Learn Version Control with GIT

DATA SCIENCE_BWF_TASK_07

CHAPTER-01: THE BASICS

VERSION CONTROL SYSTEM (VCS)

It is like a "database", which keeps records of the changes we made in our project's files. It allows multiple people to collaborate on a project.

Version control is a system that records changes to a file or set of files over time.

It allows you to revert files back to a previous state, revert the entire project back to a previous state, compare changes over time, see who last modified something that might be causing a problem, who introduced an issue and when, and more.

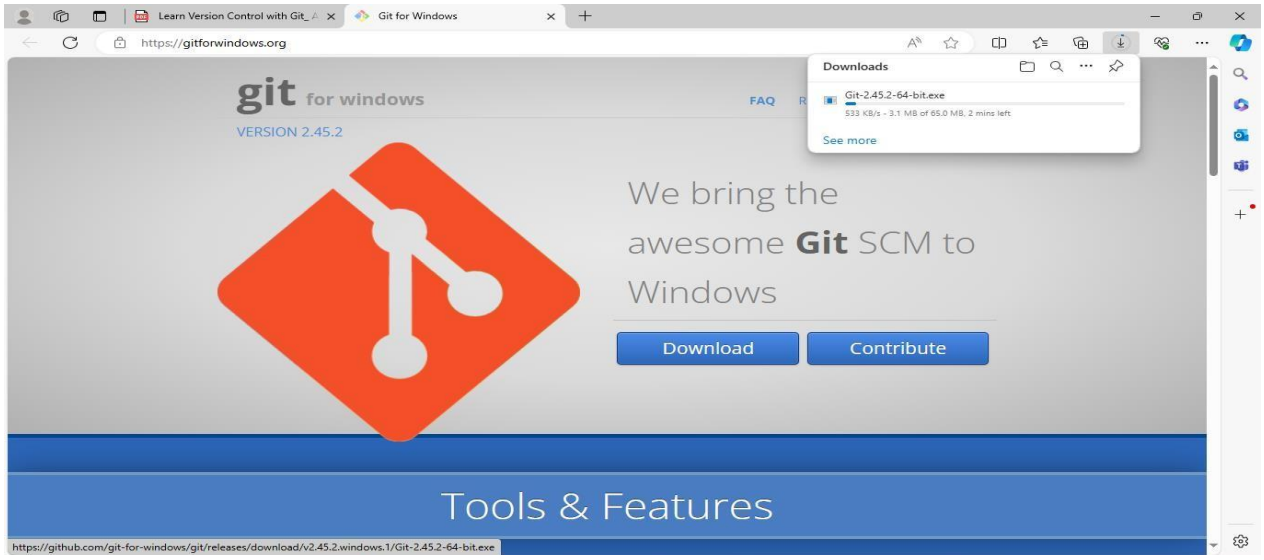
Why Use a Version Control System?

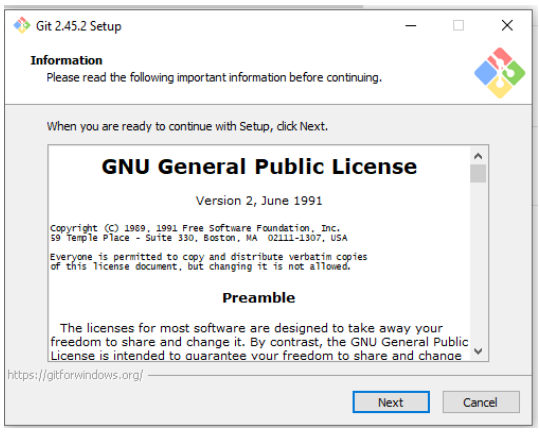
- Allows you to track changes and revert if needed.
- Enables collaboration with others on a project.
- Provides a backup of your files.

