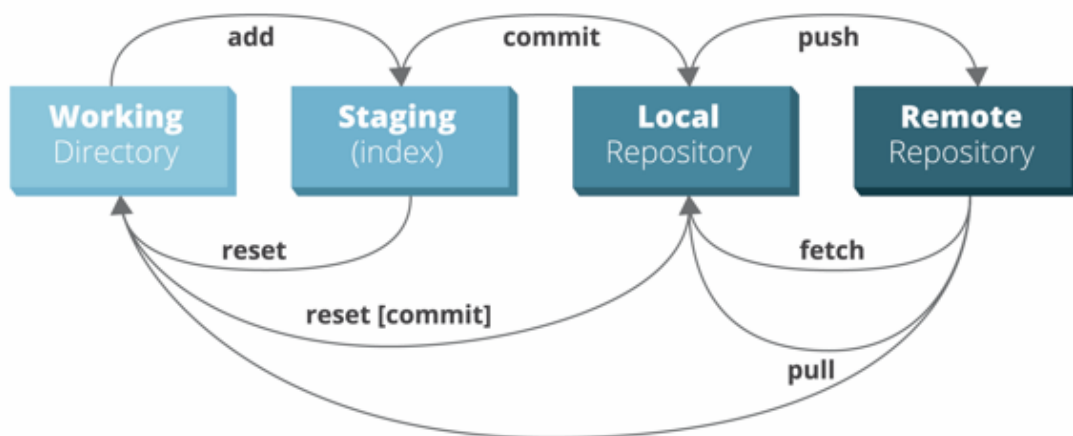
```
user@DESKTOP-M8K550L MINGW64 ~/Documents/Git/Proba (master)
$ git commit -m "elso feltoltes"
[master (root-commit) 9455dd3] elso feltoltes
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 elso.txt
```

És ezzel készen is vagyunk a lokális résszel. A továbbiakhoz létre kell hozni egy Github fiókot, abban egy üres repositoryt, ami majd a távoli repository lesz, ezt az órán mutatom meg. Ha ezzel készen vagyunk, akkor össze kell kötni a kettőt. Ehhez először tudnia kell egymásról a két repositorynak. Ennek formája a következő:

`git remote add origin https://github.com/janosij/13c.git`

ahol a link a távoli repository elérési útja. Már tudnak egymásról, de még nem történt meg a hitelesítés, ami ahhoz kell, hogy mi feltölthessünk adatokat!

- A fájlok feltöltése a `git push -u origin master` paranccsal történik. A parancs kiadása után el kell végezni a hitelesítést is!!! Ezzel kész a feltöltés
- Ha egy másik számítógépen szeretnénk folytatni a munkát, akkor a távoli repository tartalmát le kell tölteni a lokális gépre és így mindig az AKTUÁLIS ÁLLAPOTOT LÁTJUK, ez az egyik értelme az egésznek! A letöltés a CLONE paranccsal történik. Kell hozzá a távoli repository elérési útja is!
- `git clone https://github.com/janosij/13c.git`



`git pull origin master --allow-unrelated-histories` //szinkronizáljuk a két repót