浙江大学爱丁堡大学联合学院

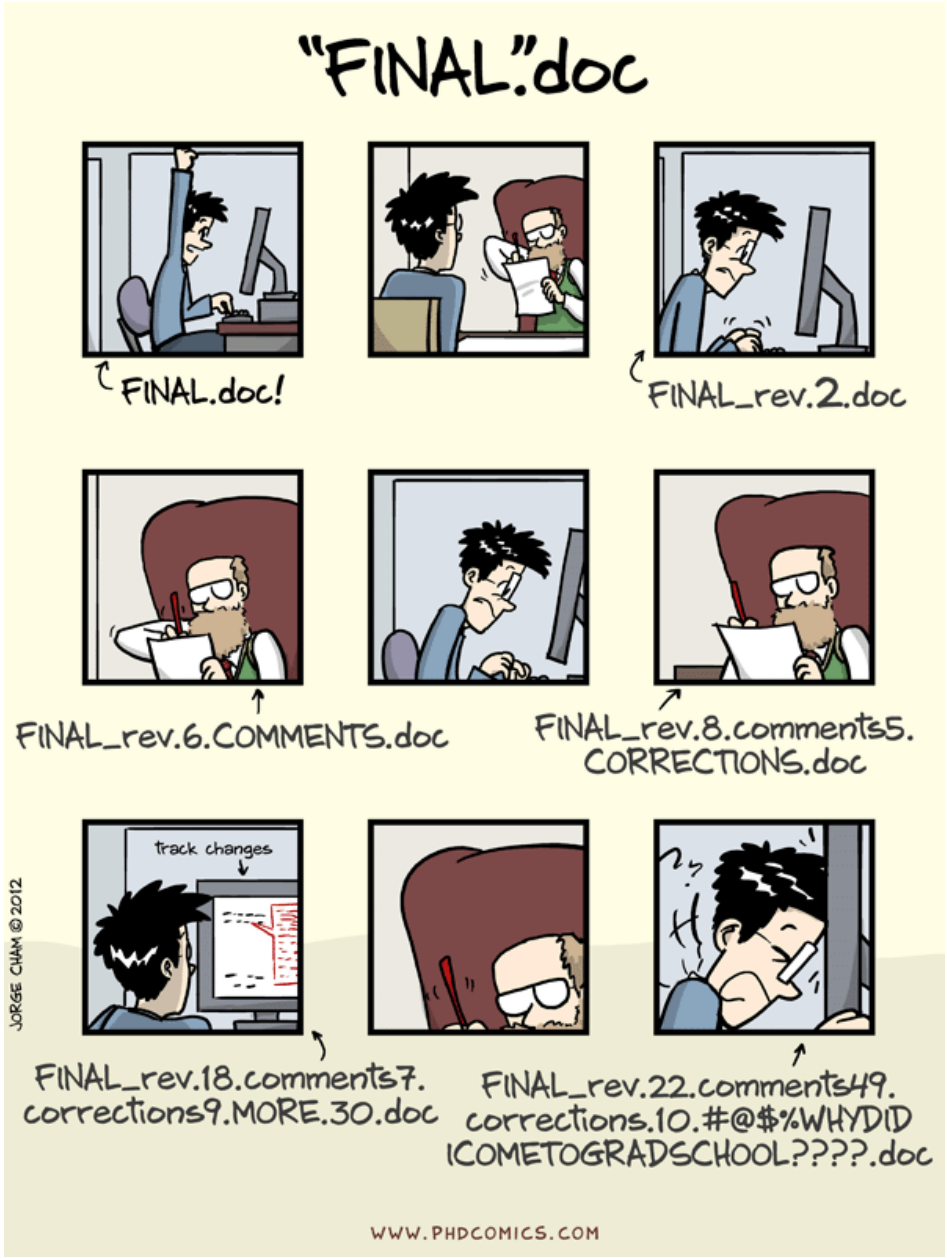
ZJU-UoE Institute

# Introduction to version control

IBI 1, Lecture 4.1

Melanie Stefan - melanie.stefan@ed.ac.uk

Semester 2, 2019/20




"Piled Higher and Deeper" by Jorge Cham, `www.phdcomics.com`

# Is there a better way to do this?

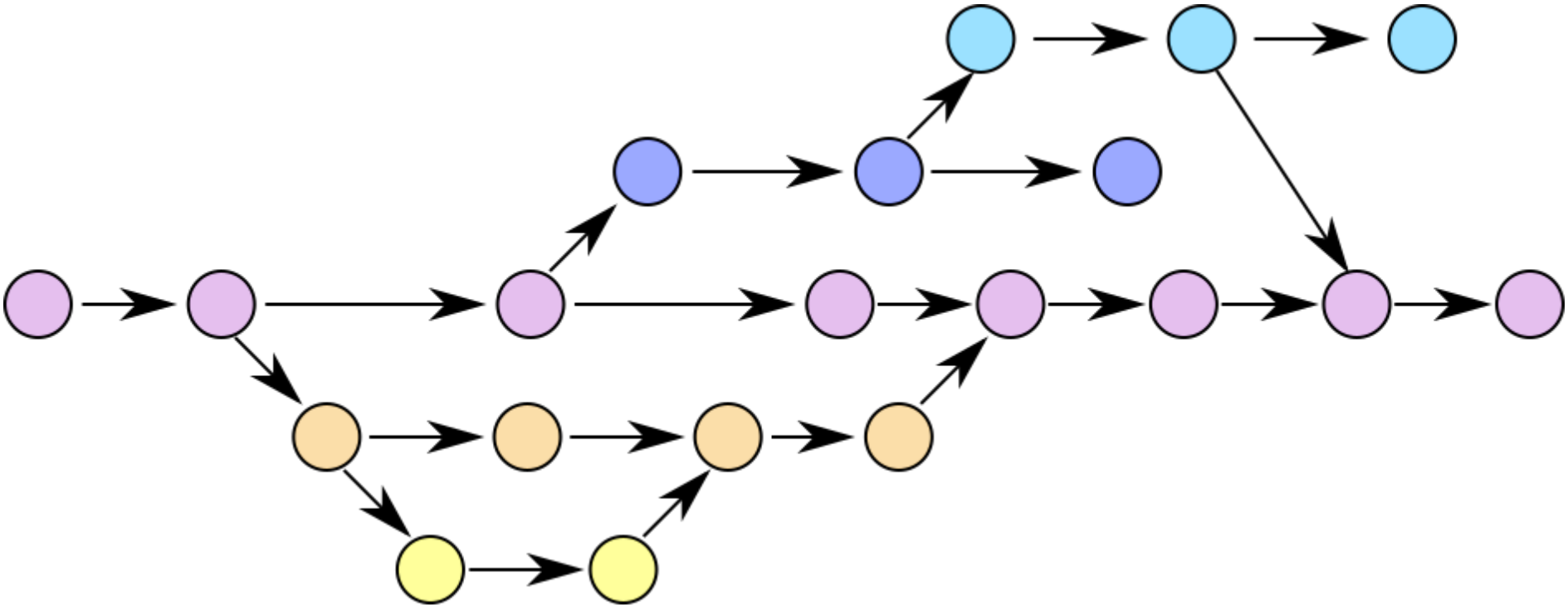
# Is there a better way to do this?

Why yes, there is. And this is what this lecture is about.

