

GitHub

Luděk Roleček

Použití materiálů

Toto dílo je licencováno pod

Creative Commons Attribution-NonCommercial-ShareAlike 4.0
International License



GitHub



- GitHub není Git
- **Git** je program, který běží u nás na počítače a sleduje změny. Historie projektu je uložena lokálně v projektu ve složce .git
- **GitHub** je internetová služba pro vzdálené uchovávání repozitářů. Tj. můžeme náš projekt včetně jeho historie poslat na GitHub a mít tak vzdálenou zálohu naší práce.

Zaregistruj se a vytvoř si účet:

<https://github.com/>

Nový repozitář

- Na GitHubu si vytvoříme nový repozitář.



Nový repozitář

Repository template

Start your repository with a template repository's contents.

No template ▾

Owner *



Irolecek ▾

/

Repository name *

pokus



Great repository names are short and memorable. Need inspiration? How about [congenial-couscous?](#)

Description (optional)



Public

Anyone on the internet can see this repository. You choose who can commit.



Private

You choose who can see and commit to this repository.

Initialize this repository with:

Skip this step if you're importing an existing repository.



Add a README file

This is where you can write a long description for your project. [Learn more.](#)



Add .gitignore

Choose which files not to track from a list of templates. [Learn more.](#)



Choose a license

A license tells others what they can and can't do with your code. [Learn more.](#)

Create repository



Vybereme jméno projektu bez mezer a českých znaků

Zvolíme, zda je naše repozitář veřejný (vidí ho kdokoliv) nebo privátní (vidím ho jenom já).

Další volby zatím buď necháme prázdné a nevybereme nic, nebo zaškrtneme volbu "Add README file".

README.md je soubor, který se obvykle přidává do repozitáře, aby se dovnitř napsal popis obsahu repozitáře (o jaký projekt se jedná, návod na použití, apod.) Tento soubor pak můžeme na Githubu vidět, když navštívíme repozitář.

Instrukce pro napojení na projekt



Quick setup — if you've done this kind of thing before

Set up in Desktop or HTTPS SSH `https://github.com/lrolecek/pokus.git`

Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

...or create a new repository on the command line

```
echo "# pokus" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/lrolecek/pokus.git
git push -u origin main
```

...or push an existing repository from the command line

```
git remote add origin https://github.com/lrolecek/pokus.git
git branch -M main
git push -u origin main
```

...or import code from another repository

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

Import code

Po vytvoření repozitáře se objeví instrukce pro napojení našeho lokálního projektu do tohoto vzdáleného repozitáře.

Adresa repozitáře. Použijeme např. pro klonování repozitáře na náš počítač.

Instrukce v případě, že ještě nemáme lokální git inicializovaný

Instrukce v případě, že už máme git inicializovaný a jenom ho chceme propojit s GitHubem

Vzdálený GitHub a lokální Git repozitář



Když máme na GitHubu vytvořený nový repozitář, máme dvě možnosti:

1. Naklonovat si tento GitHub repozitář na svůj počítač - na počítači se nám vytvoří lokální Git repozitář a oba budou navzájem propojené.
2. Pokud už máme inicializovaný lokální Git repozitář, můžeme ho sérií příkazů propojit s naším vytvořeným GitHub repozitářem.

Přihlášení

A screenshot of the GitHub Login dialog box. The dialog has a title bar with "GitHub Login" and a close button. The main content area displays the "GitHub Login" text, followed by two input fields: "Username or email" and "Password". Below the input fields are two buttons: "Login" (with a checkmark icon) and "Cancel" (with an 'X' icon). At the bottom, there are two links: "Don't have an account? Sign up" and "Forgot your password?".

Při prvním pokusu o napojení na vzdálený repozitář se nejspíš objeví login okno, kam musíme zadat svoje přihlašovací údaje na GitHub.

Klonování repozitáře

Klonování repozitáře

- Repozitář z GitHubu můžeme jednoduše naklonovat k sobě na počítač.
- Na disku se nám vytvoří složka s názvem repozitáře a z GitHubu se do ní přenesou všechny soubory (pokud tam nějaké jsou).
- Ve složce bude automaticky zapnutý Git.
- Oba repozitáře (lokální na počítači a vzdálený na GitHubu) budou automaticky propojené.

