Git Fu

"AS YOU THINK, SO YOU SHALL BECOME."
-BRUCE LEE
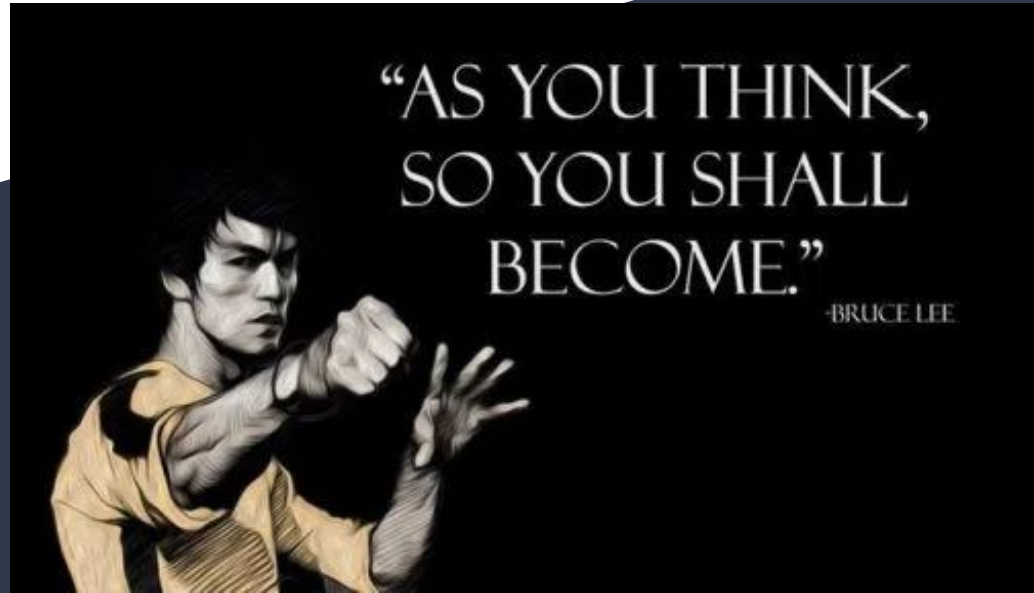
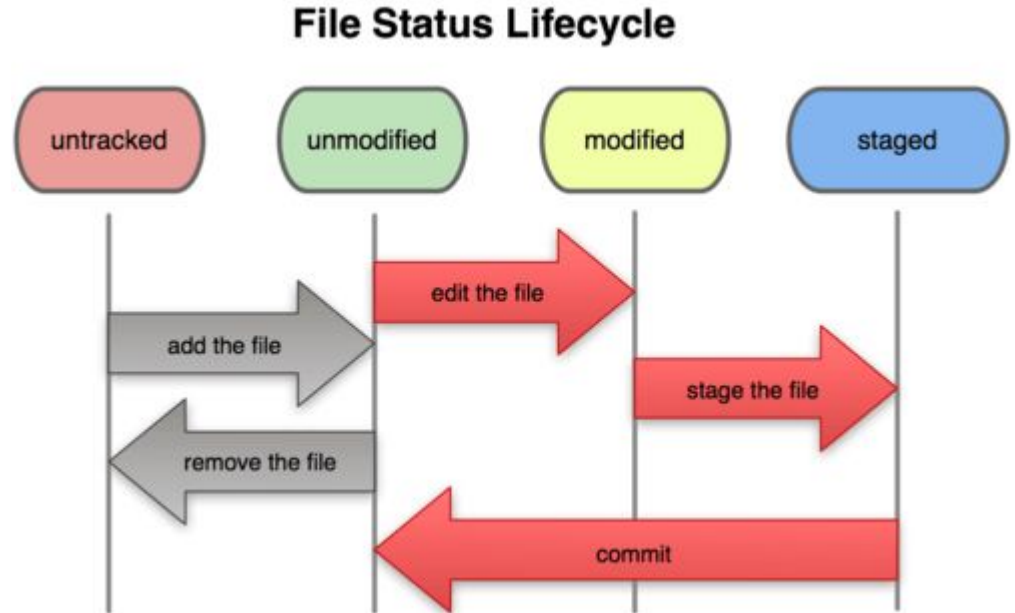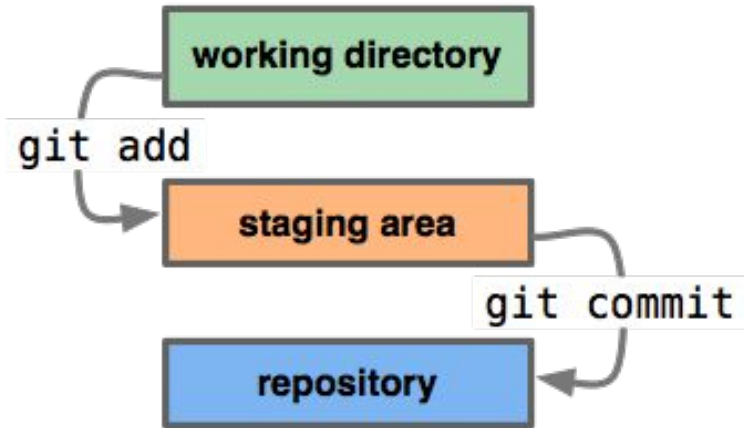# Version Control?
Git?

