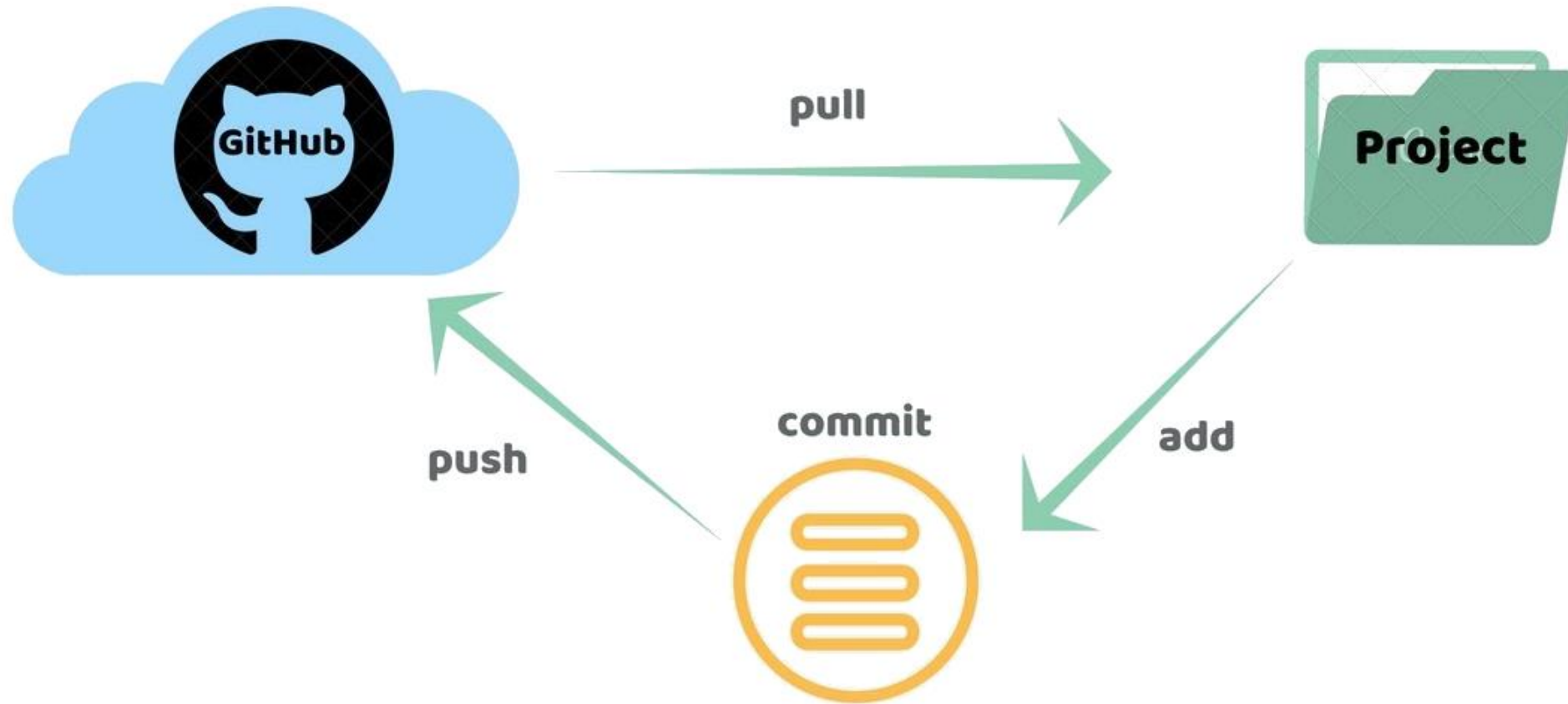


# Basic tools : Introduction to GitHub

KASHTANOVA Victoriya

