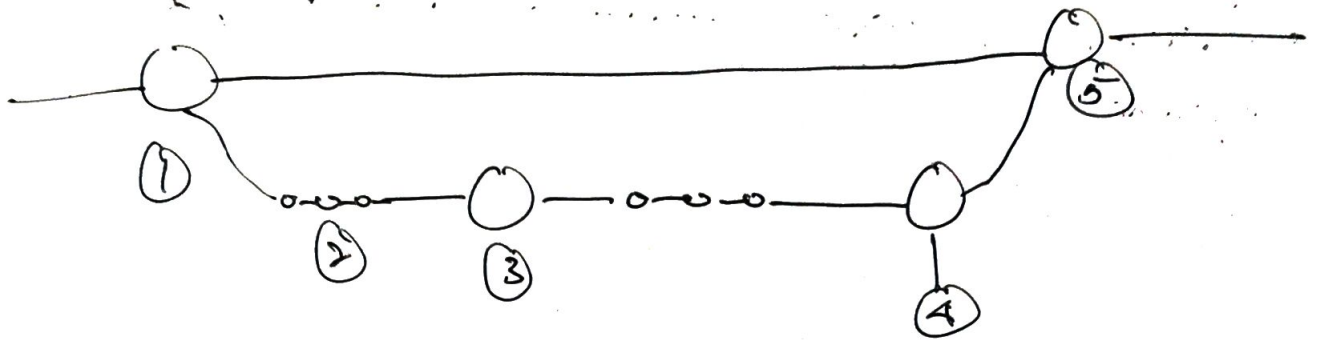


(Cont) Understanding the Github Flow.



④ Deploy

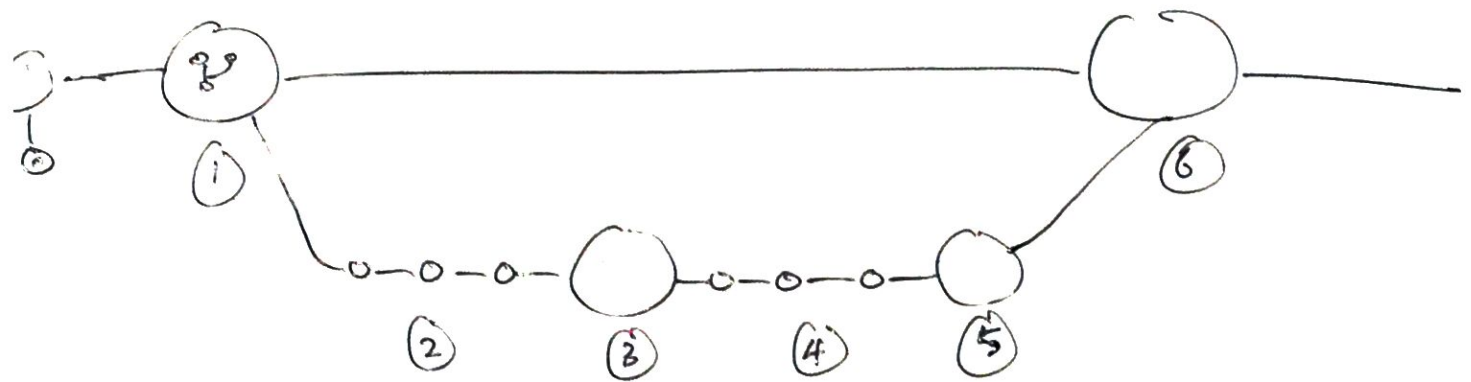
- ↳ In github, you can deploy from a branch for final testing in production before merging to master
- ↳ Once PR is reviewed & the branch passed your tests, you can deploy your changes to verify them in production. If your branch causes issues, you can roll it back by deploying the existing master into production

⑤ Merge : once merged, PR preserve a record of the historical changes to your code. They are searchable, they let anyone go back in time to understand why and how a decision was made.

↳ By incorporating certain keywords into ^{the} text of your PR, you can associate issues with code. When your PR is merged, the related issues are also closed.

↳ eg. Entering the phrase Closes #32 would close issue number 32 in the repo.

Github Workflow Summary



① Github repo → Add collaborator → ^{git clone} mkdir ~~the~~ folder → cd folder
→ git init → touch index.html ~~index~~ index.css ... → git push — master

② At folder, git branch <branch-name>, git checkout <branch-name>
→ (add sth...) , git push link <branch-name>
^{↑ feature-branch}

→ ~~git pull~~ (other) git push link ^{his} <branch> → merge to feature-branch on github

→ git pull link feature-branch

③ $\frac{1}{\uparrow}$ → README ; raw.github.com for link ;

④ → pull request (add link to the comment) for ~~the~~ code review using form

⑤ Get reply & amend

⑥ k ⑥ fa da! Done & merge.