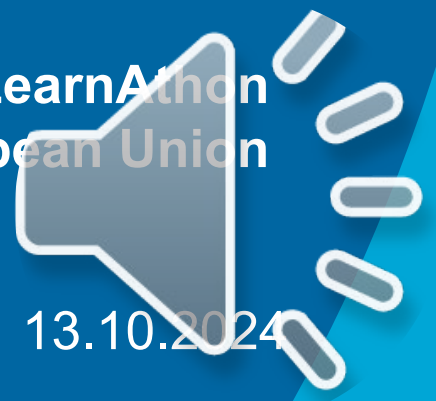
MachineLearnAthon - Microlecture Introduction to Git

Jan Timko

MachineLearnAthon
A project Co-funded by the Erasmus+ programme of the European Union



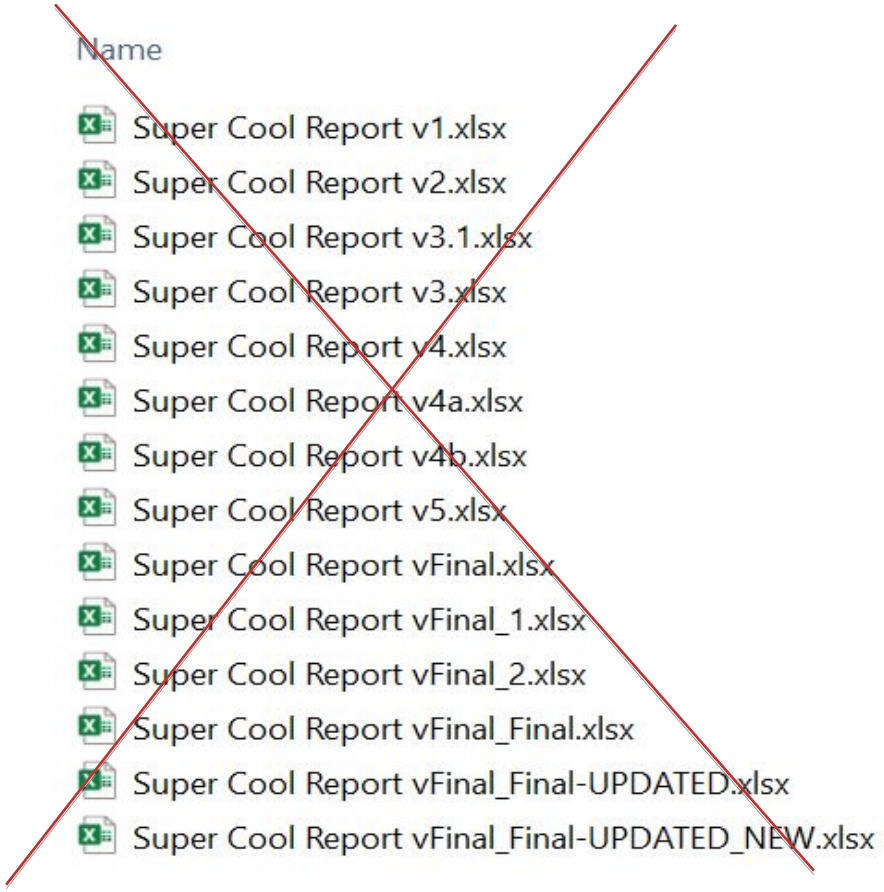
13.10.2024

Agenda today

- What is version control and why we need it
- Introduction to Git
- Key commands
- GitHub and Google Colab

Why we need version control?

Example of saving many files without version control



Not this way!

- **Unsustainable:** Manually saving versions quickly becomes messy.
- **Not Scalable:** Difficult to manage as the project grows.
- **No Change Tracking:** Impossible to see who made what changes or when.
- **Error-Prone:** High risk of overwriting or losing important work.
- **Collaboration Issues:** Hard for multiple people to work on the same project.

source: https://www.reddit.com/r/ProgrammerHumor/comments/oz4pfb/version_controlcan_anyone_relate/#lightbox

What is Version Control

Definition: System to track changes in code or documents over time, allowing collaboration and management of multiple versions.

Key Benefits:

- Collaboration: Multiple developers can work on the same project simultaneously.
- History Tracking: Keeps a record of all changes, who made them, and when.
- Branching: Supports parallel development by creating separate branches for new features/bug fixes.
- Merging: Combines changes from different branches into one.
- Reversion: Easily revert to a previous version if errors are introduced.