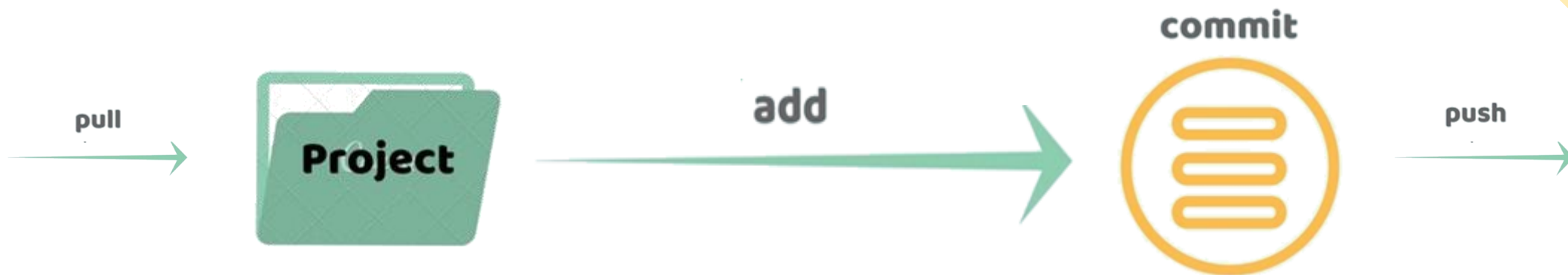
Inria, Epione

# GitHub working flow (personal