C: Version control

Challenges of large analytical projects:

- Many, many lines of code: easily 100's or 1000's of lines
- Analysis and input data evolves over time:
  - Need to track what changes and why
  - May need to revert to earlier version
- Can have many collaborators and need to communicate

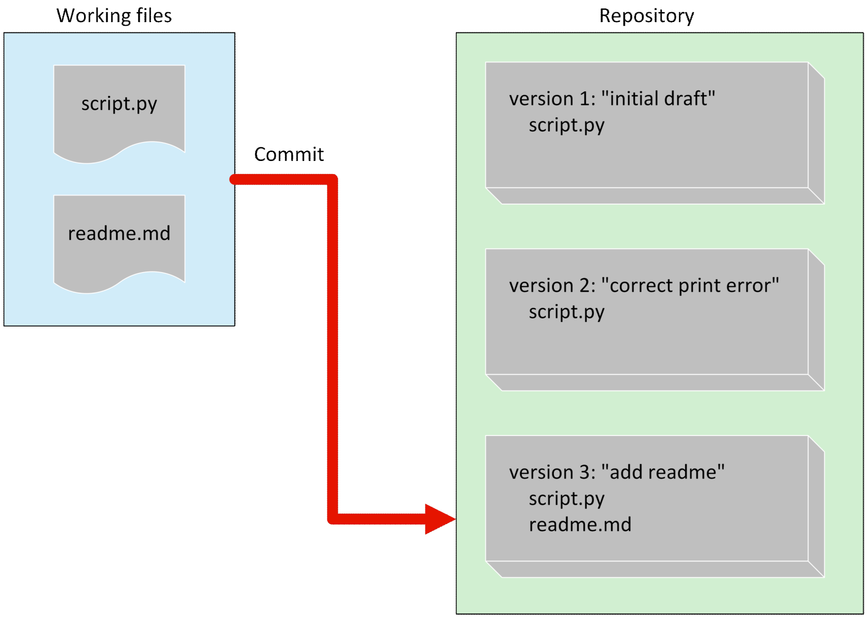
Solution:

Software tools known as **version control systems**

Key concepts:

1. Set of **working files**
2. **Repository** of tracked and logged changes
3. Working files are periodically **committed** to the repository

Schematically:



Benefits:

- Can see **what changed** between versions
  - Very useful when new code causes unexpected results

- Can **revert** to previous versions
  - Very useful when changes break working code
  - Can reproduce earlier results

- Keeps track of **who** changes code and **why**
  - Very important for all but the smallest projects

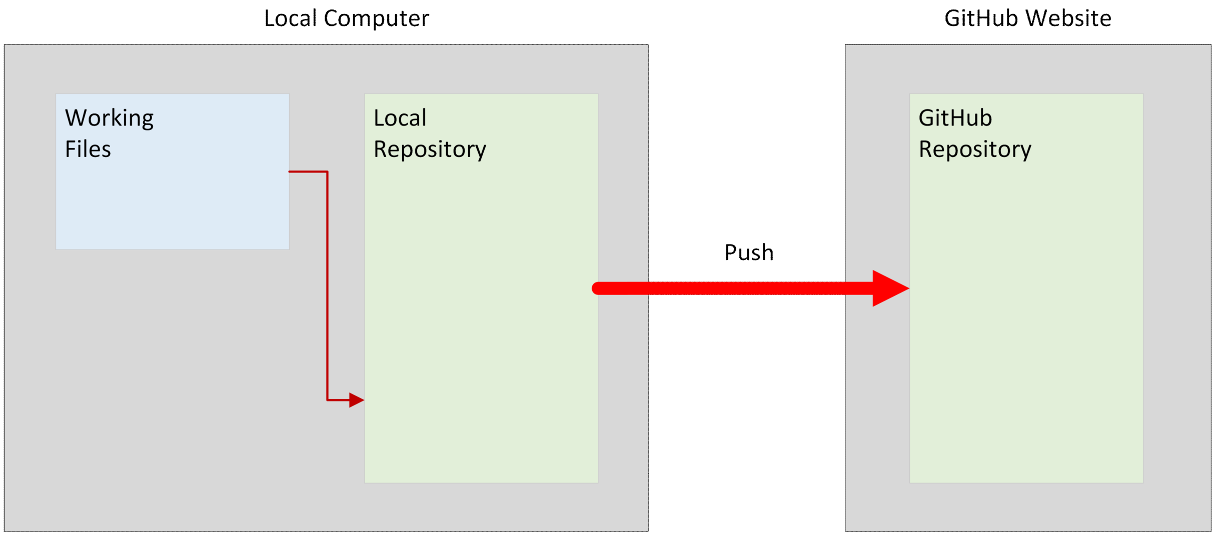
Git and GitHub

Definitions:

