

Co je to verzovací systém?

Co je to revize?

Na jaké typy projektů používáme verzovací systémy?

Proč bychom potřebovali verzovat projekt?

Je např. klasické kopírování složek verzovací systém?

Jak se liší styl zamykání souborů a slučování změn?

Na jaké typy projektů se hodí zamykání souborů a spojování změn?

Co to znamená, že je něco centralizované?

Co to znamená, že je něco decentralizované?

Co to znamená, že je něco distribuované?

Co je to Git?

Co byly cíle Gitu?

Jaké charakteristiky (co zaručuje) má Git?

Ukládá Git změny?

Jaké části má Git pro ukládání stavu souborů?

K čemu slouží working directory?

K čemu slouží staging area?

K čemu slouží Git directory?

Co je to repozitář?

Jak založíme nový repozitář?

Co se nachází ve složce .Git?

Co je to commit? Co obsahuje?

Jaké stavy souborů Git určuje?

Jak přesuneme soubor ze stavu untracked do stavu staged?

Jak zjistíme, jaké soubory jsou staged?

Jak se liší příkaz git status a git diff?

Čím se liší příkaz `git diff` a `git diff --staged`?

Co se stane, když soubor ve stavu `staged` upravíme?

Jak nové změny již `staged` souboru označíme jako `staged`?

Jako uložíme změny (tj. `staged` soubory)?

Co se stane, když při zavolání `Git commit` nepřidáme přepínač pro jméno?

Můžeme používat Git jen pro svůj projekt bez internetu?

Umí Git zaznamenat přesunutí souboru?

Jak v Gitu smažeme ze sledování soubor?

Jak v Gitu přidáme veškeré soubory ve všech složkách, včetně nových a odstraněných souborů do `stage area`?

K čemu v Gitu slouží soubor `.gitignore`?

Co udělá v souboru `.gitignore` zápis `"lib/twitter/sdk/lib/*"`?

Co udělá v souboru `.gitignore` zápis `"!test*"`?

Co udělá v souboru `.gitignore` zápis `"/build"`?

Co udělá v souboru `.gitignore` zápis `"logs/**/dev/*.log"`?

K čemu slouží příkaz `git log`?

Jaké znáte přepínače na příkazu `git log`?

Jak v příkazu `git log` omezíme počet vypsání záznamů např. na hodnotu 21?

Lze v příkazu `git log` zobrazit commity jen od jednoho autora? Jak?

Co dělá přepínač `--graph` na příkazu `git log`?

Jak odebereme označení souboru `staged`?

Jak resetujeme aktuálně změněný soubor tak, aby měl hodnotu jako v poslední revizi?

Můžeme s ostatními sdílet složku `.git`? Například pošlu kolegovi svojí složku, aby měl moje změny?

K čemu slouží vzdálené repozitáře?

Jak do lokálního repozitáře připojíme vzdálený?

Proč musíme pojmenovat vzdálený repozitář?

Můžu jednoduchými změnami souborů v working directory ovlivnit vzdálené repozitáře?

Jak stáhneme změny ze vzdáleného repozitáře?

Jak zklonujeme repozitář z internetu?

Jaký rozdíl je mezi Gitem a Gitovými službami (GitHub, GitLab, ...)?

Co je to větev v Gitu?

K čemu větve v Gitu slouží?

Na co větve ukazují?

Jaké speciální funkčnosti má hlavní větev (main / master)?

Jak vytvoříme novou lokální větev v repozitáři?

Jak pošleme / vytvoříme větev ve vzdáleném repozitáři?

Můžeme smazat větev?

Jak můžeme přejít do jiné větve?

Co se stane, když přepneme do nové větve, když máme nějaké rozpracované soubory?

Jak Git ví, v jaké větvi se aktuálně nacházíte?

K čemu slouží značka HEAD?

Co je to proces slučování větví v Gitu?

Jak spojíme aktuální větev s jinou?

Jak se spojení větví obvykle chová? Vytváří nový commit pokaždé?

Co se stane, když při spojení větví změníme v dvou revizích dva stejné soubory na stejné řádce?

Co je to merge conflict?

Jak se merge conflict řeší?

Když nastane merge conflict, kde najdeme veškeré soubory, které se musí opravit?

Jaký příkaz musíme po vyřešení merge conflictů zavolat?

K čemu slouží příkaz git branch -v?

Jak fungují vzdálené větve?

Jak odešleme změny z lokálního repozitáře na vzdálený?

Jak se můžeme vrátit k nějaké revizi (commitu) v Gitu?

Jaký je rozdíl mezi git fetch a git pull?

Může nastat merge conflict při git pull?

Jaké výhody má GitLab oproti GitHubu?

Co jsou to tzv. Issues v Gitových službách?

Co jsou to tzv. Projects v Gitových službách?

Co jsou to tzv. Pull requests v Gitových službách?

Co jsou to tzv. Releases v Gitových službách?

Jak funguje a co je to sémantické verzování?

Jak se liší MAJOR, MINOR a PATCH v sémantickém verzování?

Co jsou to metadata a předběžné verze v sémantickém verzování? Kam se zapisují a jak?

Proč se sémantické verzování používá?

Jaké věci nám sémantické verzování nabízí?

Co jsou to tzv. Milestones v Gitových službách?

Co jsou to tzv. Actions v Gitových službách?

Co můžeme dělat s veřejnými repozitáři?

Co je to tzv. forknutí repozitáře?

Jak můžeme přispívat do veřejného repozitáře? Co předtím musíme často udělat?

Co je to tzv. Wiki v Gitových službách?

K čemu slouží soubor README?

Každý commit by měl být funkční (resp. spustitelný). Proč?

Proč je důležité, aby zprávy commitů reprezentovali to, co se v něm stalo?

Proč by názvy commitů měli být konzistentní s projektem?

Jaké typy větví mají normálně obvyčejné projekty? K čemu se hodí?

Co je to Git klient?

Jaké Git klienty znáte?

Jaké mají Git klienti výhody?

Jaké mají Git klienti nevýhody?