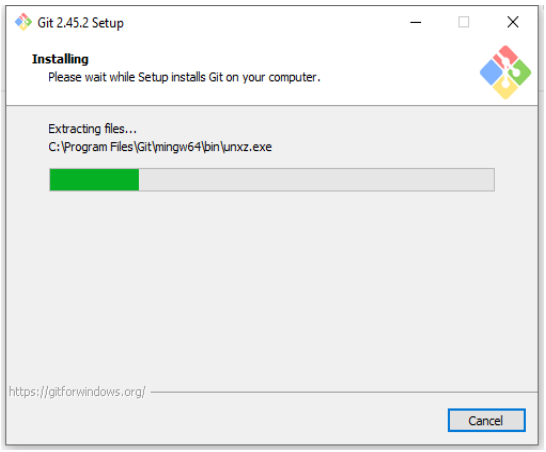
GETTING READY

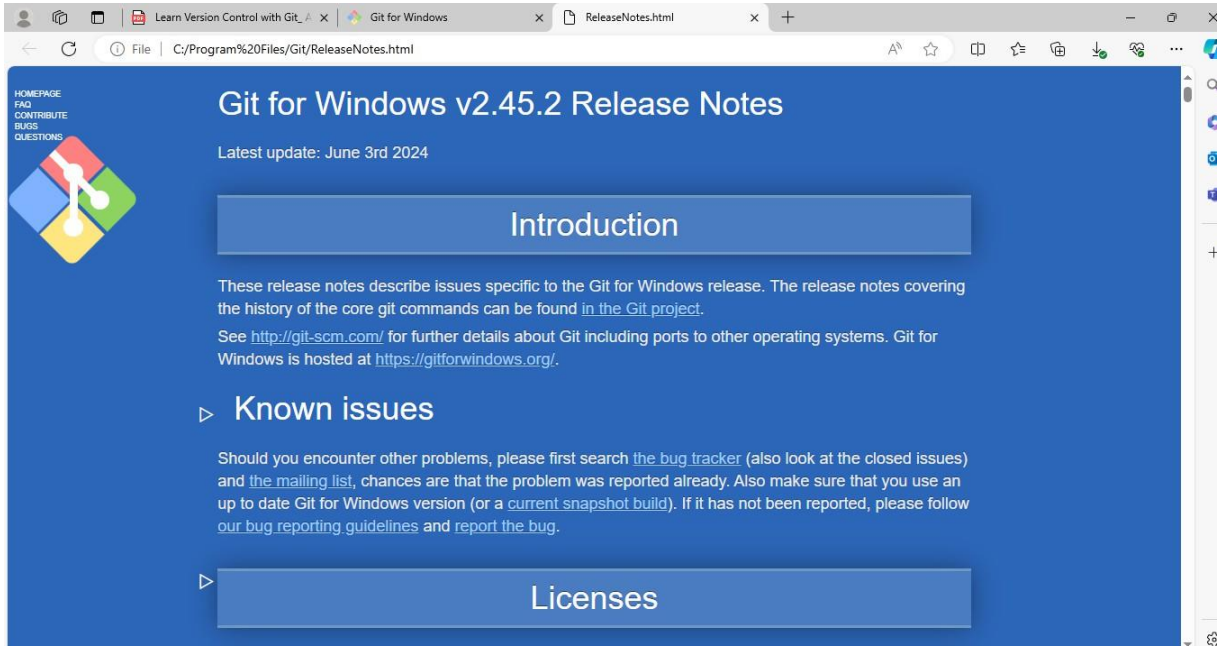
There are two main ways of working with Git: either via its **"Command Line Interface"** or with a **"GUI application".**

