TYH Lab: Git and Github workshop

Objective:

Git is a powerful and easy-to-use version control tool, and its extensive application can be from personal, Group projects to corporate database management systems. Although Git is used commonly among computer scientists and software engineers, due to Git's underlying complex ecosystem, aspiring student researchers may find Git's unintelligible output daunting after a few times of initial use. This workshop aims to guide participants to utilize the benefit and breadth of Git usage by taking a peek at Git's structure or one may call under "hood" of how git functions, in addition to learn the basics of managing a personal project on a web repository such as github.

This workshop, however, is not intended to explain all Git's API commands. Git's API is not only too vast but also it will be unhelpful to participants to gain an understanding of git. One should always remember git commands themselves are only one web search away from being understood.

Workshop I: introduction: Git the Big Picture

Purpose: git starting the basic and using git

- The What and Why of the version control system?
- Reproducibility and Open science
- What is Git?

  - A Power Distribution Revision Control System!
  - Rationale of this workshop
    - "Git is easy to use, hard to learn"

- **1st Practical** : Github account and basic setup

Workshop II : Structure and theoretical usage git: Git going!

purpose: Understand how git works by breaking the building blocks of git and its theoretical usage. Ideally one wouldn't be scared of using the command line.

- The Onion Model of git

- Git is Pessistant Map	:	Key and values
- Git is a Stupid content tracker	:	Tags and Commits
- Git is a version control system	:	Branching

- Git is a distributed version control system : Push and Pull

- Dissecting Git functions as a version control system
  - Branching: Merge and rebase
- Git as a distributed version control system
  - Social constructs of git

- **2nd Practical** : branching exercises

Workshop III : **Git Under the Hood: Git working now!**

Purpose: Understand gits commands works, how to ultized git tools.

- The Four Areas of Git
- How to think like Git?
  - How does a command move information across the four areas?
  - How does this command change the repository?
- Demonstrate basic Git commands actions how they move between the four area changes.
  - add, commit, checkout, rm --cached, git reset

- **3rd Practical** : Set up remote updates and solving merge conflicts

Workshop IV: Using a web repository: Git together

- **Purpose:** Integrating what we have learn with github.

- Git into the flow: Find your own Git workflow
  - Model of workflows
- Github demo of systematic workflows

- **4th Practical** : Join behavior analysis on github