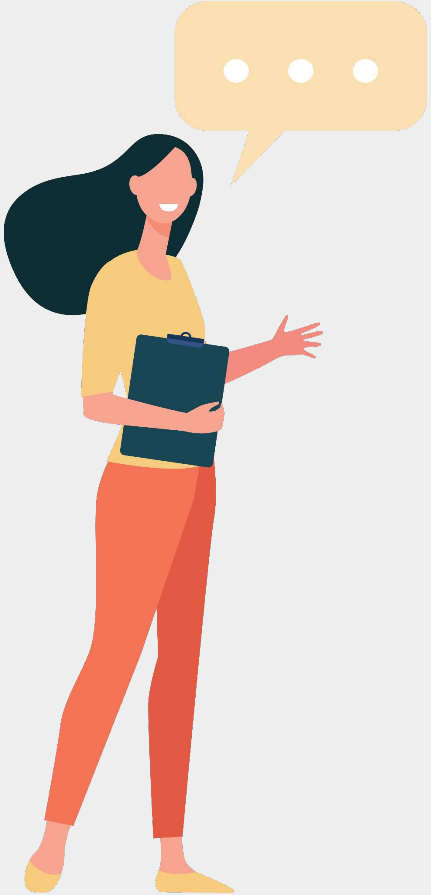
git



What is git?

- Git is software for tracking changes in any set of files, usually used for coordinating work among programmers collaboratively developing source code during software development. Its goals include speed, data integrity, and support for distributed, non-linear workflow. (thousands of parallel branches running on different systems)
- Git was created by Linus Torvalds in 2005 for development of the Linux Kernel with other kernel developers contributing to its initial development.

Git

Git project can be thought of as having three parts:

1

working directory: where you’ll be doing all the work: creating, editing, deleting and organizing files.

2

staging area: where you’ll list changes you make to the working directory

3

repository: where git permanently stores those changes as different versions of the project.

Git

- To make a directory a git project run: “git init”
- Git status: check the status of those changes that you already made
- Git add filename: git start tracking that filename changes
- Git diff filename: check the differences between the working directory and the staging area
- Git commit -m “initial commit”:
  - Last step in git workflow
  - Write the message in double quotation marks (50 character or less)
- Git log: show the list of commits

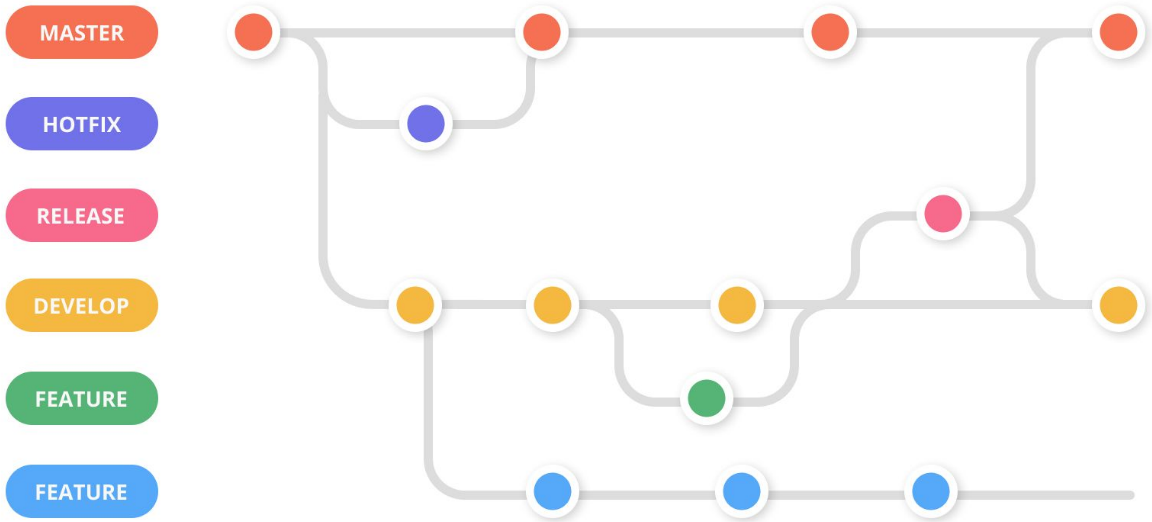
Git

- Git show HEAD
    - In git, the commit you are currently on is known as the HEAD commit
    - The most recently commit is the HEAD commit
    - Will display everything the git log command display for the HEAD commit, plus all the file changes that were committed.
- Git checkout HEAD filename
    - Will restore the file in your working directory to look exactly as it did when you last made a commit
- Git add filename_1 filename_2
    - Add the changed files to the staging area
- Git reset HEAD filename
    - Resets the file in the staging area to be the same as the HEAD commit
- Git reset commit_SHA
    - Reset the index and working tree to the desired tree

Git

- Git branch
  - show list of branches
  - Default branch is master and showed by *
- Git branch new_branch
  - Create new branch
- Git checkout branch_name
  - Change your active branch
- Git merge branch_name
  - Resolve merge conflicts
- Git branch -d branch_name
  - Remove branch

Git



Git

A remote is a shared git repository that allows multiple collaborators to work on the same git project from different locations.

