Version Control + Markdown

INFO 201

Today's Objectives

Feel comfortable with basic ***terminal commands***

Understand the basic steps necessary to implement a ***version control*** **system**

Learn how to stylize a document using ***markdown syntax***

Terminal Review

Terminal Purpose

Provide instructions directly to your machine

Navigate your file system, which is a tree structure

Will mostly be used for implementing version control commands (more on this later)

A few hints:

- Navigate to a *parent* directory using this syntax: `cd ../other-child`
- Create a new terminal window (`cmd + n`) or tab (`cmd + t`)
- Access the current user's directory with the tilde (~): `cd ~/Documents`
- Access last entered commands (up arrow, down arrow)

Action	Syntax
Copy a file	`cp OLD_FILE NEW_FILE`
Move a file	`mv OLD_FILE NEW_FILE`
Delete a file (careful!)	`rm FILE_NAME`
Create a new file (windows alternative)	`touch FILE_NAME`
Open a file	`open FILENAME (windows: start FILENAME)`
View text of a file	`less FILE_NAME`
See previous commands executed	`history (also hit up arrow)`
View Manual information for a command	`man COMMAND`

Basic Terminal Commands

Action	Syntax
Make your computer speak	`say "text"`
Watch Starwars	`telnet towel.blinkenlights.nl`
Do the same thing again	`!!`
Play pong	`emacs, esc + x, pong`
Talk to a virtual psychotherapist	`emacs, esc + x, doctor`
Request URL content	`curl URL`
Save URL content into a file	`curl URL > filename.txt`

Fun Terminal Commands

The Pipe Operator

Take the *results* from one action, and pass them in as an *argument* to the next function

On your terminal, use the vertical bar (pipe character): "|"

We'll use this more in a few weeks, but for now:

```
curl -s https://baconipsum.com/api/?type=all-meat |say
```

Version Control

What are 3 reasons that keeping track of the version of your code is important?

60% of you report having
"*moderate*" or "*lots*" of
experience programming, but
only **9%** of you have used
version control more than "*a
few times*" (n=72)

Name
® analysis_evan_v2.r
® analysis_evan.r
® analysis_mike_FINAL.r
® analysis_mike.r
® analysis_ERROR.r
® analysis_v2.r
® analysis_v1.r
® analysis_PUBLICATION.r
® analysis_NEW_FINAL.r
® analysis_FINAL.r
® analysis_12_28.r
® analysis.r

Without Version Control

Git and GitHub

Track code changes

Share code with others

Facilitate collaboration

Host websites

Go back in time

How it works

Track all files in a directory (repository)