Version Control:

- A complete record of all changes
- Can roll back to a previous version

Git:

- An open-source Version Control system
- Git is used to maintain the core linux kernel

Staging and Commits

Forking   and   Branching

Take **someone else's** *open-source* code and **modify it** for your own purposes

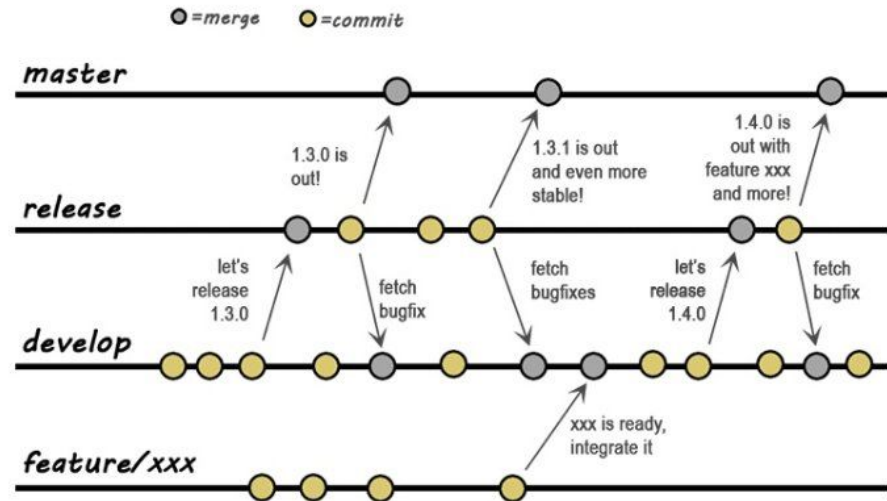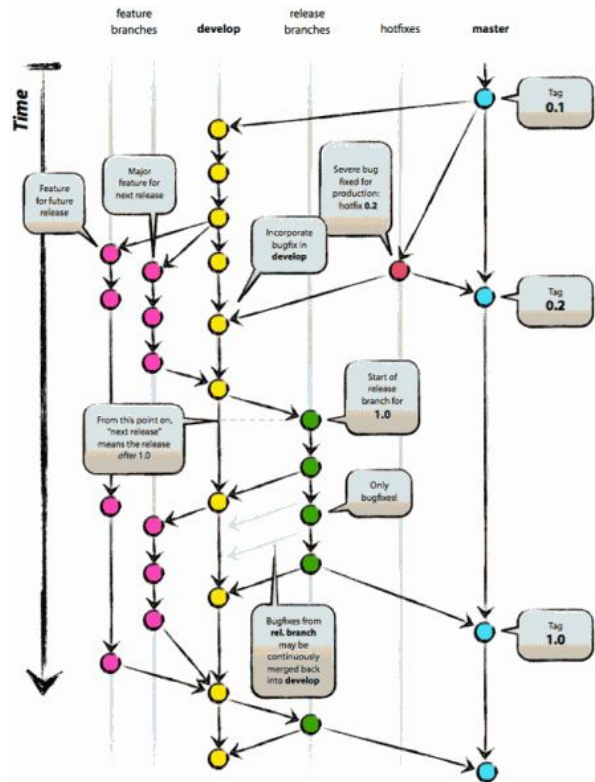Source code **owner** can **approve** a **Pull Request**