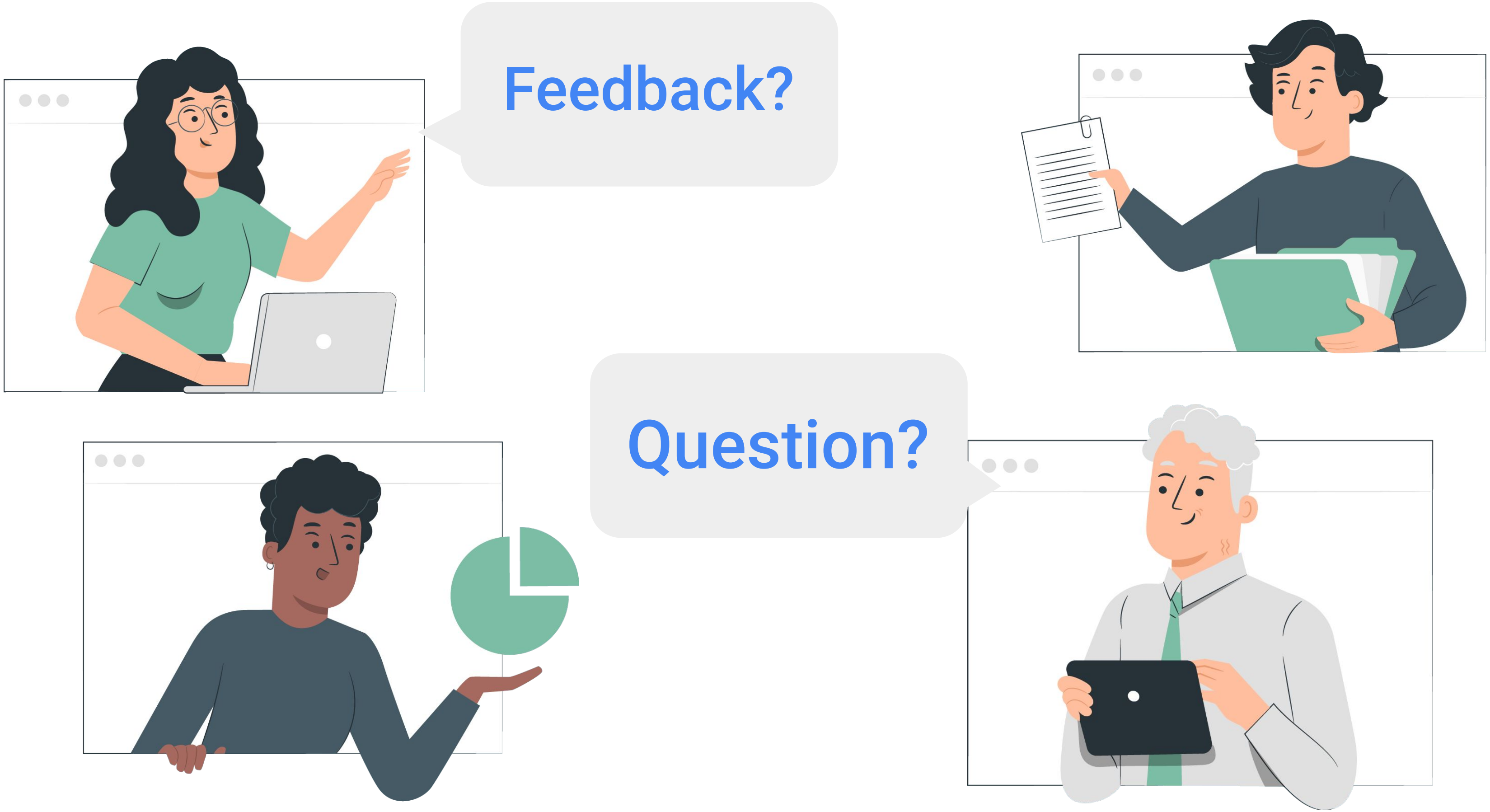
- Git clone remote_location clone_name
  - Remote location could be file path or web address
  - Clone name is the name you give to the directory in which git will clone the repository
- Git remote -v
  - Show remote origin
- Git fetch
  - Get all changes from remote and keep it in origin/master
  - You should merge it manually
- Git push origin your_branch_name
  - Will push changes to the remote origin

Git

- Config your local project:
  - git config user.name Mostafa
  - git config user.email test@gmail.com
- You config this information with --global option to make it available everywhere:
  - git config --global user.name Mostafa
  - git config --global user.email test@gmail.com
- You should define `access token` on your github account.

Git

- More topics:
  - Git workflow
  - Git Hooks
  - Merging vs. Rebasing
  - Pull Request


Feedback?
Question?

Thank You!