repository)

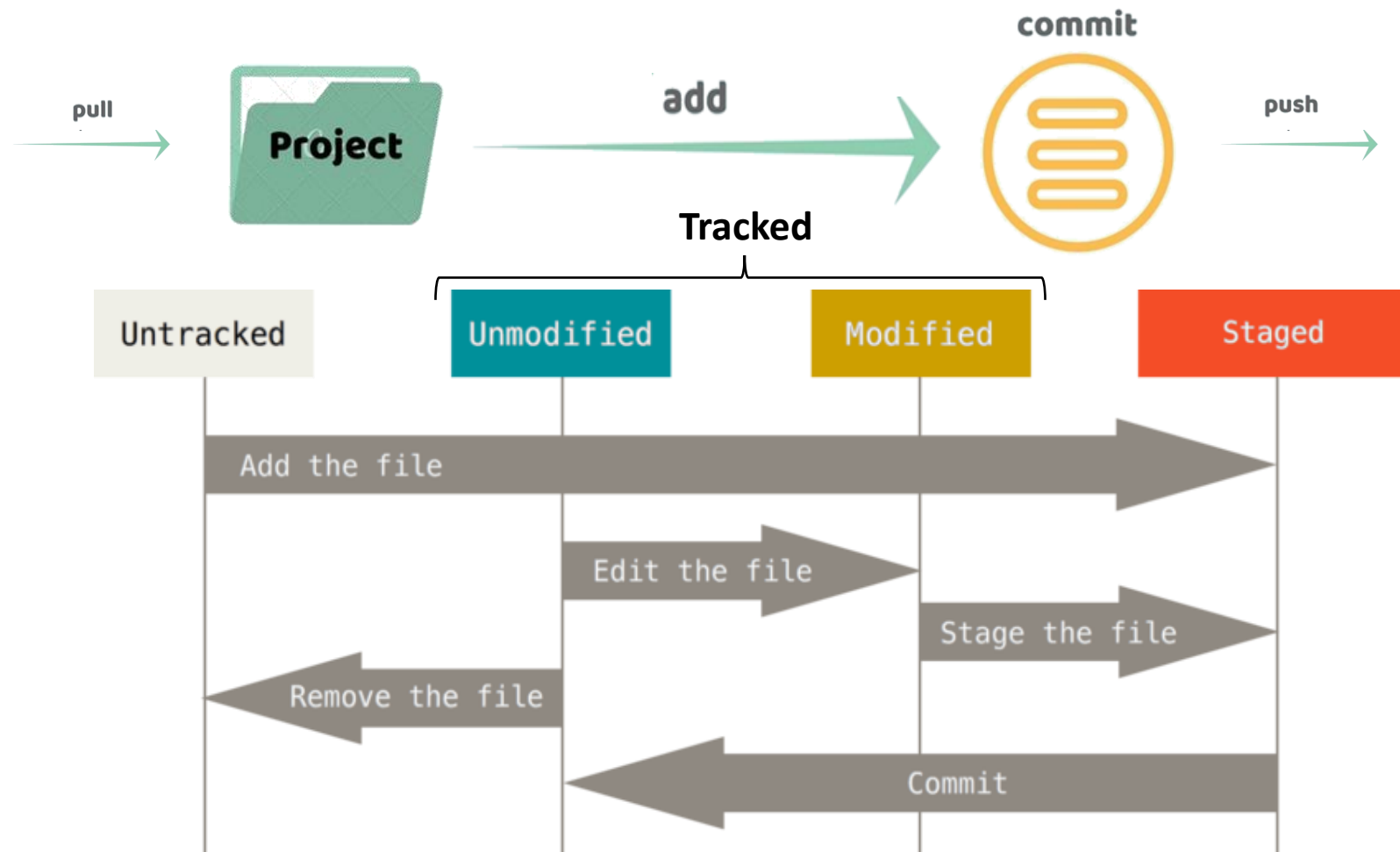


# Local directory working flow