# Learning Objective

## After this lecture, you should be able to . . .

Explain the principles of version control



# Outline


1. Idea behind version control
2. Basic concepts of version control
3. Version control in this course

# Idea behind version control

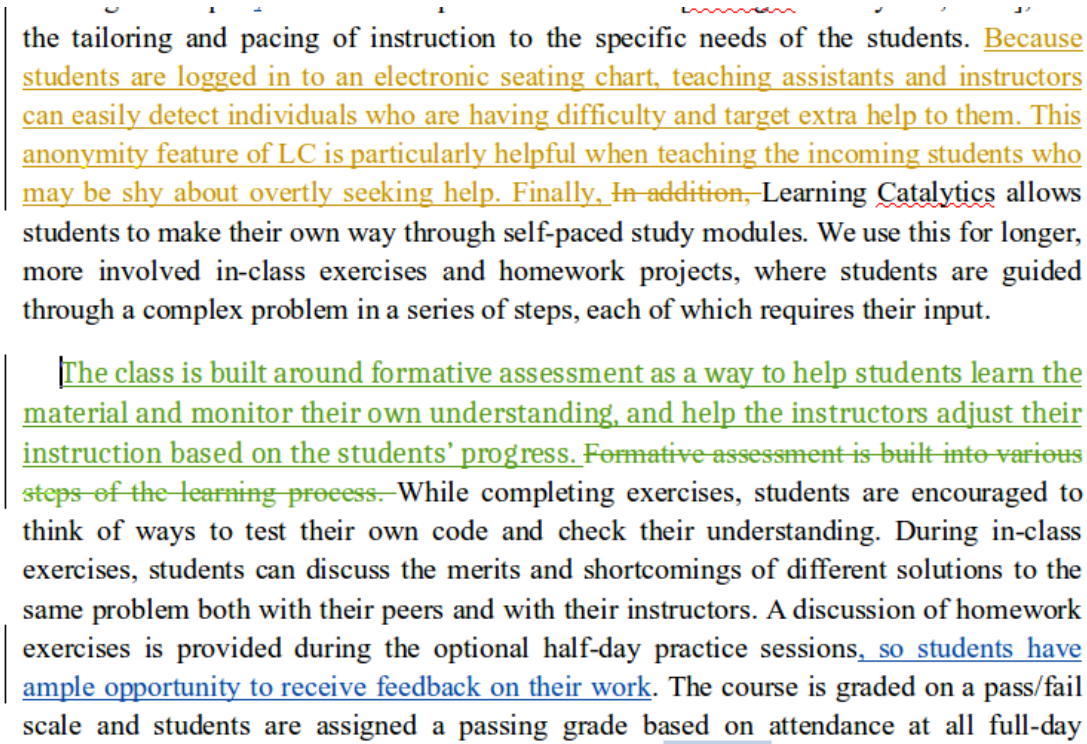
A *version-control software* keeps track of versions of your project. You only work on one file, but there is a system that keeps track of changes and allows you to *undo changes*, *go back* to previous versions, *keep some changes and discard others*, and *work simultaneously with other people*.

# Idea behind version control

A *version-control software* keeps track of versions of your project. You only work on one file, but there is a system that keeps track of changes and allows you to *undo changes*, *go back* to previous versions, *keep some changes and discard others*, and *work simultaneously with other people*. Have you seen something similar?

# Idea behind version control

A *version-control software* keeps track of versions of your project. You only work on one file, but there is a system that keeps track of changes and allows you to *undo changes*, *go back* to previous versions, *keep some changes and discard others*, and *work simultaneously with other people*. Have you seen something similar?



the tailoring and pacing of instruction to the specific needs of the students. Because students are logged in to an electronic seating chart, teaching assistants and instructors can easily detect individuals who are having difficulty and target extra help to them. This anonymity feature of LC is particularly helpful when teaching the incoming students who may be shy about overtly seeking help. Finally, ~~In addition,~~ Learning Catalytics allows students to make their own way through self-paced study modules. We use this for longer, more involved in-class exercises and homework projects, where students are guided through a complex problem in a series of steps, each of which requires their input.

The class is built around formative assessment as a way to help students learn the material and monitor their own understanding, and help the instructors adjust their instruction based on the students' progress. ~~Formative assessment is built into various steps of the learning process.~~ While completing exercises, students are encouraged to think of ways to test their own code and check their understanding. During in-class exercises, students can discuss the merits and shortcomings of different solutions to the same problem both with their peers and with their instructors. A discussion of homework exercises is provided during the optional half-day practice sessions, so students have ample opportunity to receive feedback on their work. The course is graded on a pass/fail scale and students are assigned a passing grade based on attendance at all full-day

# What should a good version control system provide?

# What should a good version control system provide?

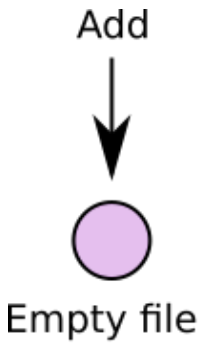
- Keeping track of versions and changes made
- Undoing changes
- Going back to a previous version of the project
- Several people working on the project at the same time
- Keeping track of who changed what when
- Bringing different versions together

# Outline

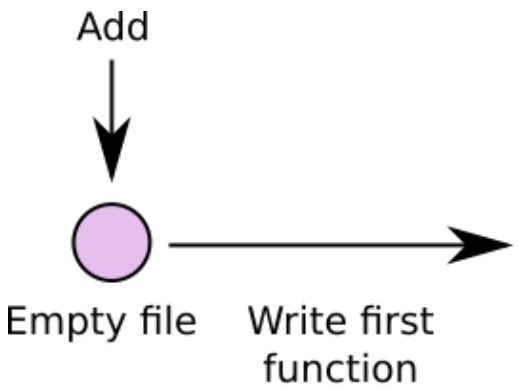
1. Idea behind version control
2. Basic concepts of version control
3. Version control in this course

