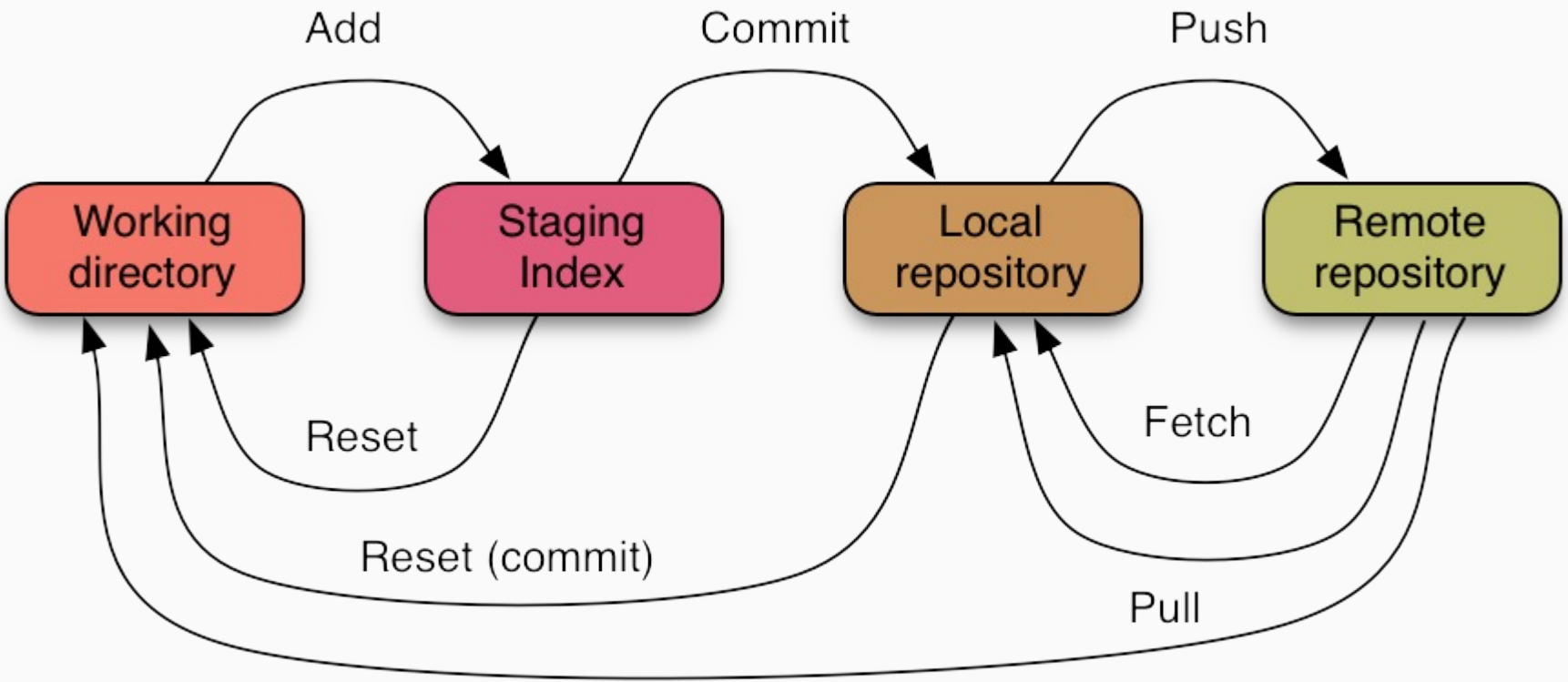
Git & GitHub

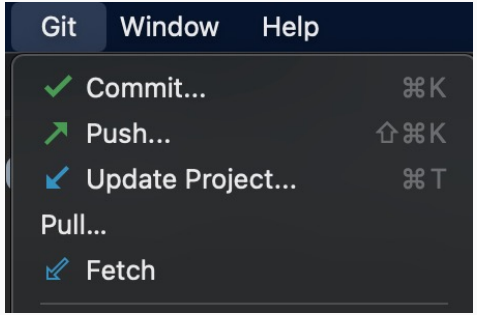
CYDEO


Add
Commit
Push
Working directory
Staging Index
Local repository
Remote repository
Reset
Fetch
Reset (commit)
Pull

CYDEO

Git fetch, update, pull

- To check if the remote repository has any changes that our local repository does not have, we can fetch
    - Fetch will bring the new data without added to our local project right away
- To update your local repository with the new changes from the remote we can pull
    - Pulling is more flexible. We will see later
        - Branch to branch
- Updating will apply changes to the same branch



CYDEO

Branching

Branching in git is like opening a new timeline for new work to not affect what's already in the master branch.

It's light-weight, easy to create and when things do not work you can easily remove it, when it works you can merge your changes into the master branch to get all your work reflected in master.

? What is the master branch ?

https://git-school.github.io/visualizing-git/

CYDEO

Git Terms

- Branch: A copy environment/flow for development
- HEAD: Current branch/commit that is being viewed
  - detached HEAD: when you are checked out to a commit
    - Do not do any changes on a detached HEAD, only on branches
- Checkout: Switch the HEAD to a branch/commit
- Merge: Combine changes from branches together

CYDEO

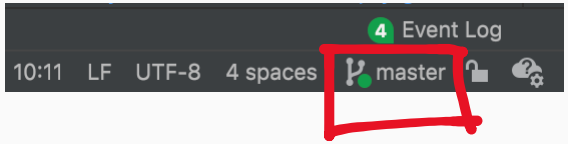
Git in IntelliJ – Make a branch

- There is multiple ways to use IntelliJ to create branches
  - Click the git menu -> new branch

    or

  - Open the git log -> right click on the master branch -> new branch from selected

- Which branch are you on?
  - Easiest way to check is to look in the bottom right corner
    - The example image shows that we are on the master branch



CYDEO

Git in IntelliJ – Use branches

- Create a new branch called feature-1 and checkout to it
- In feature-1 make some commits
    - Check if those commits are shared in the master branch as well
- Merge the new commits from feature-1 to the master branch
    - Checkout to master branch then merge:
        - Right click on the feature-1 -> Merge into selected or
        - Git -> merge (select no ff option for merge commit)

        No option: no merge commit history (ff)

CYDEO

Merge Conflict

- Merge conflict occur when trying to merge but there is some difference between branches or repositories
- What is the difference?
  - The same file having different updates
  - Let’s force a conflict to see an example
- What's the best way to handle conflicts?
  - Avoid them

CYDEO

Reset Commit

Right click on the commit you want to go back to and select Reset Current Branch to Here
Git Reset options:

Soft: file doesn't update, but changes are staged automatically

Mixed: file doesn't update, but changes are not staged automatically

Hard: file will lose all changes until the selected commit

Revert vs Reset: Revert is for pushed commits, history is preserved

CYDEO

Pull Request

- Layer to check if the branch should be merged
- Usually used in a team flow where peer review is done before merging
  - Flow: Work in feature branch, push branch to origin, create pull request to merge from branch to another branch
  - Review is done by a team member
- In the pull request you can:
  - Comment/discuss, check individual commits and changes, confirm merges

CYDEO