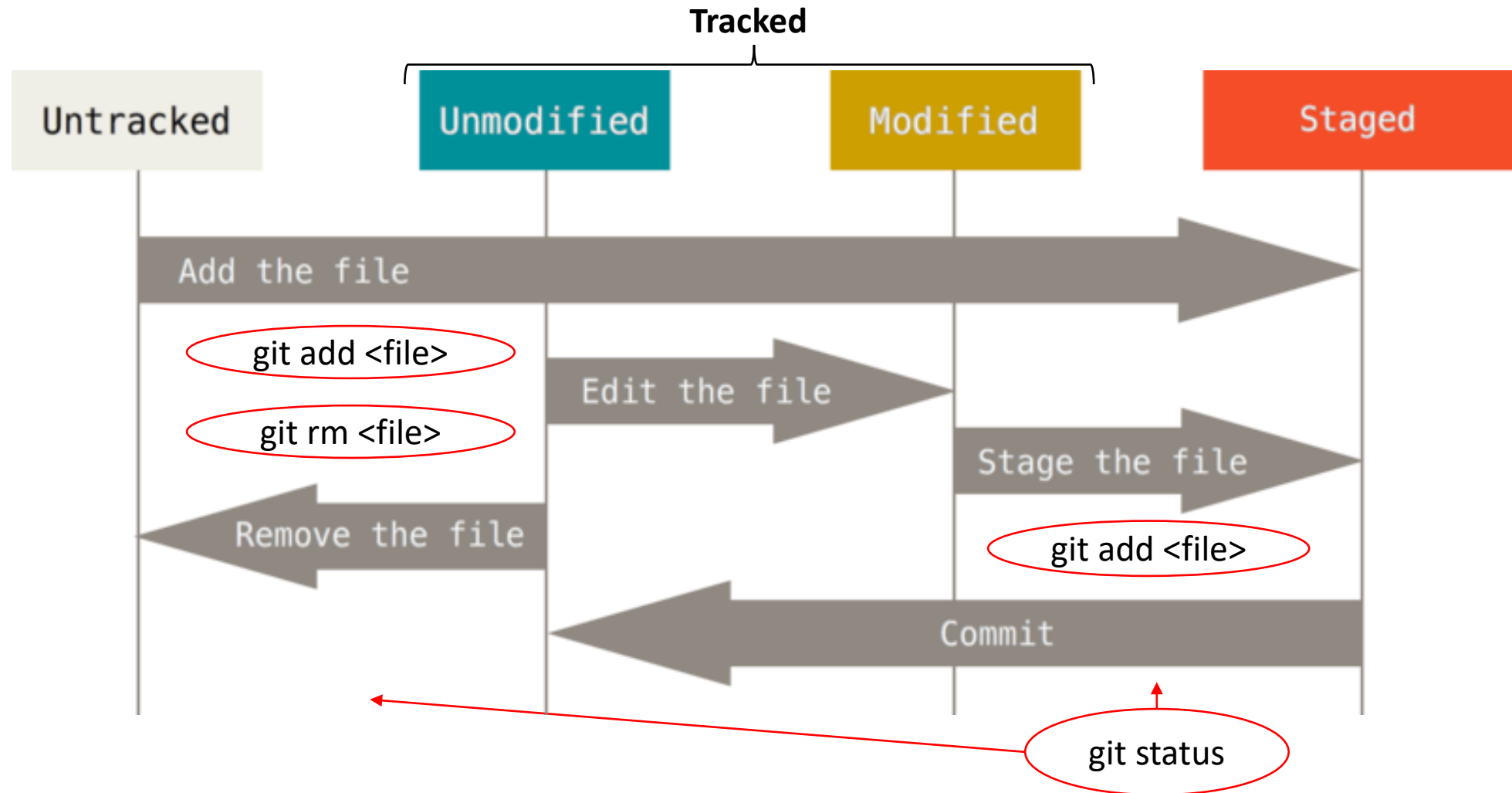
---



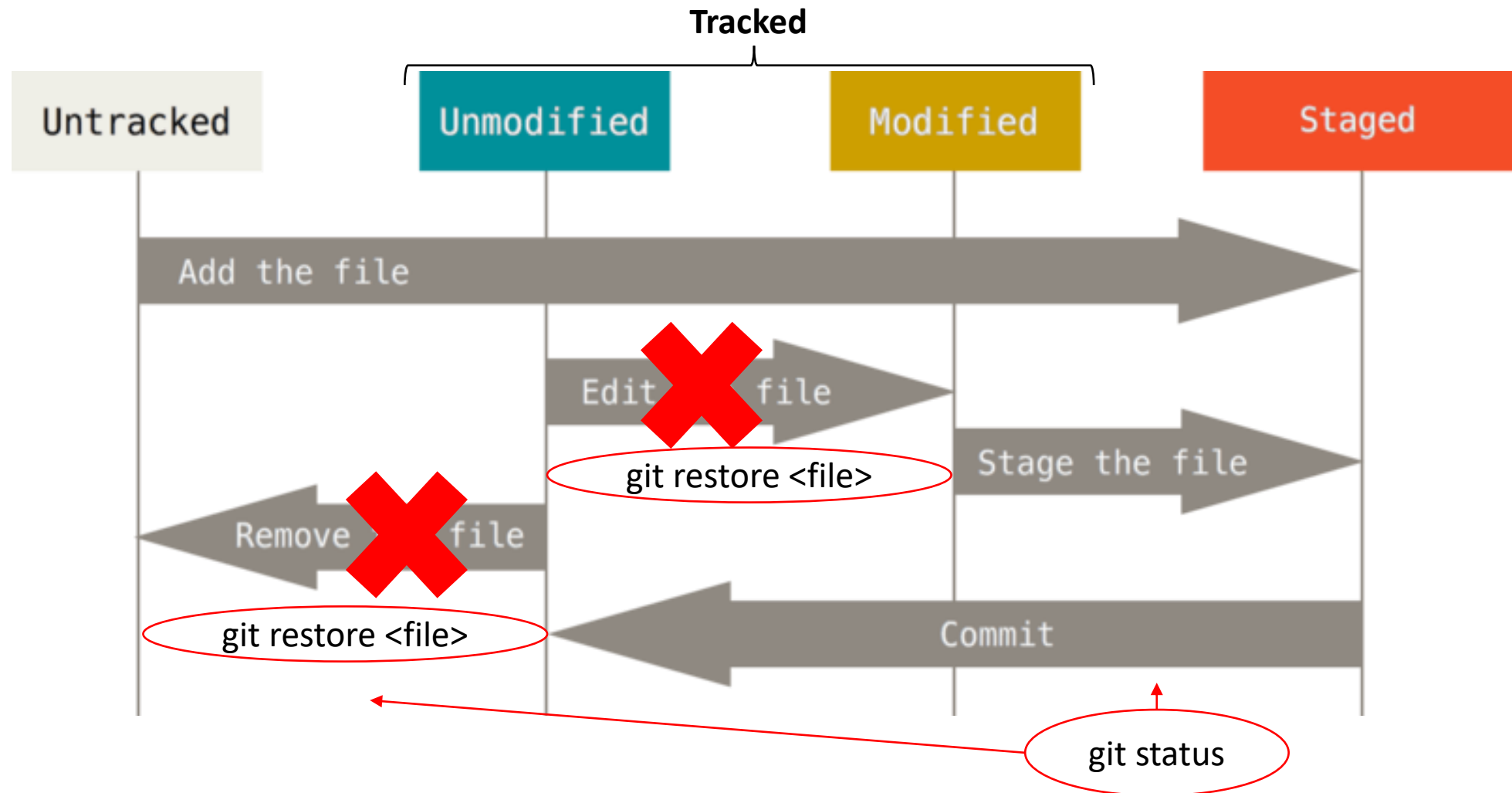
# Local directory working