# Commit

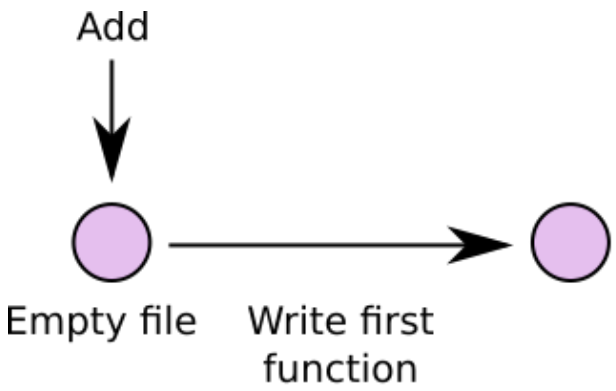
# Commit



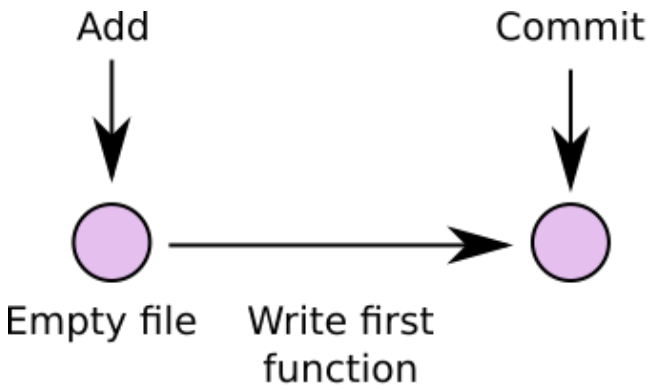
# Commit



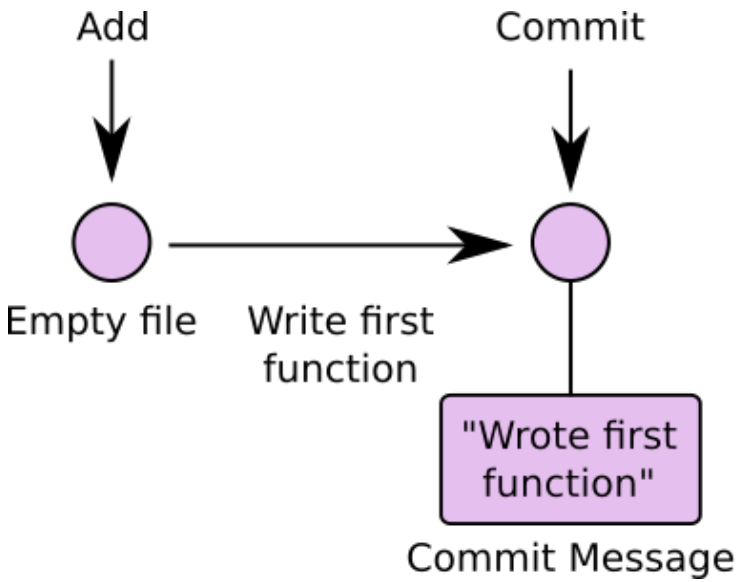
# Commit



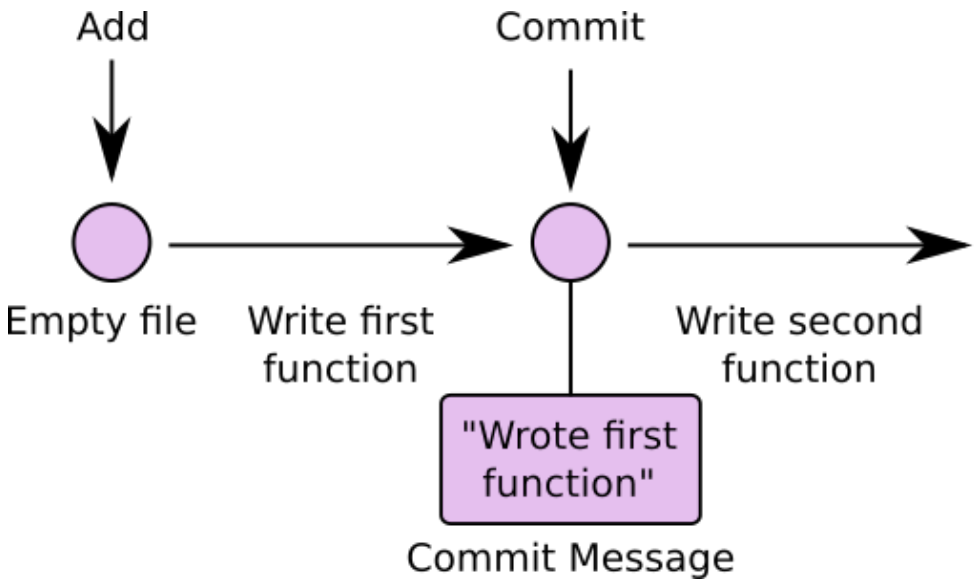
# Commit



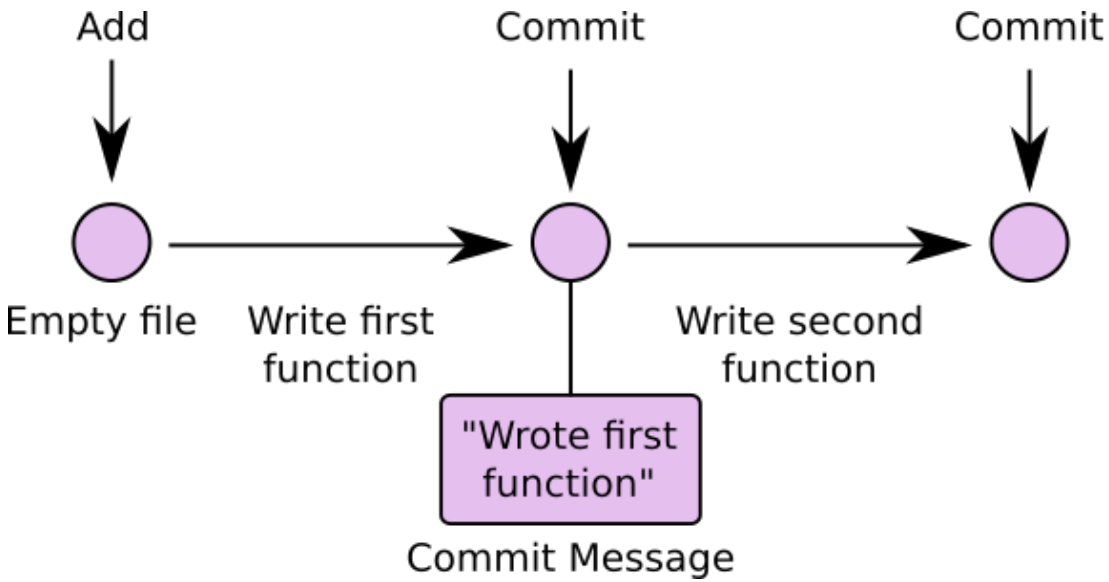
# Commit


Add
Commit
Empty file
Write first function
"Wrote first function"
Commit Message

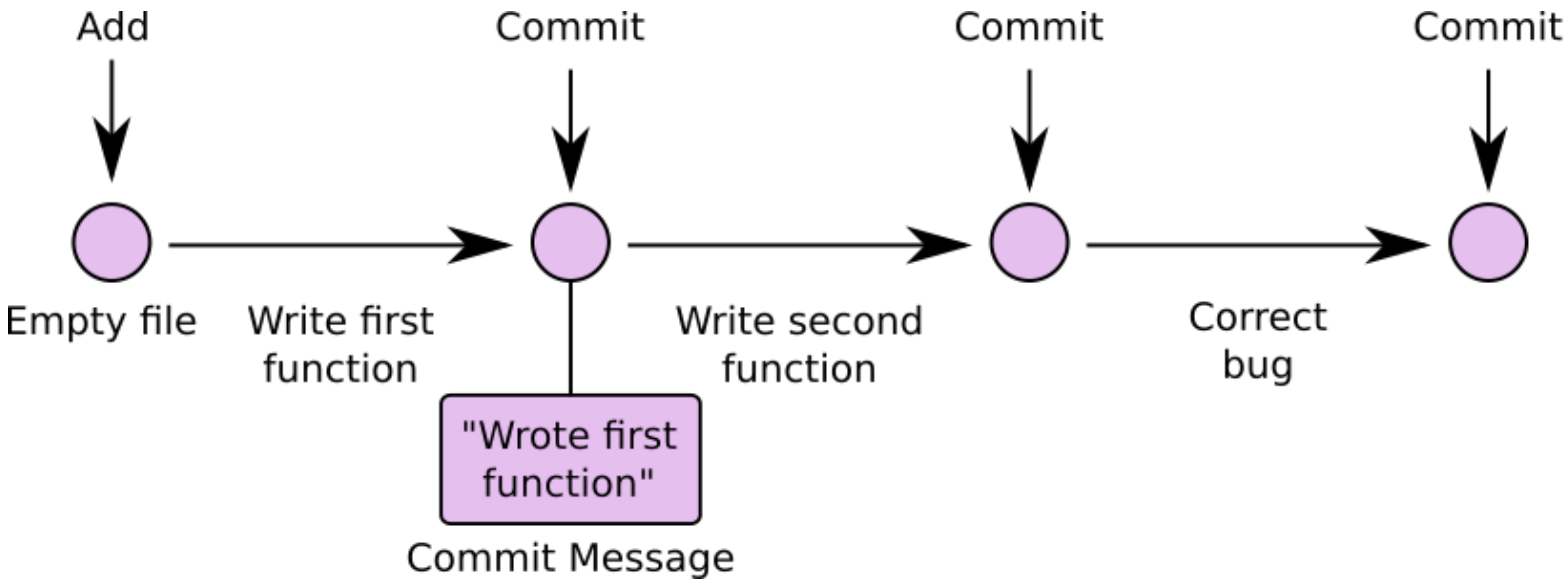
# Commit


Add
Commit
Empty file
Write first function
Write second function
"Wrote first function"
Commit Message

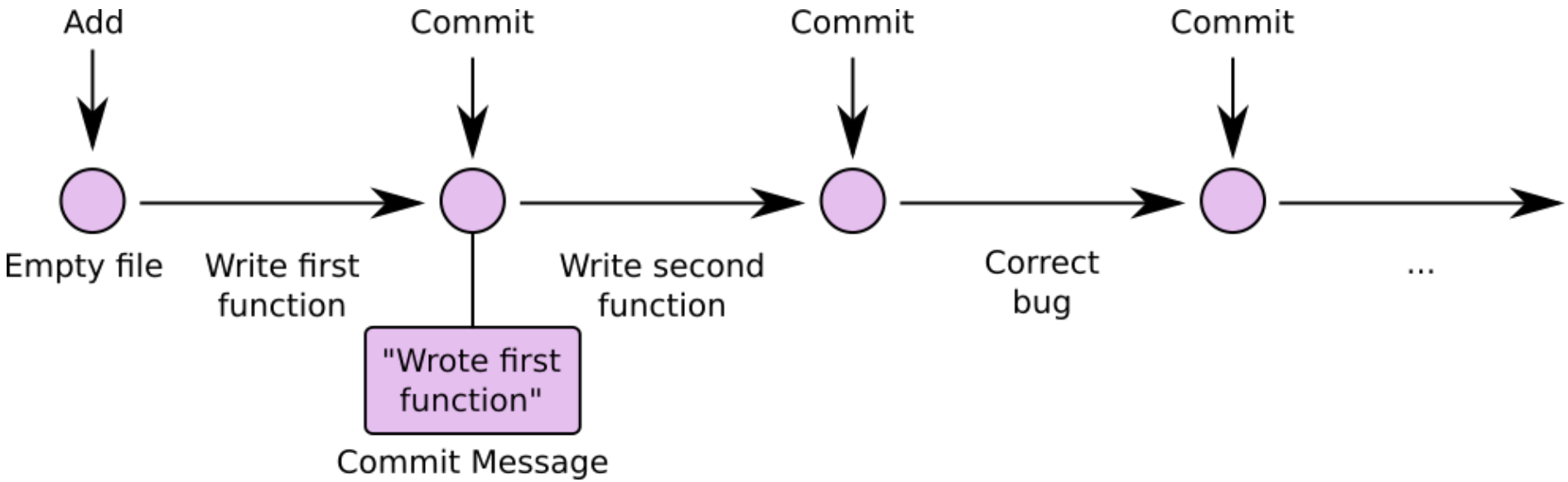
# Commit


Add
Commit
Commit
Empty file
Write first function
Write second function
"Wrote first function"
Commit Message

