

BEVEZETÉS A GIT-BE

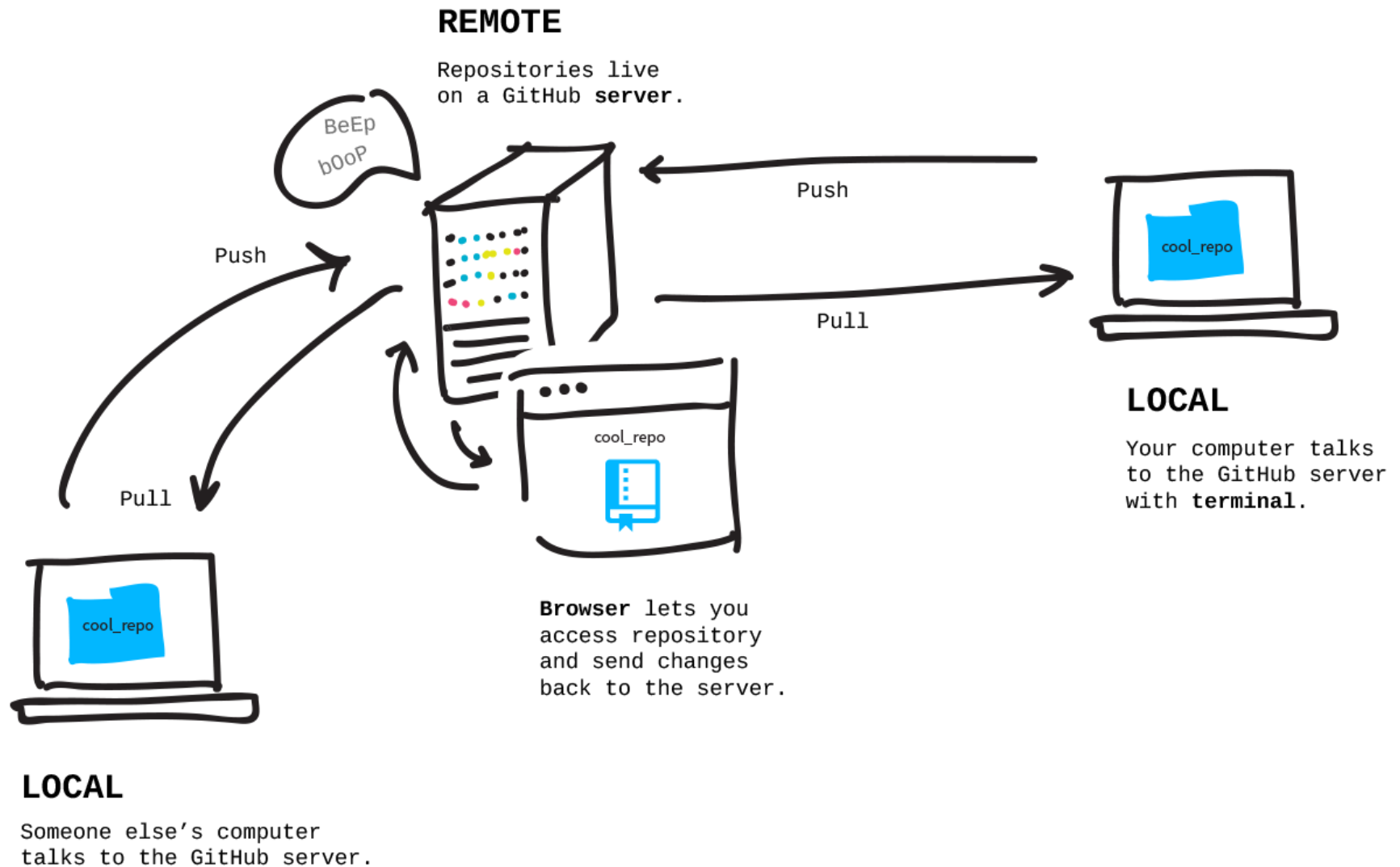
Mai nap programja

1. Git áttekintés
2. Git parancsok
3. Git + RStudio

Mi az a git?

- Verziókezelő szoftver
- Git segítségével több ember egyszerre tud dolgozni ugyanazon a kódbázison, majd könnyedén tudják szinkronizálni a munkájukat
- Git segítségével a saját munkádról készíthetsz biztonsági mentést
- Bármelyik régi kódverzióra visszaállhatsz és dolgozhatsz a projekten
- Kompatibilis bármelyik operációs rendszerrel

Overview



Szótár

Repository (repo)

File-ok, könyvtárak gyűjteménye, amit gittel verziózol. A repository tartalmazza a projekt minden valaha történt változtatását

Commit

Mentési lépés. Ez a mentés csak lokálisan, a te gépeden fog létezni mindaddig, amíg nem "pusholod" a változtatást.

Push

Szinkronizálja a commit-jaidat a távoli, közös repository-val (pl. ami githubon él). Egyszerre több commit is pusholható.

Pull

Távoli repository-ból az új változtatások másolása a lokális repoba.

