

Basic Git Workshop

TechJI

University of Michigan - Shanghai Jiaotong University

Joint Institute

2023.10.8

Table of Contents



Table of Contents



What is Git?

Git is a free and open source distributed version control system designed to handle everything from small to very large projects with speed and efficiency.



When to use Git?

- ▶ Writing code
- ▶ Managing docs
- ▶ Managing projects
- ▶ Version control
- ▶ Team corporation



Git Installation

You can go to the official website to download the installation package and refer to *Installation_git.md* file in the repo.



Where to use Git?

We first introduce shell commands, then we will use Git in the shell.



What is Shell?

Shell is a program that takes commands from the keyboard and gives them to the operating system to perform. It is also referred to as a command-line interpreter or CLI.

Common shells:

- ▶ Bash
- ▶ Zsh
- ▶ etc

A brief history of bash



- ▶ Born: 1989
- ▶ Probably played Pokémon on the Game Boy
- ▶ Is an umbrella term for zsh, fish, ...
- ▶ Runs on Unix-like environments

A brief history of Unix



- ▶ Born: 1969
- ▶ Probably listened to Michael Jackson
- ▶ Gave rise to Linux, BSD, and Mac OS
- ▶ We call them “Unix-like”

Unix: The Good Part

The Unix philosophy (paraphrased):

- ▶ Store data in plain text
- ▶ Hierarchical file system
- ▶ Everything is a file
- ▶ One tool does one thing
- ▶ Tools together strong

Quote

The power of a system comes more from the relationships among programs than from the programs themselves.

— Brian Kernighan and Rob Pike ¹

¹The UNIX Programming Environment. 1984. viii



How to open a Shell?

- ▶ Windows: cmd, powershell
- ▶ Mac: Terminal
- ▶ Linux: Terminal

Table of Contents

Files

Each of these is a different **file**:

- ▶ a
- ▶ .a (Hidden)
- ▶ a.txt
- ▶ A.txt
- ▶ A.TXT

Note

The dot and suffix are part of the filename. Windows users please turn on **show file extensions**.

Avoid spaces and special characters (except `._-`). If you have to, surround filename in quotes: ``Lab Report (3) final FINAL-1.docx'`

Directories

Each of these is a **directory** (“dir” for short):

- ▶ `hteam-10086/`
- ▶ `hteam-10086/h1/`
- ▶ `hteam-10086/.gitea/` (Hidden dir)

Convention

For clarity, we add a slash (/) to the end of a directory in the slides. However, in reality it often makes no difference.

cd, pwd: Changing directory

- ▶ `cd hteam-10086/`
- ▶ `pwd`

Explanation

- ▶ `cd`: "change directory"
- ▶ `pwd`: "print working directory"
- ▶ `../` means "parent directory"
- ▶ `./` means "current directory"
- ▶ `~` means "home directory"

Paths

File \cup directory = **path**.²

No paths under the same directory can bear the same name. These **cannot** coexist:

- ▶ hteam-10086/h1/, a directory
- ▶ hteam-10086/h/ex1.m, a regular file

²At least in the scope of this workshop.

Absolute & relative paths

- ▶ Paths beginning with / are absolute: `/usr/bin/cat`
- ▶ Otherwise it is relative: `hteam-10086`

If you know where you are, you can convert a relative path to an absolute one.

Table of Contents

Some basic git commands

Tell Git who you are:

- ▶ `git config --global user.name "Your Name"`
- ▶ `git config --global user.email "Your Email"`

Some basic git commands

How to use "git add":

- ▶ `git add <file>`
- ▶ `git add *`
- ▶ `git add .`
- ▶ `git add -A`

use "git status" to check the status of your repo.

Some basic git commands

How to use "git commit":

- ▶ `git commit -m "commit message"`

If you only type:

- ▶ `git commit`

then you will enter vim /other default editor to write your commit message.

Some basic git commands

How to use "git rm":

- ▶ `git rm -cached <file>`

Then, git will stop tracking this file, but the file still exists in your repo.