# Commit

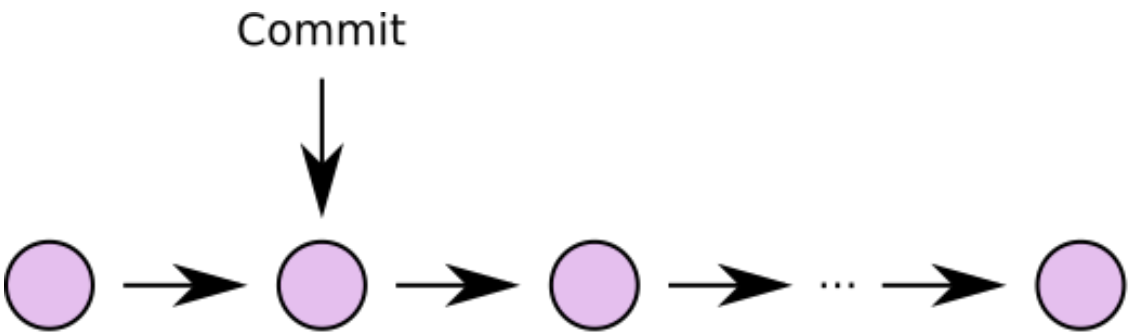

Add
Commit
Commit
Commit
Empty file
Write first function
Write second function
Correct bug
"Wrote first function"
Commit Message

# Commit

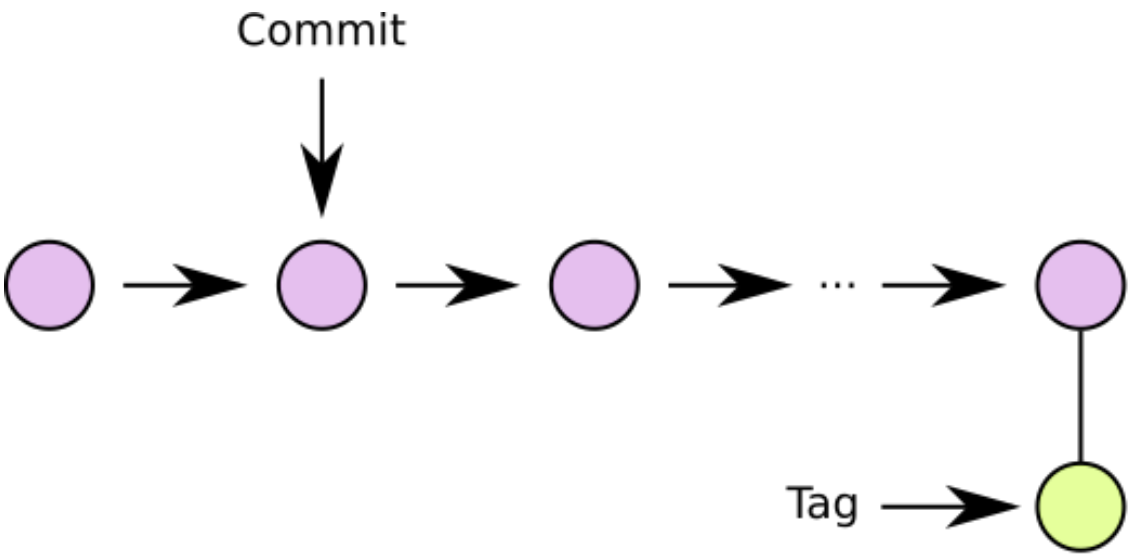

Add
Commit
Commit
Commit
Empty file
Write first function
Write second function
Correct bug
...
"Wrote first function"
Commit Message