flow



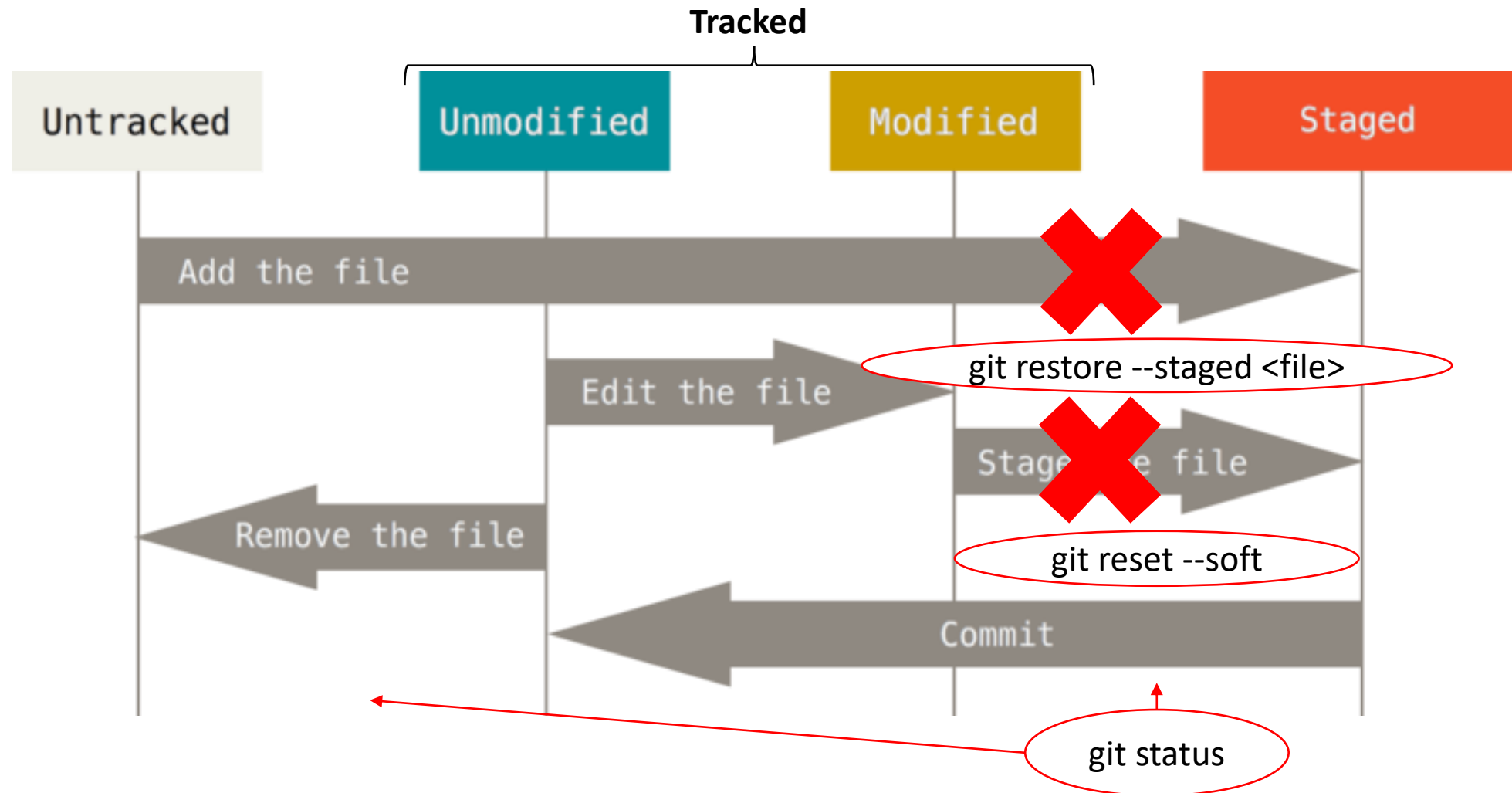
# Local directory working flow (making changes)



