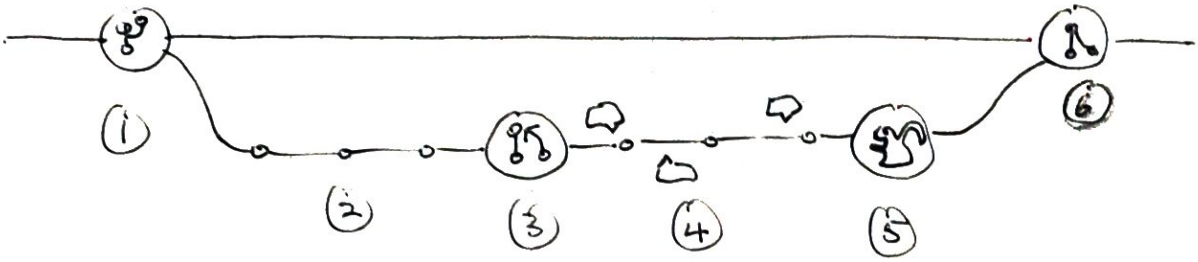
Professional Github Repo

- 3 Requirements
  1) Professional Github repo rules applied
     - README.md file (use template)
     - Meaningful name for pull requests & short summary
     - Descriptive commit messages
  2) Github ~~/ Git~~ Flow is used
     - Github Flow (Feature Branch Flow)
       - A lightweight, branch-based workflow



① Branch :- an environment where you can try out new ideas
- Free to experiment & commit changes
- 1 rule : everything in master branch is always deployable
- Descriptive Branch name

② Add Commits : A transparent history of your work that others can follow to understand what you've done & why
- Each commit has an associated commit message.
- Each commit is a seperate unit of change
- lets you roll back changes if a bug is found, or if you decide to head in a different direction.
- Commit messages are important, by writing clear commit messages, you can make it easier for other people to follow along and provide feedback.

-//-#- Professional Github Repo

(3) Open a Pull Request

- ↳ Initiate discussion about your commits
- ↳ Tightly integrated with the underlying Git repo, anyone can see exactly what changes would be merged if they accept your request.
- ↳ Can open it at any point during the development process
- ↳ To ask feedback from specific person / teams, use Github @mention system.
- ↳ Useful for contributing to open sources & managing changes to shared repo.
- ↳ For Fork & Pull model, Pull Requests provide a way to notify project maintainers about the changes you'd like them to consider.
- ↳ For Shared Repo model, Pull Requests help start code review and conversation about proposed changes before they're merged into the master branch.

(4.) Discuss & Review Your Code

- ↳ Questions / comments for person / teams reviewing.