# Tag

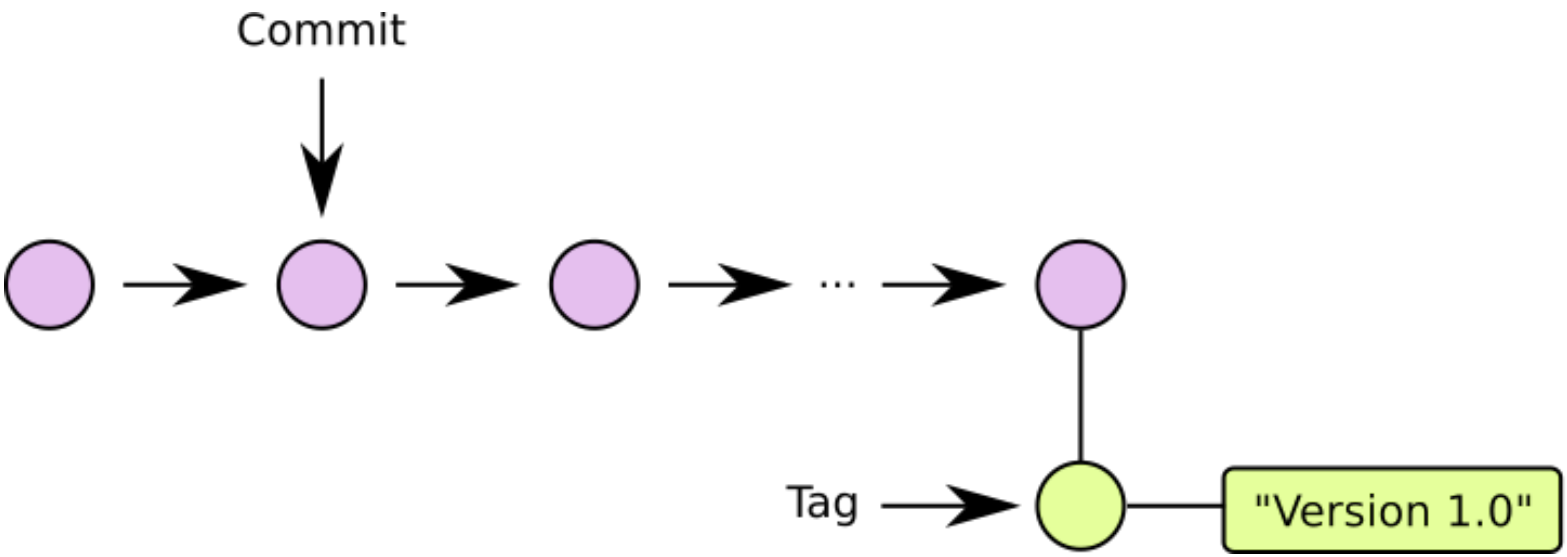
# Tag



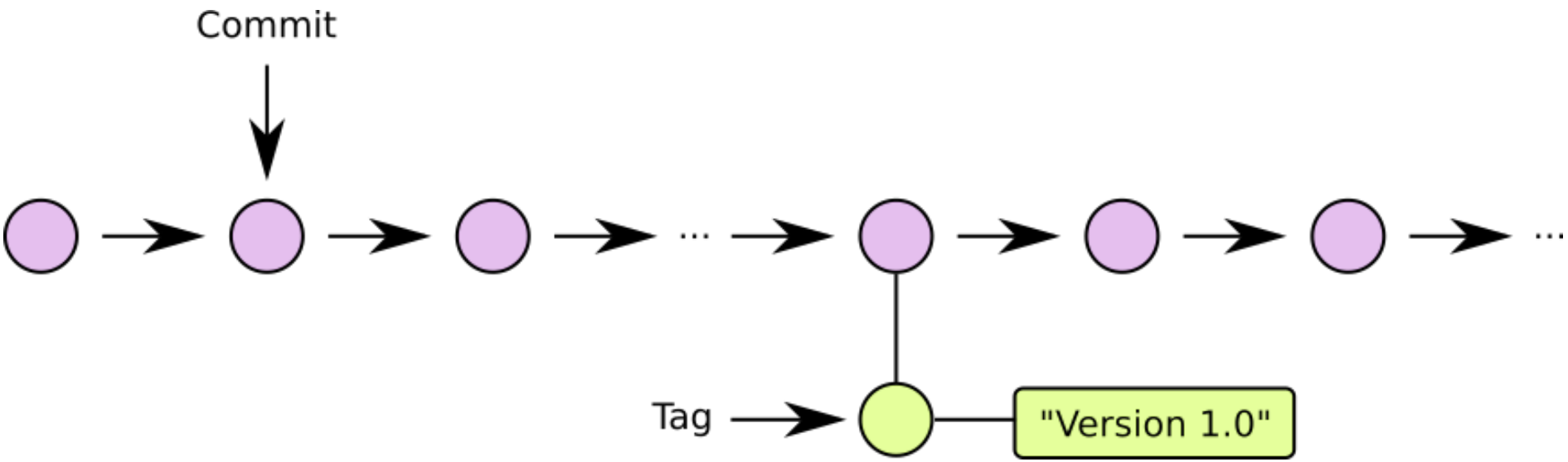
# Tag



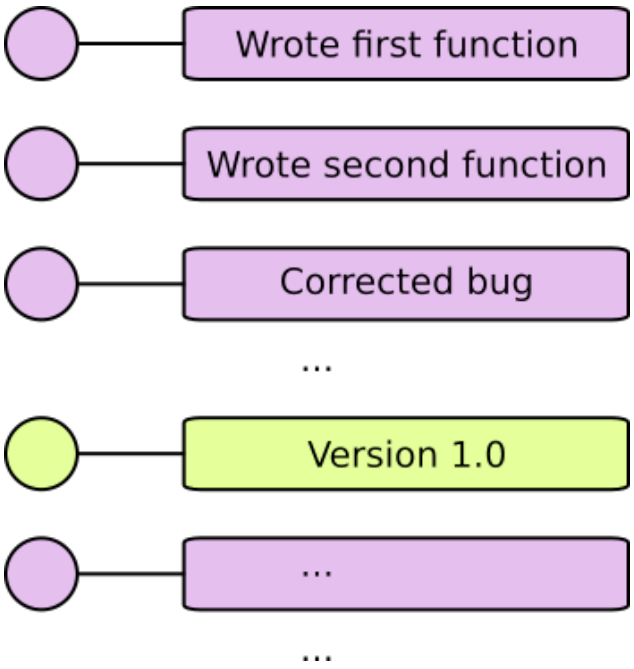
# Tag



# Tag

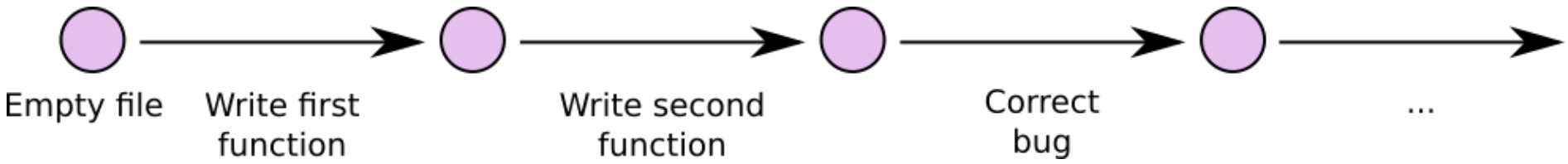


# Log

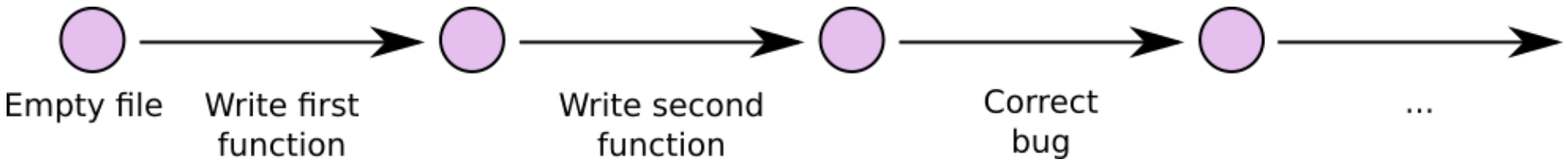



- The log shows the history of the file
- Includes info about previous commits/tags
- Includes commit messages
- Additional info: commit number, author, date

# Revert

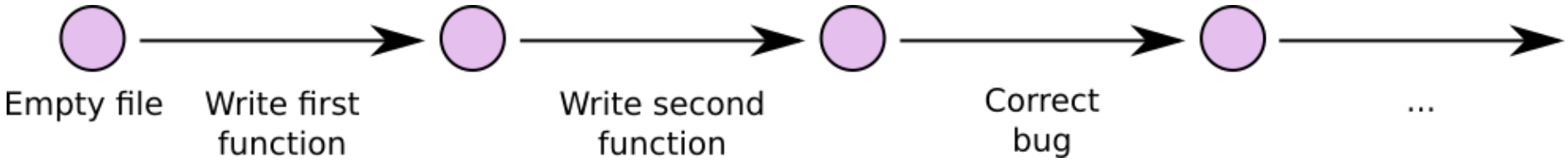

Empty file
Write first function
Write second function
Correct bug
...

# Revert

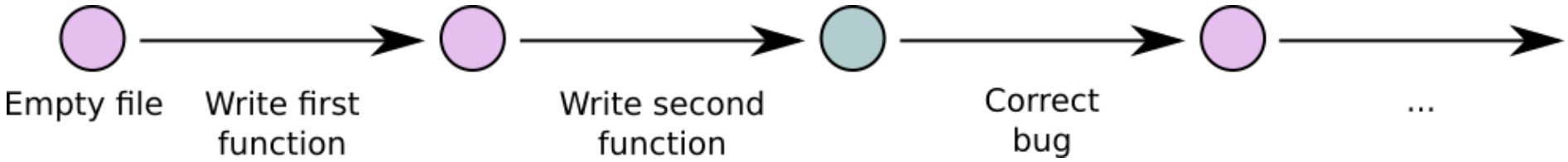



"The second function was fine, but when I thought I fixed the bug, I broke something."

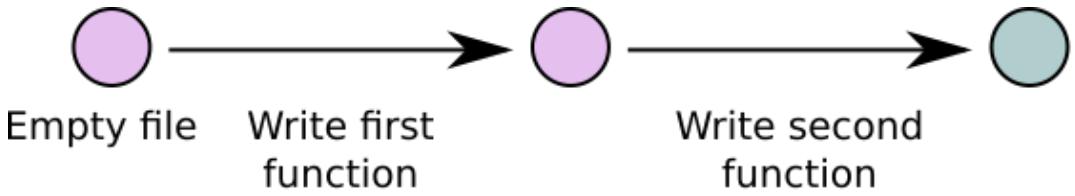
# Revert


Empty file
Write first function
Write second function
Correct bug
...

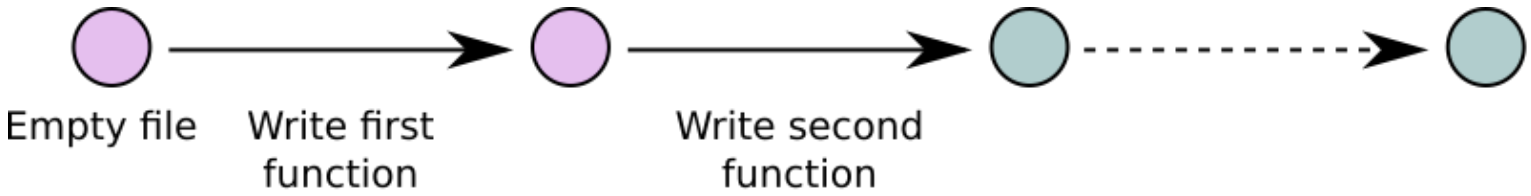
# Revert


Empty file
Write first function
Write second function
Correct bug
...