Identify line-by-line changes

Take snapshots of your files, along with a short message about changes

Share files on the cloud for collaboration

Git

Version control software

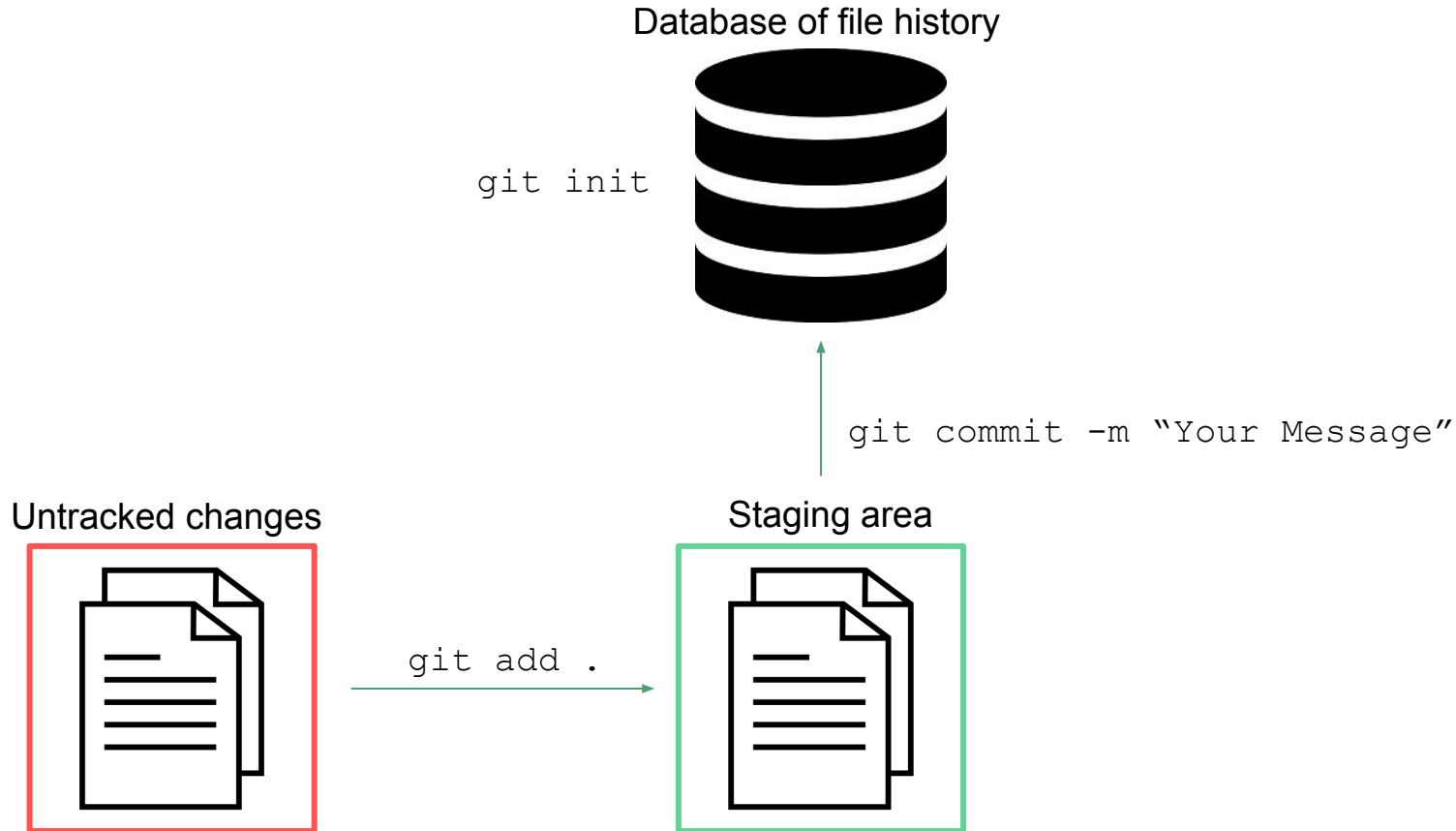Free and open-source

Enable new terminal commands for file management

Industry standard for version control

Local Git Process

Database of file history

git init

git commit -m "Your Message"

Untracked changes

git add .