Internal developers can "branch off" into another **side project** without affecting the production code

Is **mainly** used to **fix issues** and **bugs**

Branches can be **merged together**

Git has **one master branch** (the origin) where every other branch **originates from**
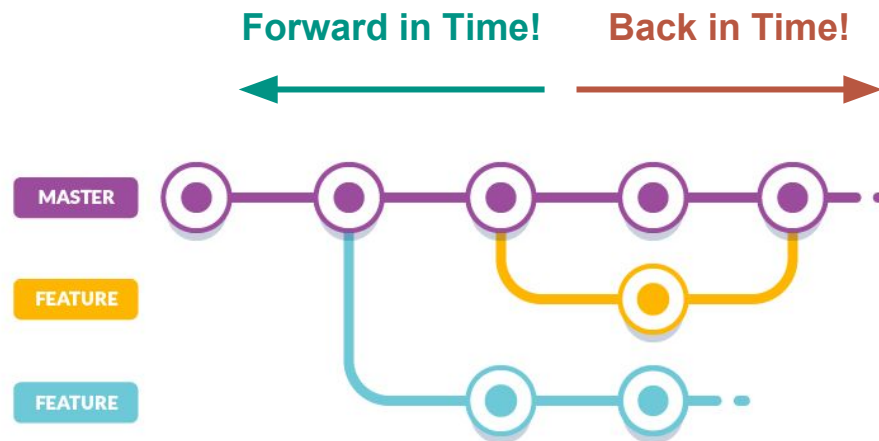
Branching visuals

Git Workflows: Basic

- **Always** the **start** of product development

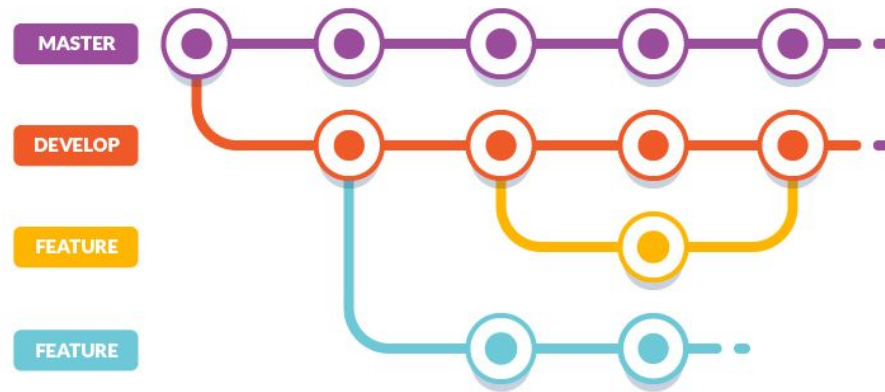- **Simple** and **quick** to set up

MASTER

Git Workflows: Feature Branch

- The **basic** workflow **is limited** in workflow freedom

- For each **new feature**, create a **new branch**

- **Merge** the feature branches **back into** the **main**

Forward in Time!   **Back in Time!**
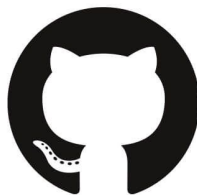
Git Workflows: Gitflow

- Invented by **Vincent Driessen** in 2010

- **Two** parallel **branches**: **Master** and **Develop**

- **Features** are still on **their own branches**

Image courtesy of https://buddy.works/blog/5-types-of-git-workflows

Git Made Easy (no command line!)

- It's totally **up to you**

- Each has its **pros** and **cons**