# Revert

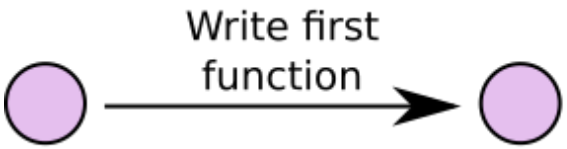


# Revert

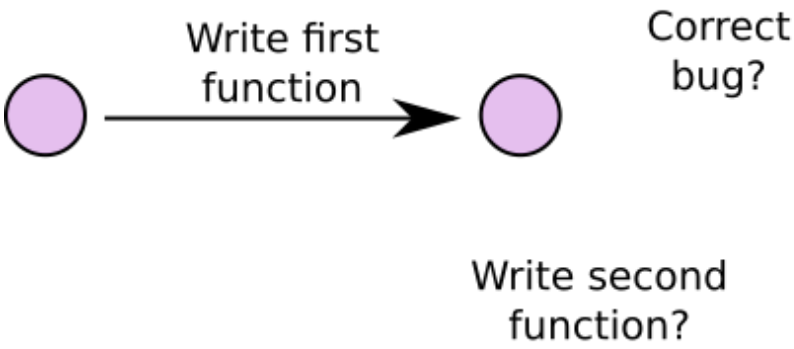


# Branch

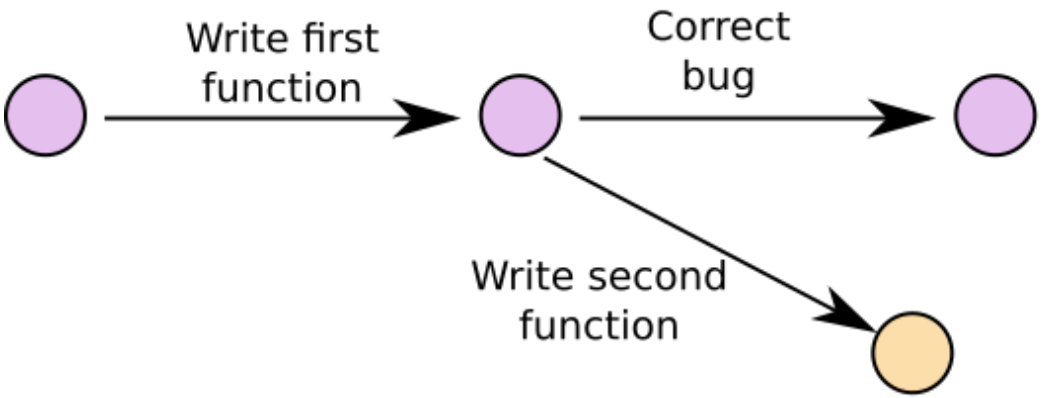
# Branch



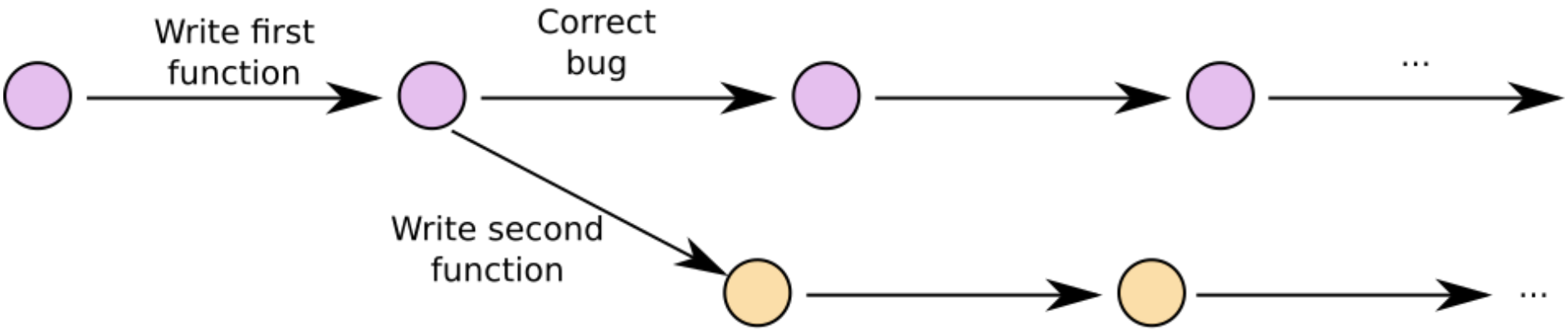
# Branch



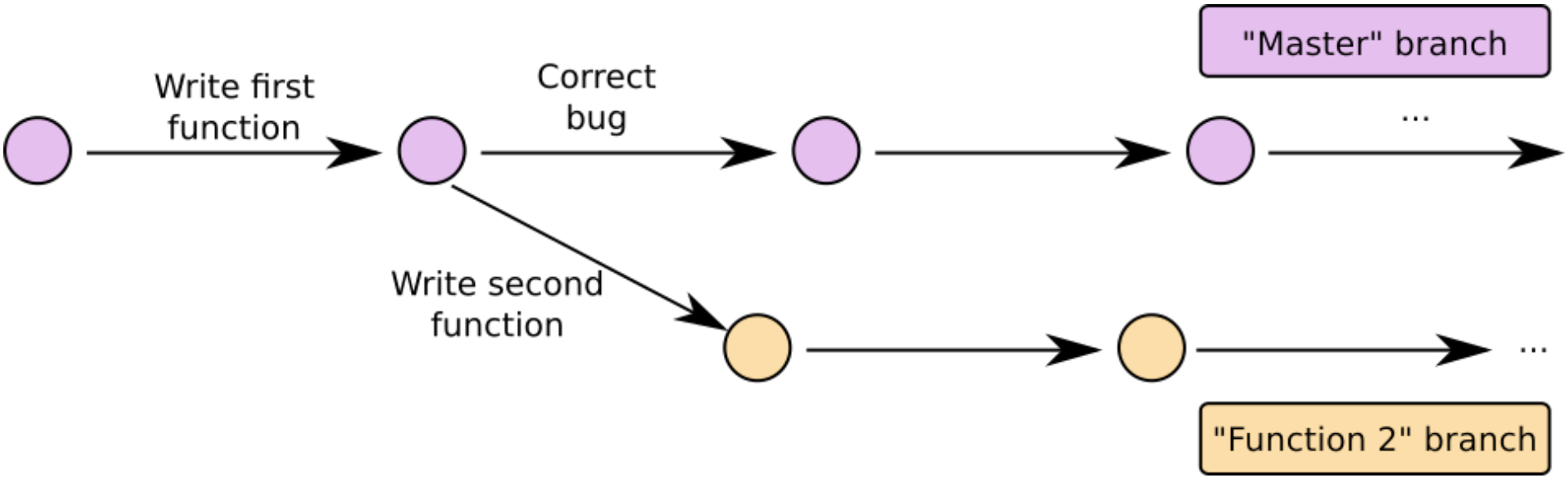
# Branch



# Branch

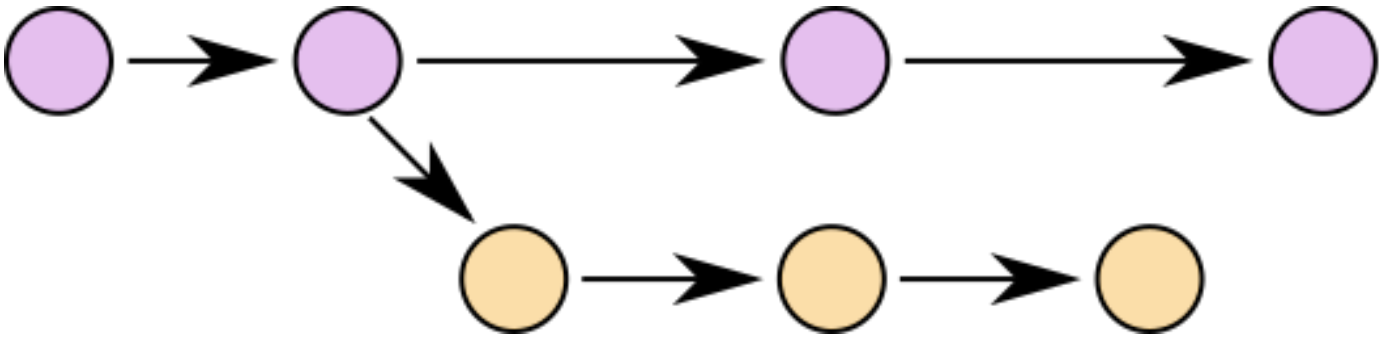


# Branch

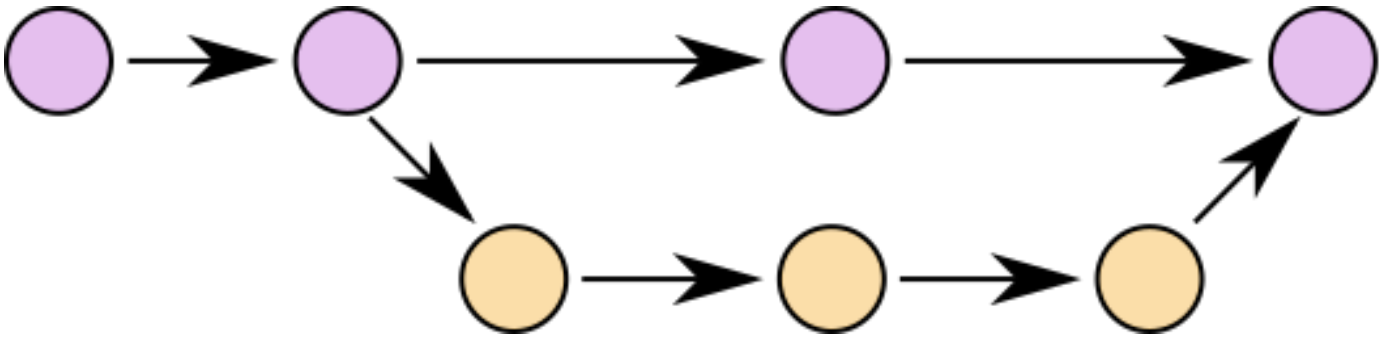

"Master" branch
Write first function
Correct bug
...
Write second function
...
"Function 2" branch