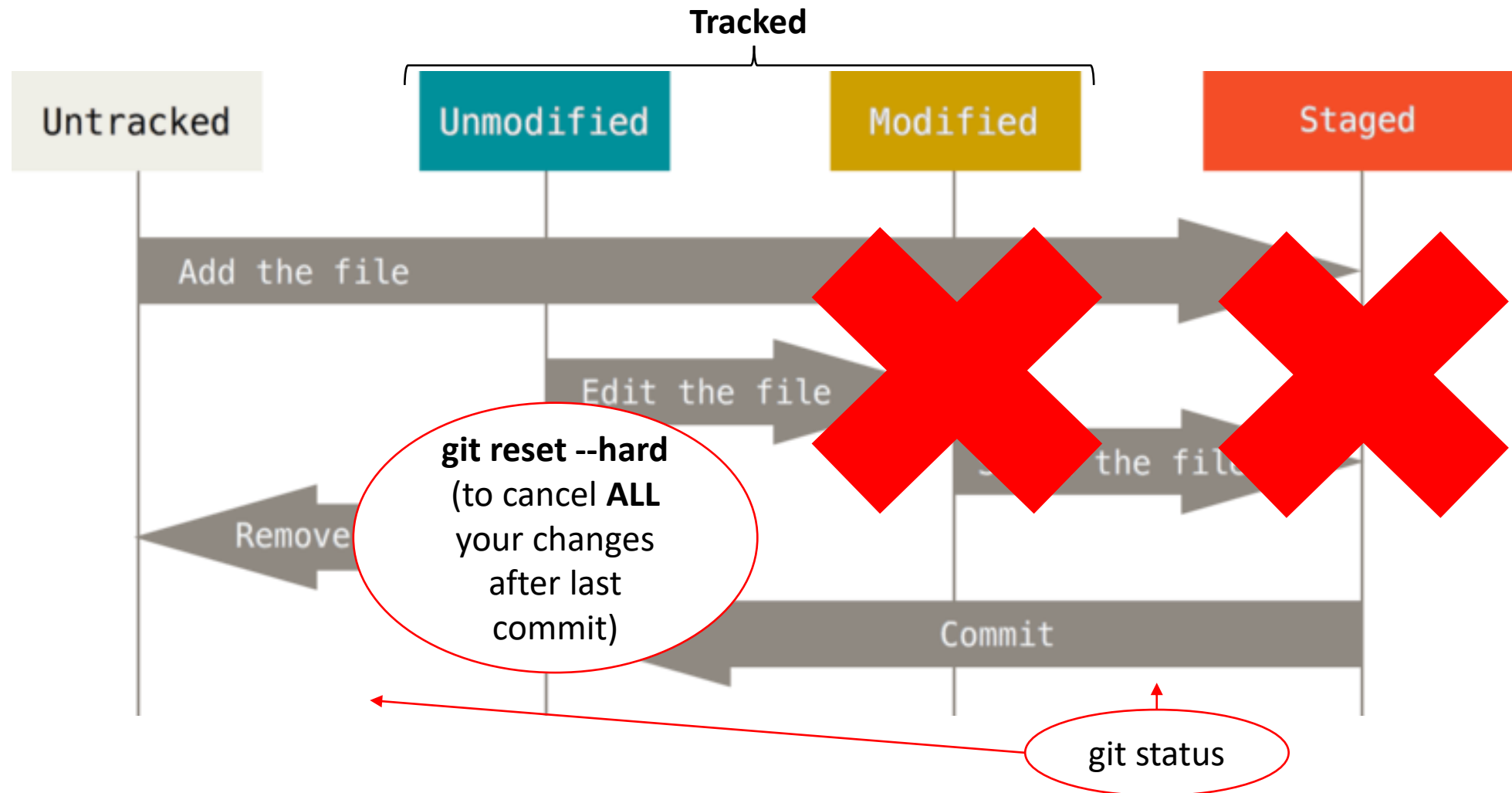
# Local directory working flow (canceling changes before saving)



