

Git cheatsheet

git init

Označí momentální složku jako **repozitář**

Vytvoří novou složku .git, ve které se nachází informace o historii a "větvích" repozitáře.

git status

Vypíše status repozitáře - které soubory jsou nové, které upraveny a které ve **staging area**

*Otestuje soubory vůči změnám, oproti staging area a oproti poslednímu **commitu** v historii repozitáře. Každý soubor který byl změněn se objeví s komentářem o jakou změnu se jedná (nový soubor, soubor smazán, soubor upraven, ...).*

git add [soubor ...]

Přidá soubor do **staging** oblasti

Označí soubory jako připravené ke commitu. Pokud v souboru provedeme další změny, neprojeví se do verze ve staging area (vyzkoušejte!)

git commit

Zapíše změny ve staging area do historie repozitáře

Každý commit (zápis v historii) má svůj hash, kterým je unikátně označen. Dále si nese informaci o datumu a autorovi commitu, můžeme tedy sledovat kdo a kdy provedl jaké změny.

Vlajky:

-m "[commit message]" - každý commit má také svůj popis (commit message). Použitím vlajky **-m** tento text specifikujeme. Pokud vlajku **-m** nepoužijeme, git otevře textový editor, ve kterém bude třeba commit message zadat. Příklad použití - **git commit -m "První commit"**
-a - přidá do commitu všechny změněné soubory (i ty, které nejsou ve staging area).

git clone [repozitář]

Vytvoří lokální kopii repozitáře a nastaví jej jako *remote*

*Pokud např. chceme pracovat na novém počítači a potřebujeme na něj dostat repozitář z GitHubu, použijeme právě **git clone** pro vytvoření lokální synchronizované kopie. Příklad použití: **git clone https://github.com/GS-Informatika/informatika-22-23-racket.git***

git pull

Stáhne nové commity z *remote* repozitáře

*Pokud se v remote nachází novější commity než lokálně, stáhne je a provede změny. Před provedením je vhodné udělat commit, pokud máme lokálně nějaké změny. Může dojít ke **konfliktu**.*

git push

Pošle naši lokální historii na *remote* repozitář

Tento příkaz vezme všechny změny, které jsme provedli a přidali do historie pomocí git commit a sdílí je na remote repozitář (např. GitHub), kde je pak mohou vidět i ostatní.

Konfigurace

git config --global user.name "UŽIVATELSKÉ JMÉNO"

git config --global user.email "email@email.com"

Git musí pro zápis do historie vědět kdo jste - těmito příkazy nastavíte své jméno a e-mail, které budou v historii zapsané.