



PREPARATION

Internet Programming

DEVELOPMENT REQUIREMENT

You are required to have the following tools ready:

- **Browser** : Front-end application requires browser to run
- **Integrated Development Environment (IDE)** : It is important to have IDE related to web development ready. Have a complete IDE will help your development easier.
- **Source Code Storage & Version Control** : Source code cannot be stored just in your computer. It is risky as your computer might be broken or you might delete your code by mistake.
- **Postman** : A very popular software to test your Backend (you will need this in the later lectures)

DEVELOPMENT REQUIREMENT

Browser

You should install at least 3 popular browsers:

- [Chrome](#) : You are required to have Chrome installed as it is very developer friendly so most of the time you will work with Chrome
- [Firefox](#) : For multiple testing purpose
- [Microsoft Edge](#) : For multiple testing

Integrated Development Environment (IDE)

For HTML, CSS, and Javascript development (FrontEnd), you can just install [Visual Studio Code](#). Other alternative IDE are also acceptable such as:

- [WebStorm](#) : IntelliJ IDEA family
- [Atom](#)
- [Sublime Text](#) : Popular on mac

VS CODE CONFIGURATION



- Set up vscode as global command line where you can access from anywhere
- Install nessary extensions
- Choose your desired Theme
- Configure your editor

You can find guideline to do above tasks by searching in google “How configure Vs Code for FrontEnd development”

Or use this [guideline](#).

SOURCE CODE STORAGE & VERSION CONTROL

You are required to keep your source code safe, and be able to trace back. We use **Git** for this.



Git is a technology and there are many platform that implement GIT:

- [GitHub](#) : Required. Register an account on GitHub if you don't have one yet

- [GitLab](#)

- [BitBucket](#)

- [Google Cloud Source](#)

[Repositories](#)

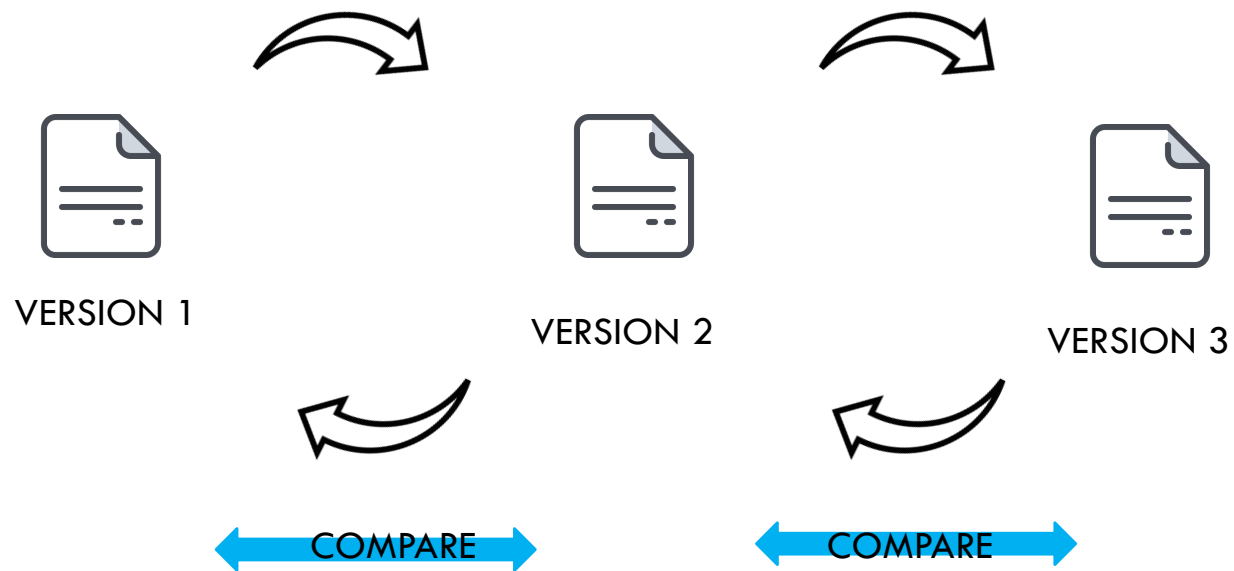
-

WHAT IS GIT?

SOFTWARE KEEPS TRACK OF
CHANGES IN



WHAT IS GIT?



GIT PRACTICE

1. Create an empty directory
2. Create a few text files and input some content and then save them.
3. Initiate this directory as a git repo
4. Make some changes in some files and then save them.
5. Record the new changes and save the information with git, and call it as “version 1”.
6. Make other changes, save them and then commit to git with the name as “version 2”.
7. Undo the last changes, and commit to git as “version 3: undo version 2”
8. Switch seat with your friends, and let them modify some thing. Save them changes as “version 4”
9. Figure out what the changes from your friend.

LET'S GO DEEPER

1. Register a github account
2. Create a repository in GitHub and Link the git repo you work with a moment ago with this repository
3. Push the changes you made to this GitHub repository
4. Invite 1 of your friend to collaborate on the same repository and ask them to modify anything, commit the changes as “version 5” and finally push the changes to the repository.
5. Update your content (make sure you update all the existing files) and commit as “version 6”
6. Update your local data from the GitHub repository (make sure that you have all the contents including the contents that your friend have changed).