Command Line, Git, Version Control, and Markdown

Hyuk Harry Son

LDI Lab Training

January 25, 2021

Command line
Overview

1. A user interface that's navigated by typing commands at prompts, instead of using a mouse.

2. Many tasks can be performed quicker and can be easier to automate and do remotely.

3. The command line is the only place you can run all Git commands. (GUI operating system is limited)

4. Examples of command line (shell):
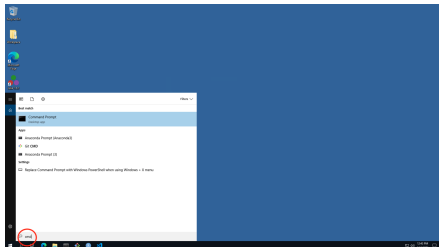   **git bash**, terminal (macOS), command (Windows), Powershell, etc

Command line

How to open command line: Terminal and command

1. Terminal: search for "terminal' 'using spotlight serach

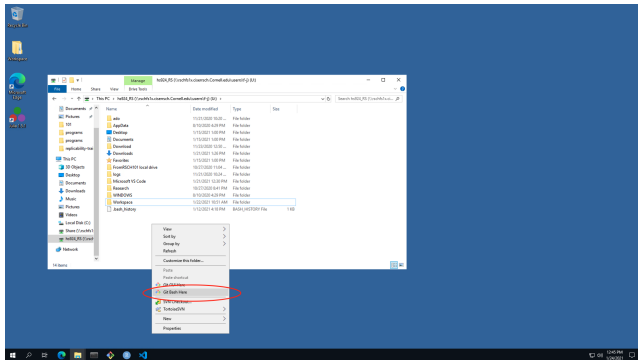2. Command prompt: search for "cmd" using start menu search
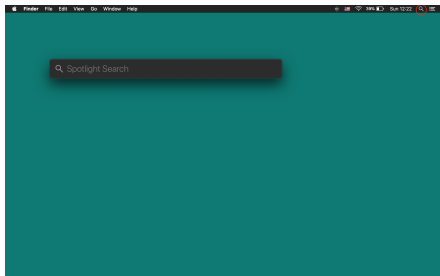
(a) Terminal (macOS)

(b) command prompt (Windows)

Command line

1. Navigate to the directory you would like to work on using file explorer, right click, select "git bash here"

(c) git bash

Command line
Example commands

1. pwd: show current working directory

2. cd: navigate directories

3. mkdir, rmdir: create and remove directories

4. For more information, click ▸ here

Git
Overview

1. One of the Distributed Version Control Systems (More on version control later).

2. Three states: Modified, staged, committed
   - Modified: You have changed the file but have not committed it to your database yet.
   - Staged: You have marked a modified file in its current version to go into your next commit snapshot.
   - Committed: The data is safely stored in your local database.

3. For more information, click ▸ here

Git
Commands

1. **git add:** take a snapshot of the contents and store in a temporary staging area.

2. **git commit:** permanently store the contents in the staging area to the local repository.

3. **git push:** update remote repository using changes in the local repository.

4. **git pull:** update local repository using changes in the remote repository.

5. **git status:** get a brief summary of the situation.

6. **git clone:** clone a repository into a new directory.

Git
Flow


```
Original --Local changes--> Modified --git add--> Staged --git commit--> Committed --git push--> Synced to the remote repository
```
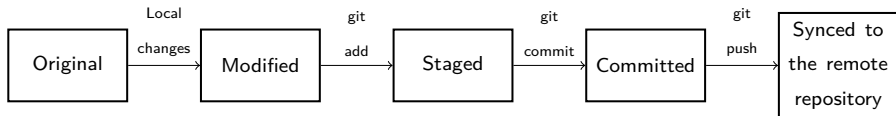
Version Control
Overview

1. A system that records changes to a file or set of files over time so that you can recall specific versions later.

2. One of the key advantages of using version control systems is... to control versions.
   - Straightforward to view multiple versions of a file (assuming proper usage).
   - Possibility to view who changed what ("blame" or "annotate").

3. For more information, click ▸ here

Version Control

Example: Track history of changes

(a) History and comments

Version Control

Example: Track history of changes

(b) List of changed files

Version Control

Example: Track history of changes

(c) Differences within file line by line

Markdown
Overview

1. A lightweight markup language that you can use to add formatting elements to plaintext text documents.

2. Why use markdown?
   - Markdown is portable. Files containing Markdown-formatted text can be opened using virtually any application.
   - Markdown is platform independent. You can create Markdown-formatted text on any device running any operating system.
   - Markdown is future proof. You can read markdown-formatted text using a text editing application, even when your application stops working.

3. For more information, click ▸ here

Markdown
Tips

1. Markdown uses syntax for text formatting.
   - You need a tool to translate from plain text to a formatted document.
   - We use: Visual Studio Code

2. Useful extensions in Visual Studio Code
   - Preview to the side
   - Excel to markdown table

Credits to the authors of the materials in the link.