Git **Version control system** for tracking changes in files

GitHub Widely used **website of repositories** for open source software

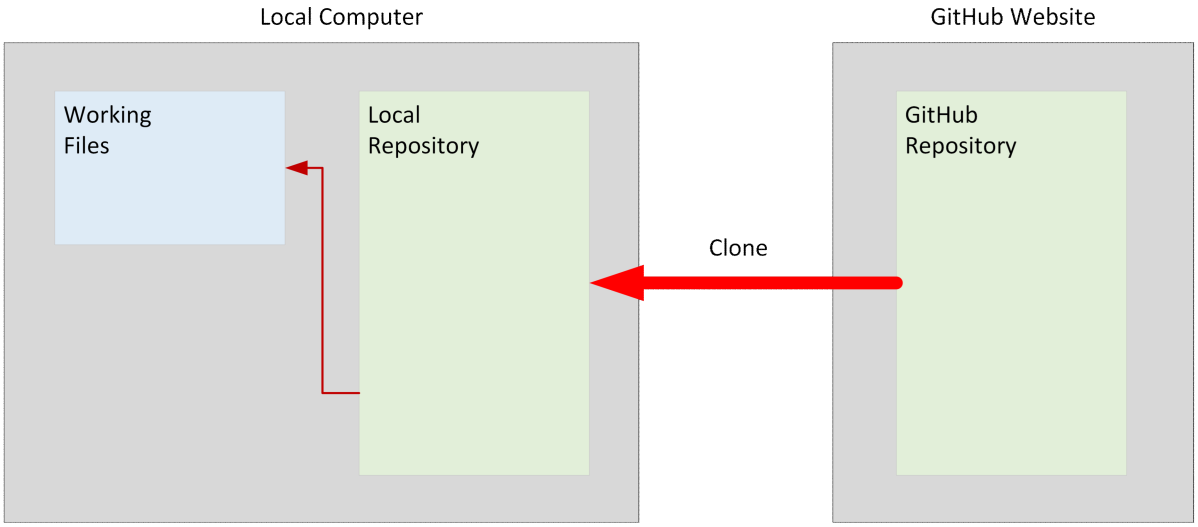
Adds an external cloud repository and an extra step:



Additional benefits from the web repository:

- Safe cloud storage
- Easy to view files and history of changes on the web

- Biggest benefit by far: can be **cloned** or **forked** by others:



- Easy, **reliable**, **robust collaboration** on shared files
- Easy to **publish open source** research for public use

We'll use Git and GitHub heavily this semester:

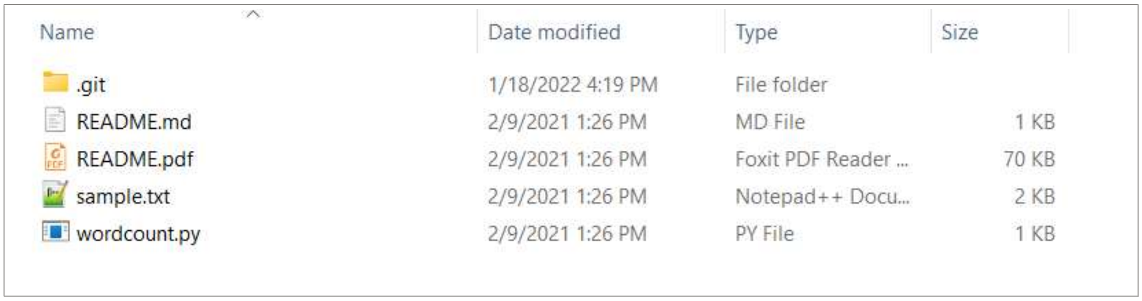
1. You'll **clone a starter repository** for each computing assignment
2. As you write your code, **you'll commit it and push it** to the web
3. After the deadline, I'll **clone your web repository** for grading

Example 1: Counting words in the Gettysburg Address

View on GitHub:

https://github.com/maxwell-pai789/e100-demo

View on local computer in directory e100-demo:



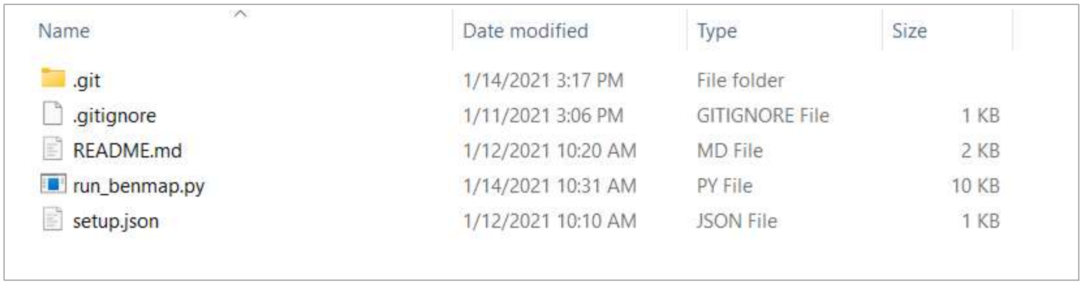
Name	Date modified	Type	Size
.git	1/18/2022 4:19 PM	File folder	
README.md	2/9/2021 1:26 PM	MD File	1 KB
README.pdf	2/9/2021 1:26 PM	Foxit PDF Reader ...	70 KB
sample.txt	2/9/2021 1:26 PM	Notepad++ Docu...	2 KB
wordcount.py	2/9/2021 1:26 PM	PY File	1 KB

Example 2: runBenMAP

View on GitHub:

https://github.com/pjwilcoxen/runBenMAP

View on local computer:



Name	Date modified	Type	Size
.git	1/14/2021 3:17 PM	File folder	
.gitignore	1/11/2021 3:06 PM	GITIGNORE File	1 KB
README.md	1/12/2021 10:20 AM	MD File	2 KB
run_benmap.py	1/14/2021 10:31 AM	PY File	10 KB
setup.json	1/12/2021 10:10 AM	JSON File	1 KB

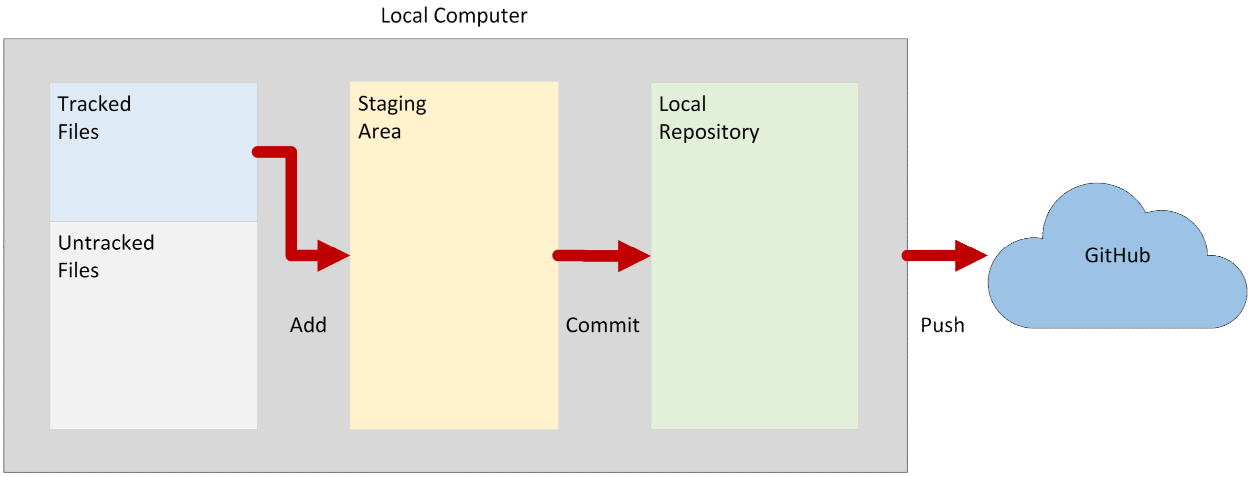
Two important additional subtleties :

1. May **not want to track some files** in a directory
   - Examples: temporary files, intermediate data files

     Handled via .gitignore files that tell Git what to ignore

2. Often want to **commit several files together** with one log message
   - Example: script and output file

     Handled by **adding** files to a **staging area** before the commit:



Will manage all this with **GitHub Desktop**:

- Set up and use in class on Monday

Google Classroom assignments