Staging area

module-4

Using GitHub

GitHub

your machine

another machine
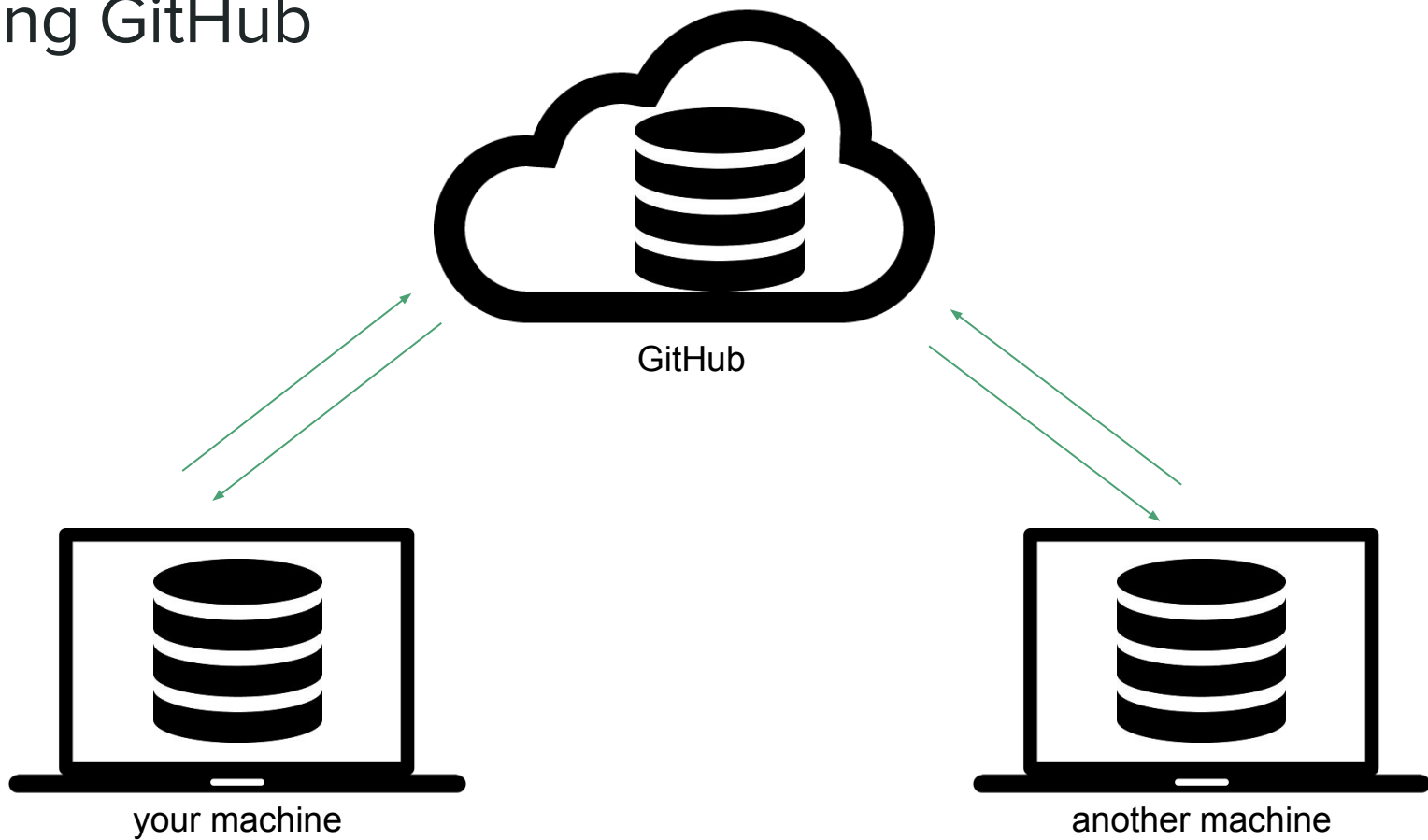
GitHub

Cloud based repository hosting service

Allows you to store your project on the web

Enables sharing and collaboration

Backbone of open-source development

your GitHub repo

starter GitHub repo

Fork the repository

git clone URL

git push

your machine

Staging area

Edit files

git commit -m "Message"

git add .

GitHub Process (used for in class exercises)

module-4

[Module-4 exercise-1](#)

your GitHub repo (inside the course organization)

Click the canvas link

Create a repo for you

git clone
URL

git push

your machine

Staging area

Edit files

git
commit -m
"Message"

git add .

GitHub Process (used for *assignments*)

module-4
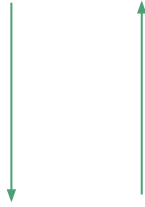
test-assignment

Markdown

Markdown

Easy to write, easy to read syntax for declaring document structure and formatting

A simple shorthand for describing content rendering

```
# Top Level Header
## Second Level Header
```
block of code
across multiple lines
```

- List item 1
- List item 2
- List item 3

>Here is a block quote.
```

Top Level Header

Second Level Header

```
block of code
across multiple lines
```

- List item 1
- List item 2
- List item 3

Here is a block quote.

module-3

Who uses Markdown?

Slack: we expect questions to be clearly formatted with `inline` code, as well as code sections (``` code ```)

GitHub: All learning modules are written in Markdown. Each repository's Readme.md file is written in Markdown and rendered on the GitHub page.

Stackoverflow: A programming question/answer forum.

Rendering Markdown

GitHub renders Readme.md when you push to your repo (but you might want a preview...)

- Online editors (dillinger)
- MacDown text editor
- Markdown Reader chrome extension
- PanDoc if you want to make the documents yourself using the terminal

[Module-3 exercise-1](#)

[Module-3 exercise-2](#)

Upcoming…

By Thursday: You should feel comfortable with **modules 3 - 4**

Due Thursday, 10/6 (***before class***): [a1-git-started](#)