# Merge

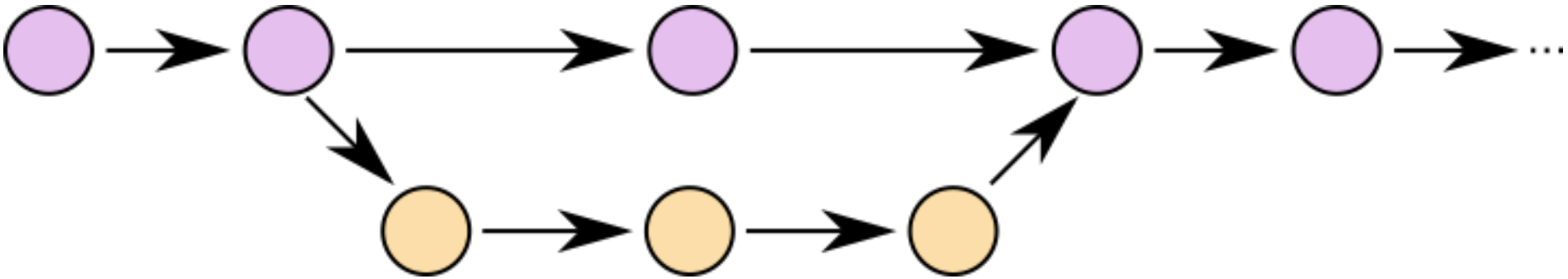
# Merge



# Merge

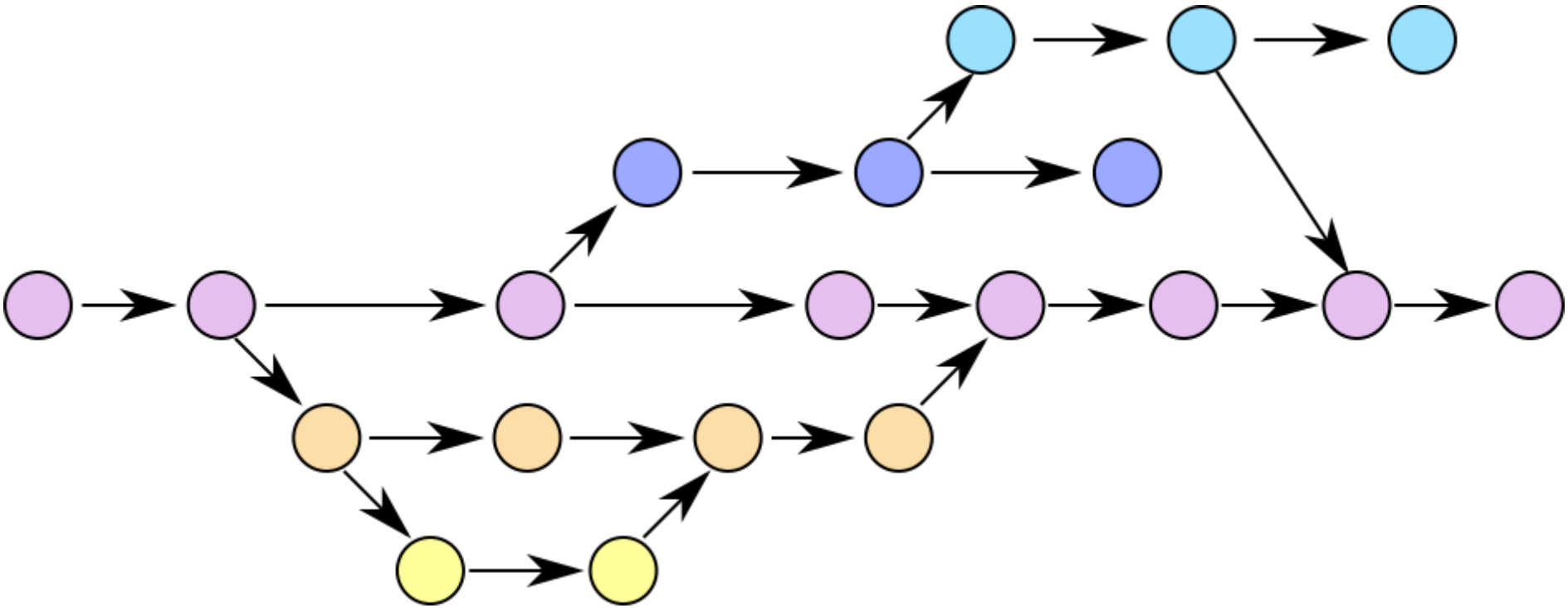


# Merge



# What happened here?

# What happened here?



# Advantages of version control

What are the advantages of using version control systems?

# Advantages of version control

What are the advantages of using version control systems?

- Can try out options (can always revert to last version that worked)
- Documents development of a project
- Clean system for keeping track of changes
- Allows parts of the project to be addressed separately
- Allows several people to collaborate
- Keeps track of contributions

# Outline


1. Idea behind version control
2. Basic concepts of version control
3. Version control in this course

# Why do we use version control in this course?

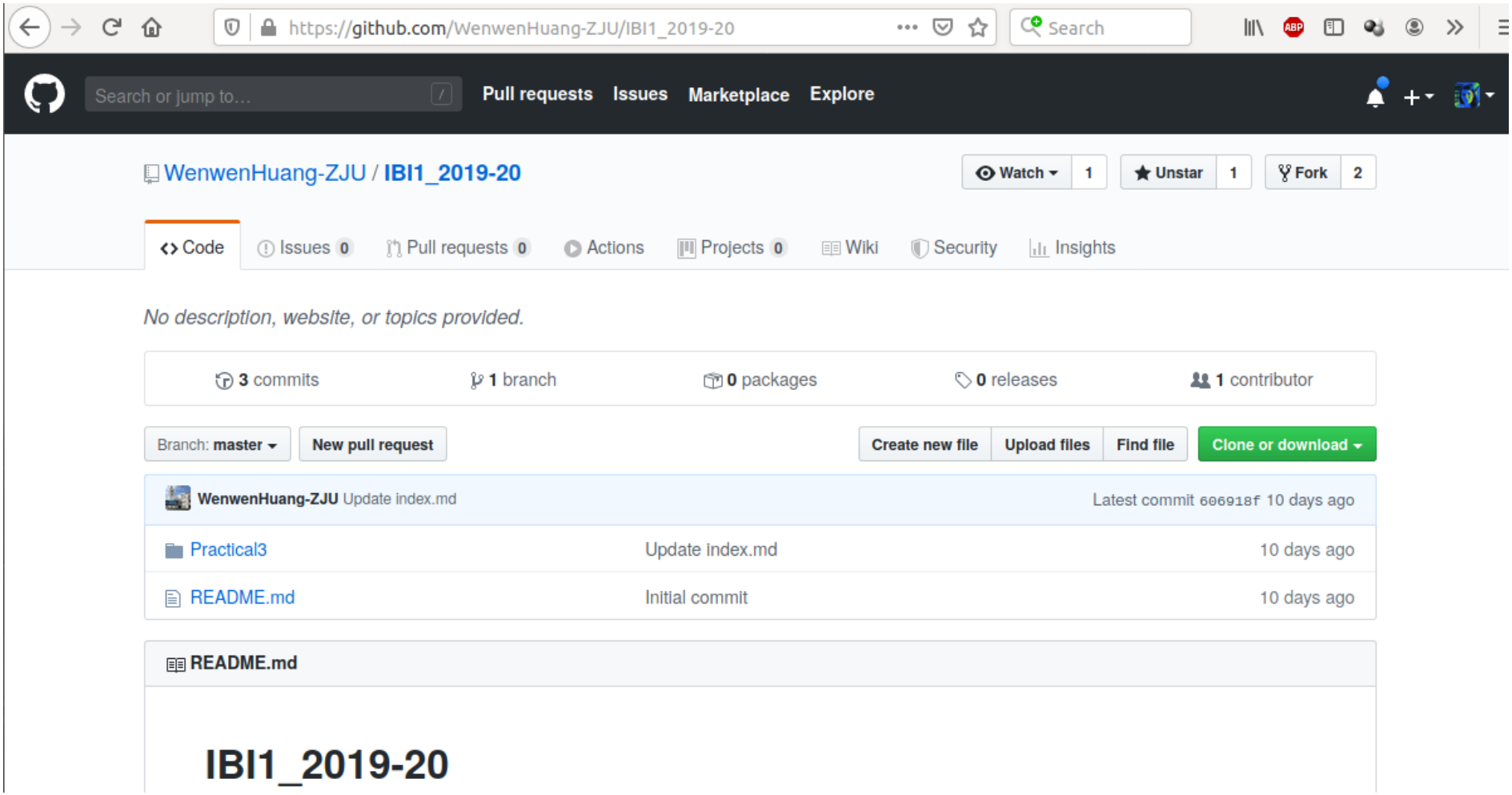
# Why do we use version control in this course?