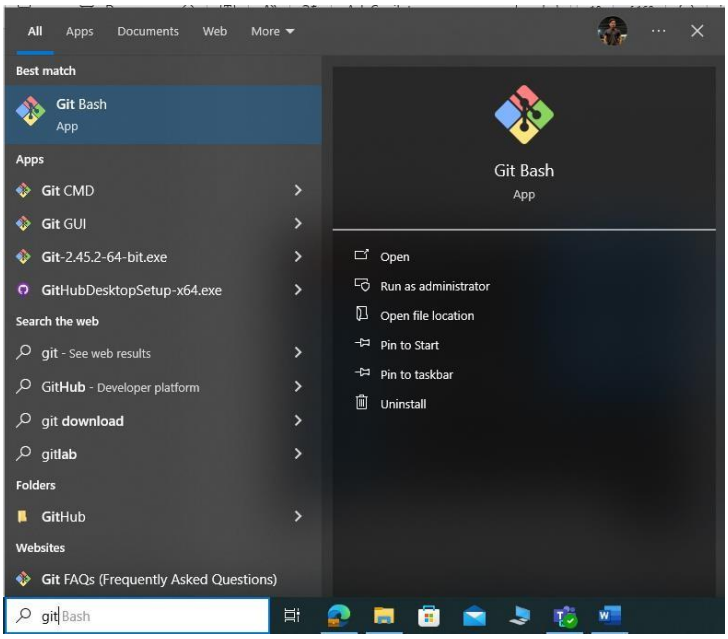
SETTING UP GIT ON YOUR COMPUTER



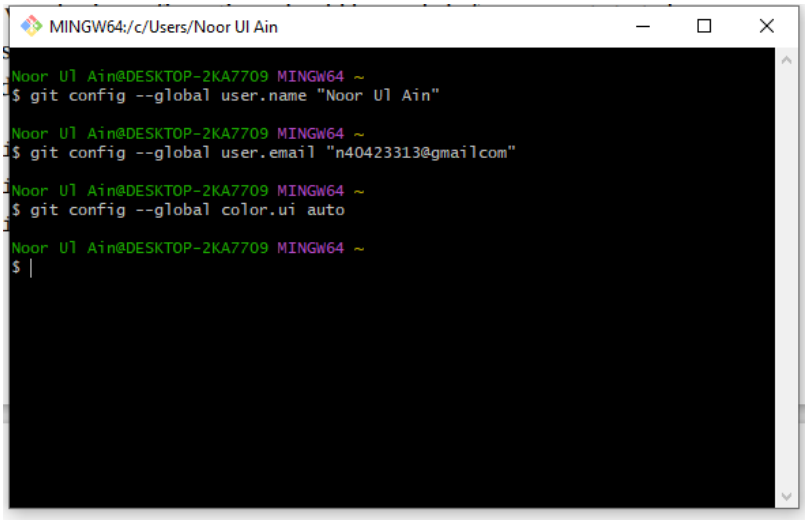

Git 2.45.2 Setup
Information
Please read the following important information before continuing.
When you are ready to continue with Setup, click Next.
GNU General Public License
Version 2, June 1991
Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place - Suite 330, Boston, MA 02111-1307, USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.
Preamble
The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change
https://gitforwindows.org/
Next
Cancel


Git 2.45.2 Setup
Installing
Please wait while Setup installs Git on your computer.
Extracting files...
C:\Program Files\Git\mingw64\bin\unxz.exe
https://gitforwindows.org/
Cancel


Learn Version Control with Git_
Git for Windows
ReleaseNotes.html
File | C:/Program%20Files/Git/ReleaseNotes.html
HOMEPAGE
FAQ
CONTRIBUTE
BUGS
QUESTIONS
Git for Windows v2.45.2 Release Notes
Latest update: June 3rd 2024
Introduction
These release notes describe issues specific to the Git for Windows release. The release notes covering the history of the core git commands can be found in the Git project.
See http://git-scm.com/ for further details about Git including ports to other operating systems. Git for Windows is hosted at https://gitforwindows.org/.
Known issues
Should you encounter other problems, please first search the bug tracker (also look at the closed issues) and the mailing list, chances are that the problem was reported already. Also make sure that you use an up to date Git for Windows version (or a current snapshot build). If it has not been reported, please follow our bug reporting guidelines and report the bug.
Licenses



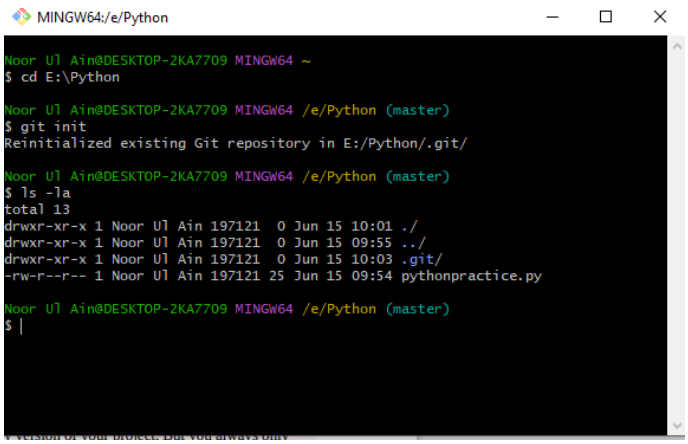
CONFIGURING GIT



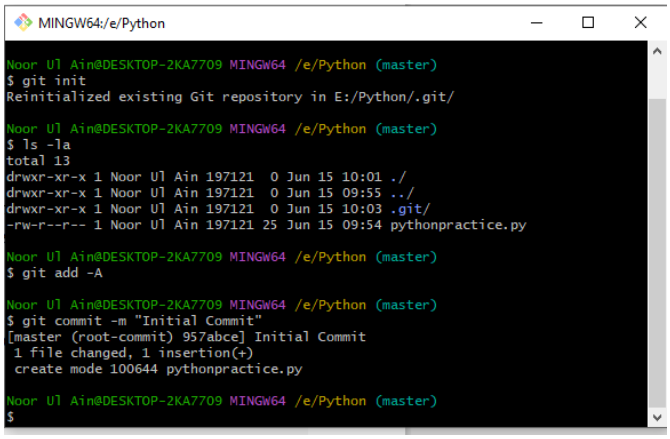
THE BASIC WORKFLOW OF VERSION CONTROL

Working with Repository: In Git, the repository is just a simple hidden folder named ".git" in the root directory of your project. There are two types of repositories