# Local directory working flow (canceling changes before saving)

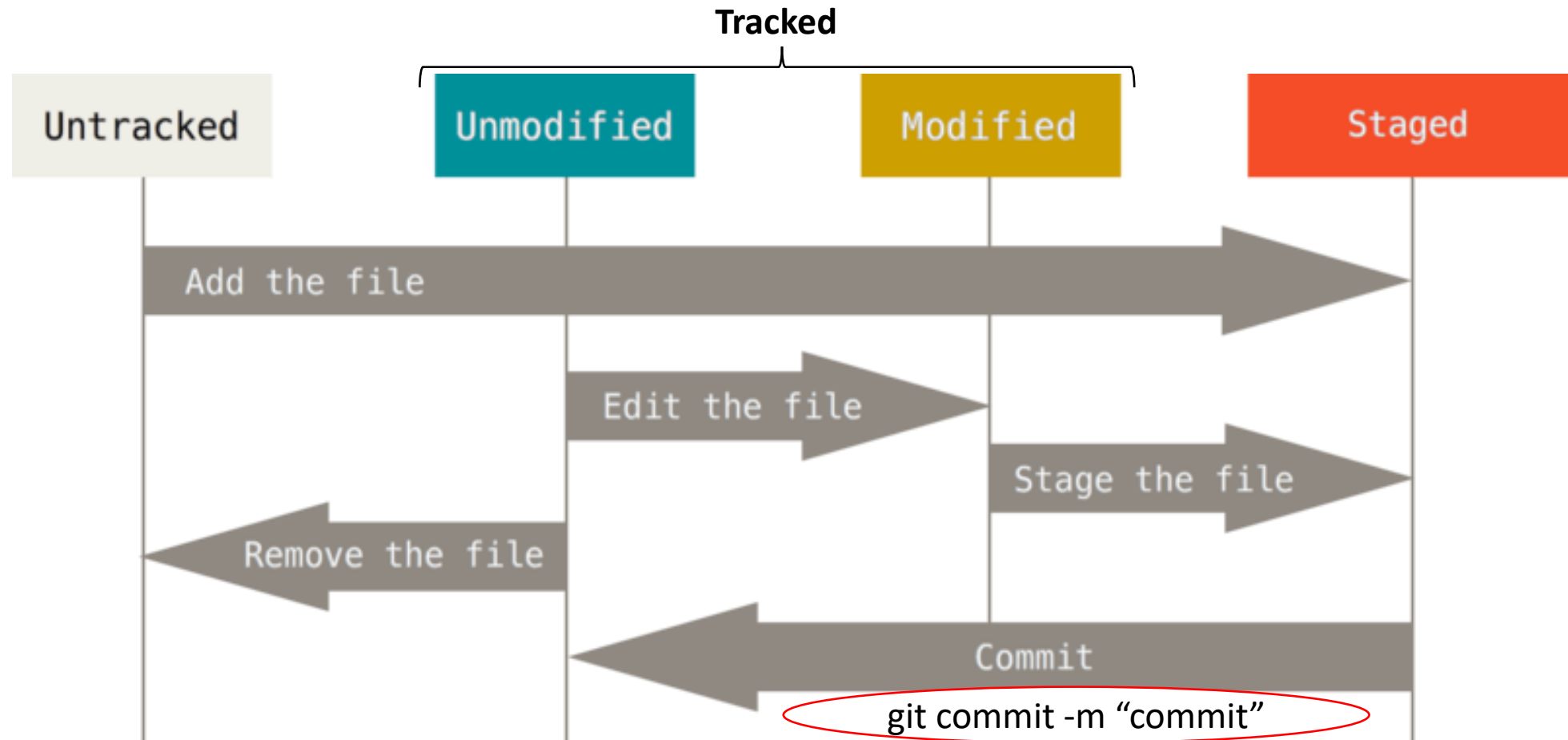


# Local directory working flow (canceling changes before saving)

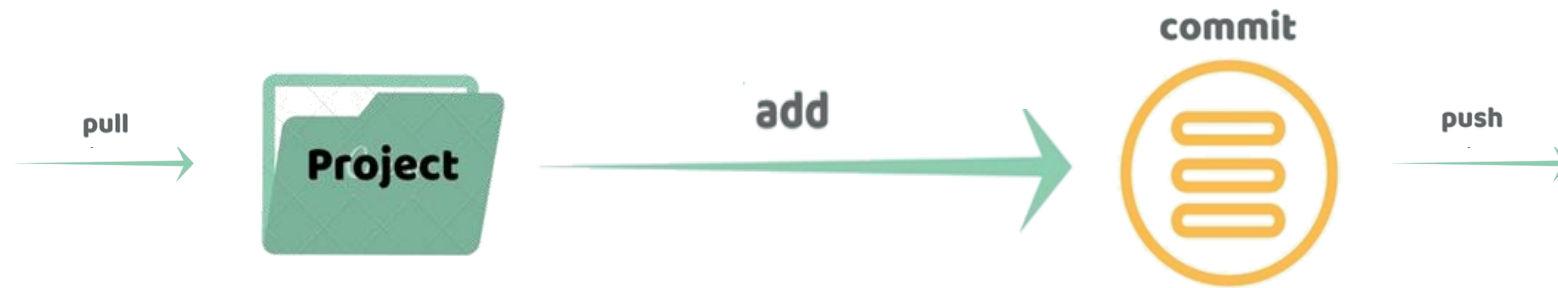




# Local directory working flow (saving changes)

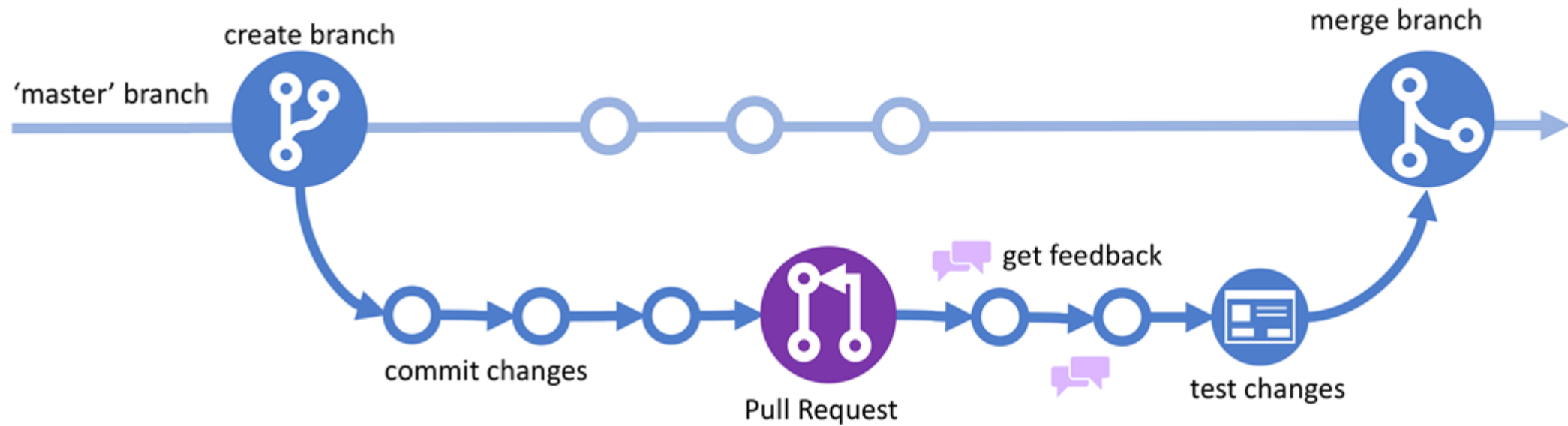


# Local directory working flow (canceling changes after saving)



- **git commit --amend "new commit"**  
(to correct last commit's message)
- **git revert HEAD or <commit hash>**  
(to revert the changes of last commit or of one particular commit)
- **git reset --hard <commit hash>**  
(to reset **ALL** changes after this particular commit and change the history of commits)
- **git checkout <commit hash> + git branch <branch name>**  
(to restore old version of repository as a new branch)

# Global GitHub working flow (branching)



# Global GitHub working flow (branching)

---

- **git branch** - to get list of all existing branches (\* is on working branch)
- **git branch <branch name>** - to create a new branch with indicated name
- **git checkout <branch name>** - to change working branch
- **git push --set-upstream origin <branch name>** - to push changes to remote
- **git pull origin <branch name>** - to pull changes from remote
- **git merge <branch name>** - to merge your working branch with indicated
- **git branch -d <branch name>** - to delete indicated branch

If you have some **conflicts** during the branch merging, you need to correct all "conflict" files with text editor and commit these changes

# Aliases

---

- **What is a shell alias?**

A shell alias is a shortcut to reference a command. It can be used to avoid typing long commands or as a means to correct incorrect input. For common patterns it can reduce keystrokes and improve efficiency. A simple example is setting default options on commands to avoid having to type them each time a command is run.

```
alias rm='rm -i'
```

- It is also possible to create an **alias for git** :  
**git config (--global) alias.SHORT\_NAME COMMAND**

```
git config --global alias.last 'log -1 HEAD'
```

# Useful sources

---

- **Cheat sheets** for GitHub <https://education.github.com/git-cheat-sheet-education.pdf>
- Take a look at [https://githowto.com/create a project](https://githowto.com/create_a_project) for **more examples**