- Key professional skill
- Early and frequent practice
- Versatile tool that can be of use in your other courses/work
- Allows you to collaborate on projects
- Allows us to see how you have progressed
- Allows us to see how you react to feedback

# How do we use version control in this course?

# How do we use version control in this course?

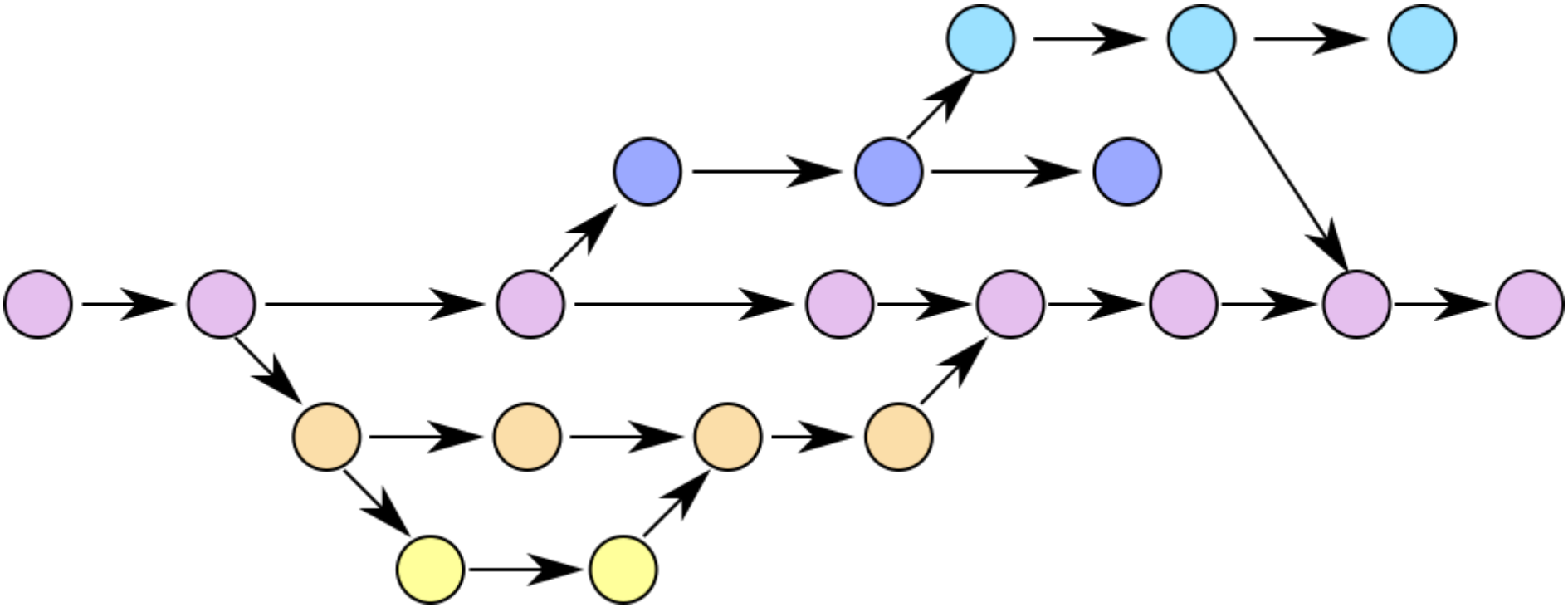
GitHub repository for practical work



# Review

## Now, you should be able to . . .

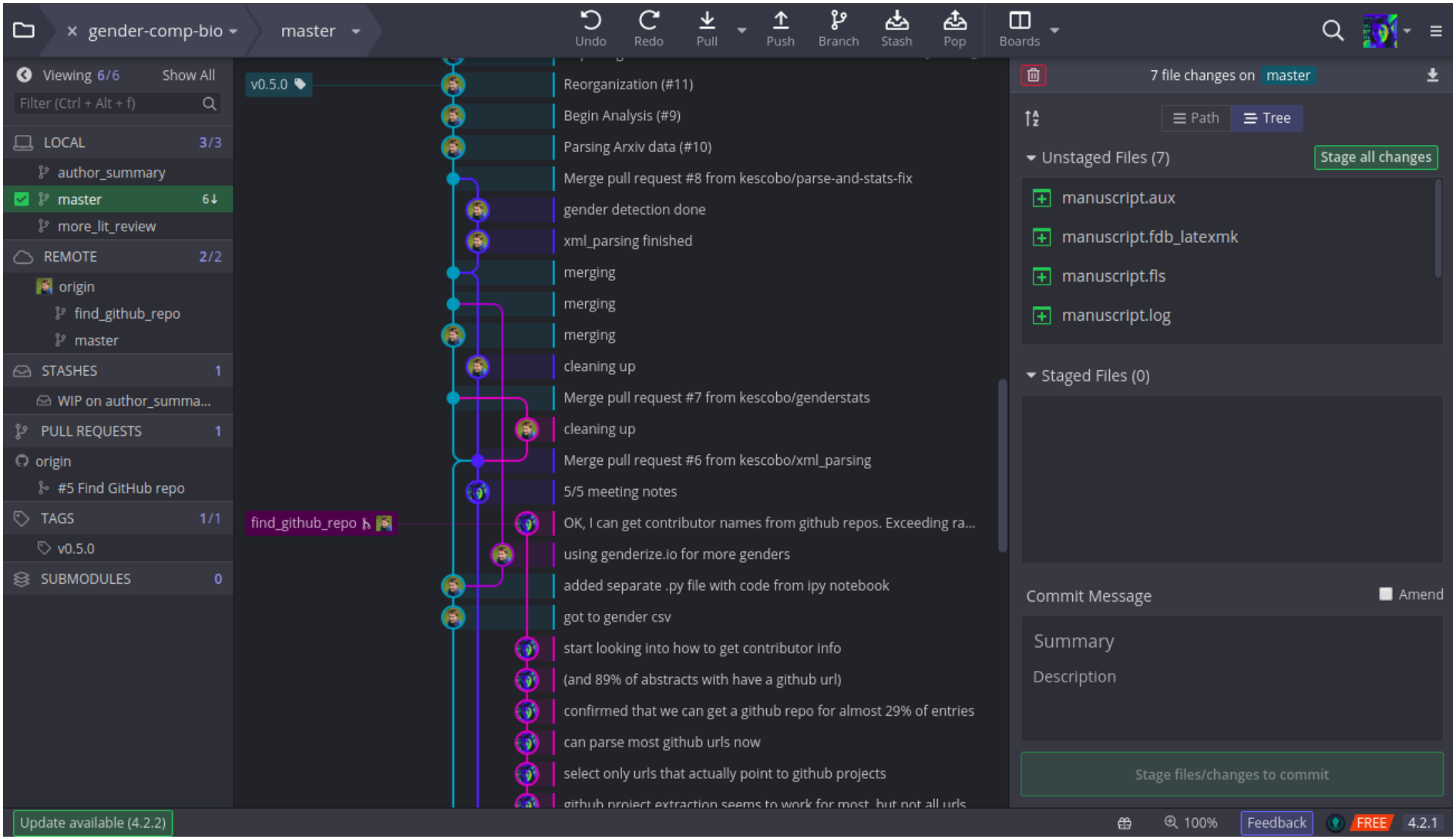
Explain the principles of version control



# Next . . .

# Next . . .

## How to actually do it



# Image credits

- Collaborative work on a text file, with edits from different authors shown in different colours. My own screenshot, 2019. The document is an early draft of Stefan *et al.*, PLoS Comput Biol, 2015.
- Diagrams of version control processes. My own work, CC BY-SA 4.0, 2019.
- "Final".doc (2012). From "Piled Higher and Deeper" by Jorge Cham. `www.phdcomics.com` With permission.
- Screenshot from a project with the Gitkraken software, 2019.
- Screenshot of IBI1 GitHub repository, 2020.
- Two students collaborating on their computers. By Yuuki Guzman and Agoston Tyll (Okinawa Institute of Science and Technology), 2015.