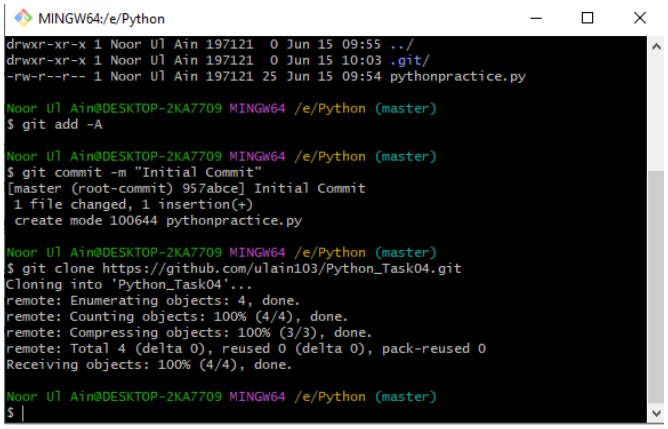
- Local Repository
- Remote Repository



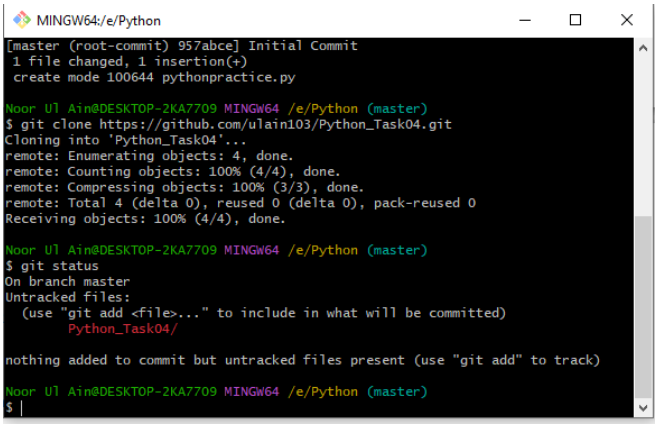
A new, hidden folder was added, named ".git". All that happened is that Git created an empty local repository for us.



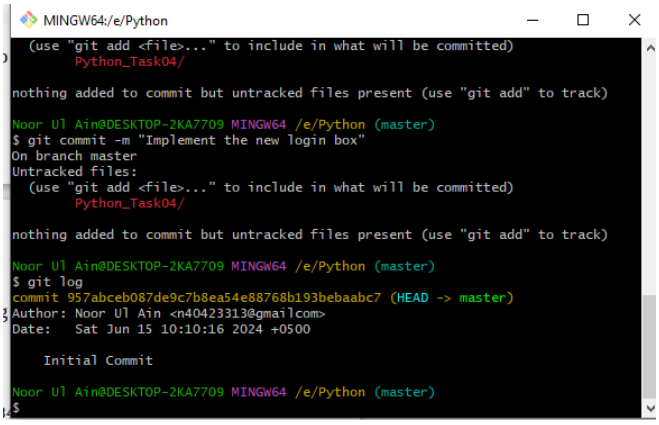
STARTING WITH AN EXISTING PROJECT ON A SERVER

