



Git Going With Git

Git Overview

Git overview for Data Fridays

<https://gitgoingwithgit.com/>

- Data Friday
- October 15, 2021

What is Git?

“ Git is software for tracking changes in any set of files, usually used for coordinating work among programmers collaboratively...
-- Wikipedia ”

- A file structure for logging the history of a software project
- A set of tools for maintaining said file structure

Clone the repository for this presentation:

```
git clone https://github.com/Git-Going-With-Git/Data-Fridays-Presentation
```

Why Is Git Useful?

It makes collaboration easier.

- It's easy to "pull" the changes on remote repos from collaborators.
- It's easy to see the changes you and your collaborators make.
- You can maintain separate branches of work.
- You can merge your work.
- If you accidentally break something, git can revert it.
- Github (and clones) are *the* "hubs" for collaboration.

Terminology

- repository
- commit
- diff
- branch
- remote
- merge
- conflict

A Short Demo

Git Is Not Magic. Collaboration Is Still Hard

- Git knows nothing about your source code
- Git cannot resolve conflicts that arise during collaboration
- If your workflow is convoluted, then it will be hard to collaborate
- If your project structure is unorganized, then it will be hard to collaborate

Markdown

```
# Git Going With Git
```

```
## Overview
```

- * point 1
- * point 2
- * point 3

```
## Resources
```

- * resource 1
- * resource 2
- * resource 3

Installation

Installation of git may vary depending on the operating system.

Download `git` from git-scm.com

Configuration

After installing git, we must set our name and email:

```
git config --global user.name "SpongeBob Squarepants"  
git config --global user.email "spongebob@squarepants.com"  
git config --global credential.helper store
```

Verify this was saved by listing the global configuration:

```
git config -l
```

Tools

- command line
- GUI clients
- IDE integration
- web applications

Command Line

Git can be used entirely from the command line, and it is the preferred way to use git for many.

However, an easier way to get started is through graphical clients or integrations with your integrated development environment (IDE) of choice, such as RStudio and VSCode.

GUI Clients

Git GUI clients are graphical applications to make working with git easier.

Some popular ones can be found in [Git SCM's GUIs](#).

IDE Integration

Most IDEs (Integrated Development Environments) facilitate performing the most common git tasks such as

- commit
- push
- logs
- pull

Web Applications

GitHub is an example of a web application that works with git.

Other such applications include Overleaf.

*More details in **GitHub Overview**.*

Resources

Interactive Learning

- [Learn Git Branching](#)
- [Git Kata](#)
- [GitHub Learning](#)

Books

- [Pro Git](#)

Cheat Sheets

- [Interactive Cheat Sheet](#)
- [GitHub Markdown Cheatsheet.](#)
- [GitHub Git Cheat Sheet](#)

General

- [GitLab Training](#)
- [Official Documentation](#)
- [Command list](#)