

# GIT pro začátečníky

Martin Štulc

# Stažení

- Linux
  - `sudo apt-get install git git-gui gitk`
- Windows
  - <https://git-scm.com/download/win>
  - Při instalaci nechat zaštrnuté původní hodnoty

# Příprava

- Linux
  - cd do složky kde chceme vytvořit repozitář
  - Např. ~/repo/git
- Windows
  - Spustit Git Bash
    - fungují zde linuxové příkazy
  - Příkazem cd přejít do složky, kde chceme vytvořit repozitář
- Github
  - github.com – založit účet – volitelně

# Co je to GIT?

- Distribuovaný systém správy verzí
- Linus Torvalds
- Původně pro vývoj jádra Linuxu
- Původně - nízkoúrovňový základ pro vývoj různých systémů, správy verzí
- Postupně rozrostlo na kompletní systém pro správu revizí, použitelný bez doplňkových nástrojů
- Vývojář má k dispozici lokální kopii celé historie vývoje

# Předchůdci (alternativy)

- CVS - Concurrent Version System
  - Nedostatečná podpora UTF; file-by-file
  - Problematické větvení; neatomický commit
  - Problémy s binárními soubory
  - Integrace s GİTem - pluginy
- SVN – Apache Subversion
  - commit – revision
  - Proti gitu pomalé – pro velké projekty ne příliš vhodné
  - Integrace s gitem – nástroj *git-svn*
- BitKeeper, Monotone

# Pojmy

- Untracked, tracked, changed, staged, committed
- Commit
  - Předek, potomek, SHA-1 hash obsahu, message
- Revision (HEAD – ukazatel na poslední commit)
- Push
- Conflict
- Branch, tag
- Fetch
- Merge, rebase
- Pull (fetch + merge)
- Pull – rebase(fetch + rebase)

# Příprava

- Linux
  - cd do složky kde chceme vytvořit repozitář
  - Např. ~/repo/git
- Windows
  - Spustit Git Bash
    - fungují zde linuxové příkazy
  - Příkazem cd přejít do složky, kde chceme vytvořit repozitář
- Github
  - github.com – založit účet – volitelně

# Jméno a mail

- *git config --global --add user.email <mail>*
- *git config --global --add user.name <vaše jméno>*



# Založení(stažení) repository

- Vzdálené – např. Git hub
  - *git clone* <https://github.com/stulc/gitlecture.git>  
*gitLecture*
- *Lokální*
  - *mkdir*
  - *git init*
  - *git remote add origin <adresa>*

# Pull – Commit - Push

- *git status*
- *git pull origin master*
- Udělat změny
- *git status*
- *git add <soubor>*
- *git commit -m "first commit"*
- *git push origin master*

# GUI

- *git gui*
- *gitk* – grafická verze *git log*

# Větvení 1/2

- *git status*
- Vytvoření větve – *git branch <jméno>*
- List větví – *git branch*
  - Včetně vzdálených - *git branch --all*
  - *merged, no-merged*
- Smazání větve – *git branch -d <jméno>*
- Přepnutí do větve – *git checkout <jméno>*
- Vytvoření větve s přepnutím
  - *git checkout -b <jméno>*
- Pomocí *gitk*

# Větvení 2/2

- Push větve do nové vzdálené větve
  - *git push origin <lokalni>:<vzdalena>*
- Smazání vzdálené větve
  - *git push origin :<vzdalena>*
  - Vždy nejprve mažeme vzdálenou a teprve poté lokální větev
- *git checkout -b <vetev> origin/<vetev>*

# Merge x Rebase

- Změny v jedné větvi promítnout do druhé
- Merge
  - Sloučí větve do jedné – všechny conflicty najednou
  - V historii je vidět větvení a spojení
- Rebase
  - Mé commity jsou po jednom aplikovány na větev na kterou dělám rebase
  - Conflicty se musí řešit commit po commitu
  - Historie se tváří, že k větvení nedošlo

# Rebase

- *git rebase <vetev>*
  - Mé commity se aplikují na větev
  - Změněna bude aktuální větev <vetev> zůstane nedotčená
  - Pokud nastane konflikt, vyřeší se, soubor se dá do stavu připraven ke commitu
  - *git rebase --continue*
  - Zrušení rebase - *git rebase --abort*
- *git pull --rebase*

# Životní cyklus úkolu s merge

- *git pull*
- *git checkout -b ukol*
- Pracuji - commituji – každá věc, která jde pojmenovat = commit
- *git checkout master*
- *git pull*
- *git merge ukol*
- *git branch -d ukol*



# Životní cyklus úkolu s rebase

- ...
- *git fetch*
- *git rebase origin/master*
- *git checkout master*
- *git pull*
- *git merge ukol*
- *git branch -d ukol*

# Cherry-pick

- Aplikace jednoho konkrétního commitu z jedné větve na jinou větev
- Rebase provádí sérii cherry-picků
- Např. přenos funkcionality z hlavní větve do patchové
- Cherry-pick – nový commit
- gitk

# git reset

- Změna indexu větve na jinou revision
- *git reset --hard*
- soft, mixed, merge, keep
- <https://git-scm.com/docs/git-reset>
- *git checkout <soubor>*
- *git blame <soubor>*

# Stash

- Dočasné odložení necommitnutých změn
- Zásobník
- *git stash*
- *git stash list*
- *git stash pop (apply + drop)*
- *git stash branch <vetev>*

# Tag

- Značka k revision
- *git tag*
- *git tag -a v1.2 -m 'version 1.2'*
- *git push origin v1.2*
- *git checkout -b newbranch v1.2*

Dotazy