Klonování repozitáře

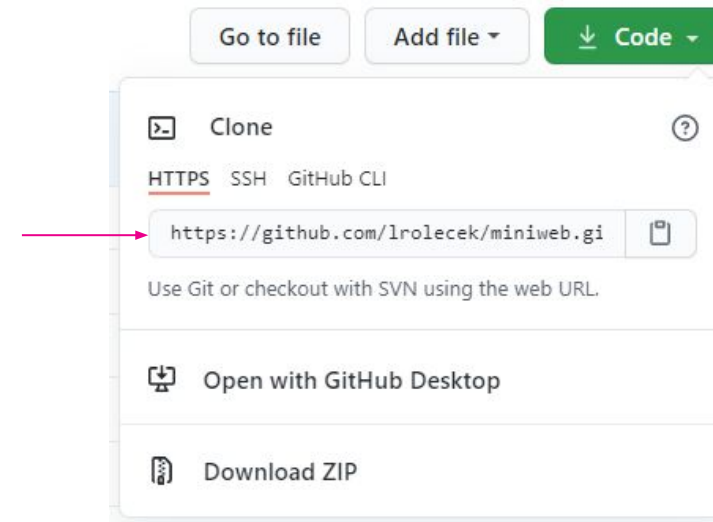


- Klonování můžeme použít jako nejrychlejší a nejjednodušší způsob, když začínáme nový projekt.

Na příkazové řádce:

```
git clone https://adresa.repa.git
```

Adresa každého repozitáře je uvedena na GitHubu buď na úvodní stránce prázdného repozitáře nebo po stisknutí na zelené tlačítko Code.



Odeslání změn na GitHub

Odeslání změn na GitHub

V projektu u sebe na počítači děláme změny a provádíme commity, jak potřebujeme.

Kdykoliv potom chceme odeslat lokální změny (commity) provedené v projektu na GitHub, tak provedeme příkaz:

Na příkazové řádce:

```
git push
```

Rekapitulace

Varianta 1 (složitější)

1

Vytvořím si na disku složku s projektem, otevřu příkazový řádek a provedu:

```
git init
```

V projektu musím mít alespoň jeden soubor a udělám alespoň jeden commit (viz bod 3), jinak nebude další postup fungovat.

2

Pokud chci projekt napojit na vzdálený GitHub repozitář, tak než mohu provádět **push**, musím:

1. Vytvořit na GitHubu nový repozitář
2. Postupovat podle instrukcí na GitHubu a nastavit tzv. remote origin a provést první push. Série příkazů obvykle vypadá nějak takto:

```
git remote add origin https://adresa.git  
git branch -M main  
git push -u origin main
```

3

Pokaždé, když provedu v projektu změny (obvykle nějaký ucelený blok), provedu commit a vytvořím tak záznam v historii:

```
git add .  
git commit -m "Popis"
```

4

Po provedení commitu (lokálně), ho mohu odeslat do vzdáleného repozitáře na GitHubu:

```
git push
```

Kroky 4 a 5 opakuji po každém bloku provedených změn a užívám si pocitu, že můj kód je v bezpečí na GitHubu, i kdybych si na notebook vylila čaj nebo mi vybuchnul.

Varianta 2 (jednodušší)

1

Pokud začínám nový projekt, je jednodušší **vytvořit nejprve nový repozitář na GitHubu.**

2

Repozitář si naklonuji k sobě na počítač. Mám ušetřenou práci, protože všechno bude nastaveno a navzájem propojeno.

```
git clone https://adresa.git
```

Klonování mi na disku vytvoří složku s názvem repozitáře a do ní zkopíruje soubory z GitHubu (pokud tam nějaké jsou).

Kroky 3 a 4 jsou už pak stejné jako v předchozí variantě...

3

Pokaždé, když provedu v projektu změny (obvykle nějaký ucelený blok), provedu commit a vytvořím tak záznam v historii:

```
git add .  
git commit -m "Popis"
```

4

Po provedení commitu (lokálně), ho mohu odeslat do vzdáleného repozitáře na GitHubu:

```
git push
```

Kroky 4 a 5 opakuji po každém bloku provedených změn a užívám si pocitu, že můj kód je v bezpečí na GitHubu, i kdybych si na notebook vylila čaj nebo mi vybuchnul.