Introduction to GIT

A **distributed version control** system for software projects

- Introduced in 2005 by Linus Torvalds
- Essential tool for modern developers
- Integrated into major IDEs such as VS Code or PyCharm

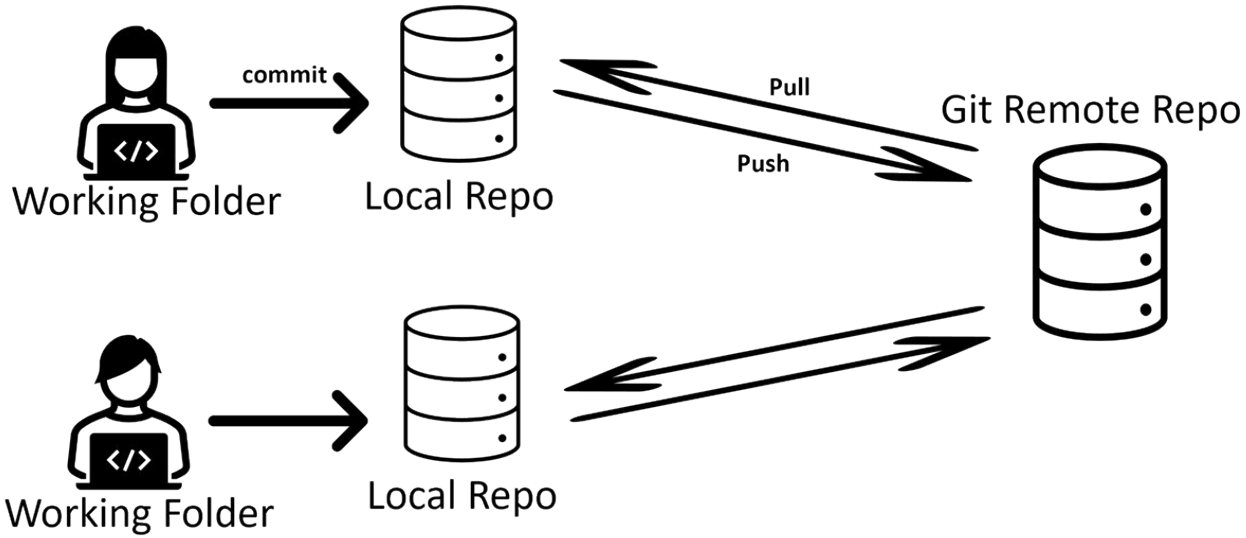
Key concepts

- **Repositories:** Central location to store project files and their history
- **Commands:** Used to commit changes, manage versions, and collaborate with others
- **Branches and Merging**: Enable parallel work on different versions of the codebase and combine changes when ready

Repositories - local and remote

Repository: is a collection of files and folders that Git tracks to monitor changes and versions over time.

Git uses repositories to manage version history for project files.



source: https://se.mathworks.com/help/matlab/matlab_prog/source_control_git.png

- **Remote Repository:** A version of the project stored on a server (e.g., GitHub, GitLab).‘
- **Local Repository:** A copy of the project saved on your computer.

Key Git commands

- *git clone* – Clone a remote repository to a local folder
- *git commit* – Send changes to the local repository
- *git push* – Send changes to a remote repository
- *git status* – Check the current branch's status (e.g., changes, staged files)
- *git checkout* – Switch branches or restore files
- *git fetch* – Fetch updates from the remote repository (e.g., new branches or commits)
- *git pull* – Fetch and integrate (pull) changes from a remote branch into your current branch

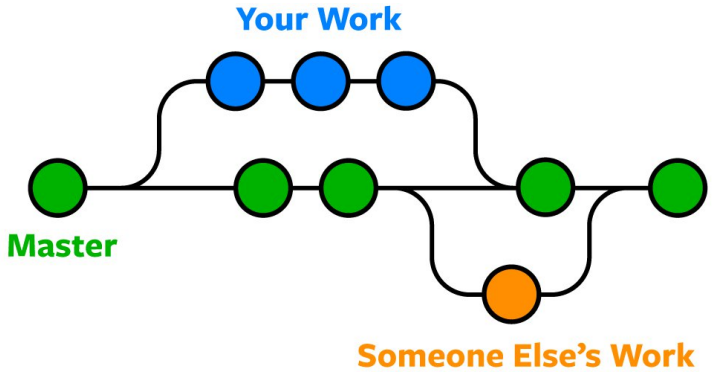
Branches and merging

In order to better organize flow of the work, we are usually using different branches

Main/master – the most important branch, keeping the recent production version of the project. Cannot be deleted

Project branches – created for implementing new functionality, modification or fix.

- New code is firstly developed in here, and only once tested and approved it is merged to **main/master**



https://www.nobledesktop.com/learn/git/git-branches

Examples:

- *new-function*
- *fix-bug*

Pull Request

- Key part of the review process
- Before merging to master we need to open PR
  - usually required 2-3 approvals, by different person
  - assure code quality check
  - only once fully approved, changes might be merged to master

GitHub

- Remote server for storing of Git project and repositories
- Most popular platform nowadays

- However, there are alternatives
  - GitLab
  - BitBucket

source: https://github.blog/wp-content/uploads/2024/07/github-logo.png

Google Colab and Git

The European Commission support for the production of this publication does not constitute an endorsement of the contents which reflects the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.