Szótár

Github (<https://github.com/>)

A legnépszerűbb git repo hosztoló szolgáltatás (remote storage solution). Kezelhetsz jogosultságokat, értesítéseket, sőt akár webfelületen is editálhatsz kódot.

Clone:

A távoli, közös repository-t lemásolása a lokális gépedre (ha még nem volt ott előtte).

Szótár

Branch:

A git repo olyan, mint egy fa. A fő ága a `master` branch, de erről leágazhatnak egyéb branch-ek. Ekkor a history-juk közös, de az branch-eken különböző, egymástól független változtatások megjelenhetnek.

Használata: új feature fejlesztése

Merge:

Két branch egyesítése, a commitok összefésülése. PéldaÉ a te branch-ed merge-lese a master branch-be.

Vigyázz, mert conflict előfordulhat, ezeket manuálisan mindig fel kell oldani merge előtt.

DEMO

Git telepítése

Linux

Ubuntu: terminálban add ki ezt parancsot: `sudo apt-get install git-all` . Vagy keresd ki itt a megfelelő parancsot: <https://git-scm.com/download/linux>

Windows:

Installer letölthető innen: <https://git-scm.com/download/win>. Ha a letöltés befejeződött, futtasd le, hogy ténylegesen feltelepüljön.
Vagy: <https://gitforwindows.org/>

Mac:

terminálban géped be, hogy `git` . Ha nincs telepítve a program, akkor (elvileg) felugrik egy ablak, és megkérdezi, hogy szeretnéd-e telepíteni. (Ha ez nem működne:
<https://www.atlassian.com/git/tutorials/install-git>)

Github

1. Menj <https://github.com/>
2. Hozz létre egy accountot
3. Hozz létre egy új repo-t (ititalize with README)
4. Repo neve: r-ladies-git-tutorial

Példa: https://github.com/YOUR_ACCOUNT/r-ladies-git-tutorial

Git a terminálban

1. Nyiss meg egy terminált!
2. `mkdir rladies/gitrepos`
(Windows: `mkdir rladies\gitrepos`)
3. `cd rladies/gitrepos`
(Windows: `cd rladies\gitrepos`)
4. `git clone https://github.com/YOUR_ACCOUNT/r-ladies-git-tutorial.git`
5. `cd r-ladies-git-tutorial`
6. `git status`

Dolgozz!

1. Nyisd meg az Rstudio-t!
2. Új projekt létező forrásból
3. Hozz létre egy új R file-t (`helloworld.R`) az alábbi tartalommal:

```
# My first program in R Programming  
myString <- "Hello, World!"  
  
print ( myString)
```

4. Módosíts a README-n is!

Git a terminálban

1. Menj vissza a terminálba!
2. `git status`
3. `git add .gitignore README.md helloworld.R` (vagy `git add .`)
4. `git commit -m"Hello world"`
5. `git push`

Github

1. Menj githubra
2. Nézd meg a commitjaidat!
3. Editáld itt is a README-t a webes felületen
4. Nézd meg a commitjaidat újra!

Git a terminálban

1. Menj vissza a terminálba!
2. `git status`
3. `git pull`
4. `git status`

Git <3 RStudio

1. Menj be az RStudioba!

2. Készíts egy új R file-t (`simple_plot.R`):

```
# Define 2 vectors
cars <- c(1, 3, 6, 4, 9)
trucks <- c(2, 5, 4, 5, 12)

# Graph cars using a y axis that ranges from 0 to 12
plot(cars, type="o", col="blue", ylim=c(0,12))
# Graph trucks with red dashed line and square points
lines(trucks, type="o", pch=22, lty=2, col="red")
# Create a title with a red, bold/italic font
title(main="Autos", col.main="red", font.main=4)
# Save image
dev.copy(png, 'beautiful_plot.jpg')
dev.off()
```

3. Update-eld a Readme-t!

4. Keresd meg a `Tools > Version Control` menüpontot, majd commitolj, pusholj innen!

Branch

1. [Terminál] `git checkout -b very-important-change`
2. [RStudio] Változtass valamit a `simple_plot.R`, pl. a `red` színt `orange`-ra. Futtasd le a scriptet!
3. [RStudio] A `Tools > Version Control` menüpontban commitolj! (Tudsz pusholni?)
4. [Terminál] `git push`. Figyeld meg, mit ír ki, és kövesd az utasítást!
5. [Github] Menj fel a repod oldalára, nézd meg a branchet, nyiss Merge Requestet!

Néhány jótanács

- A git a kódot, szövegesfile-okat kezeli hatékonyan, képeket, bináris file-okat kevésbé
- Adatot legfeljebb akkor commitolj, ha kicsi és ha nyilvános adat.
- Figyelj, hogy public vagy private a repod! Githubon ingyenesen csak public repod lehet. Ha private repora lenne szükséged, érdemes a Bitbucket-et kipróbálni
- Ha bármikor elakadsz, a Google a barátod

Sources

<https://www.elegantthemes.com/blog/resources/git-and-github-a-beginners-guide-for-complete-newbies>

https://jennybc.github.io/2014-05-12-ubc/ubc-r/session03_git.html