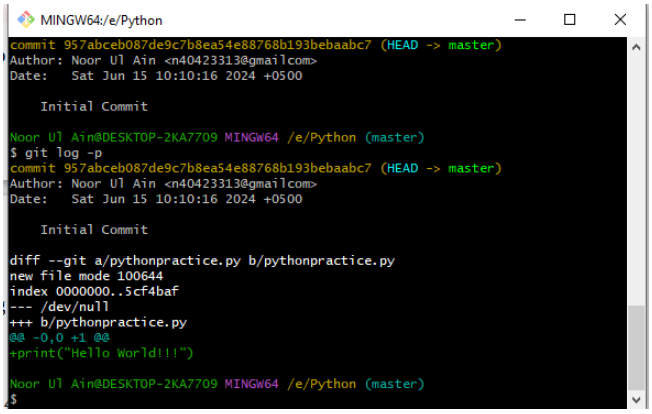


WORKING ON YOUR PROJECT



GETTING READY TO COMMIT

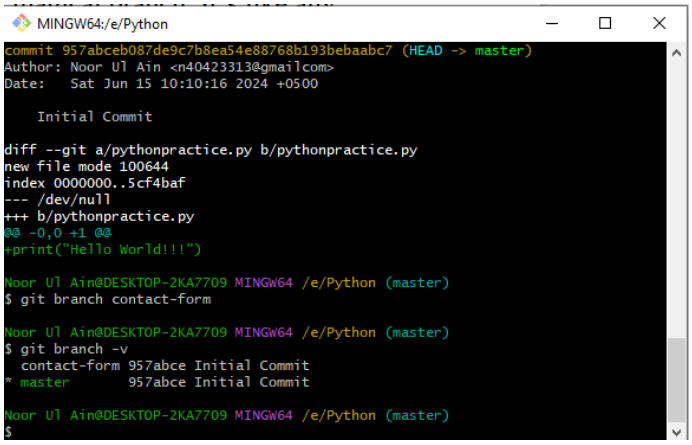




CHAPTER-02: BRANCHING AND MERGING

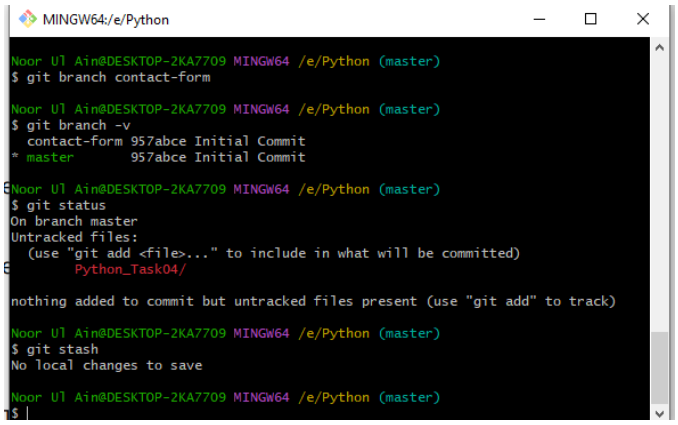
Branches are the perfect tool which helps us avoid mixing up different lines of development. We should use branches extensively in your development workflows: for new features, bug fixes, experiments, ideas.

WORKING WITH BRANCHES

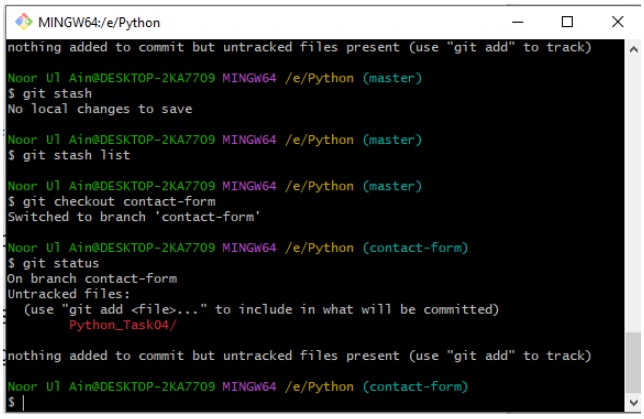


The Stash

Consider it as a clipboard on steroids: it takes all the changes in our working copy and saves them for us on a new clipboard. We ‘re left with a clean working copy, i.e. we have no more local changes.



CHECKING OUT A LOCAL BRANCH

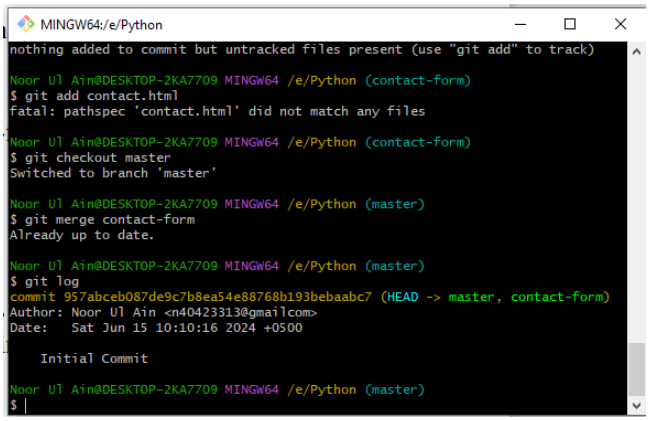


MERGING CHANGES

It requires just two steps:

1. Check out the branch that ha receives the changes.

2. Call the “git merge” command with the name of the branch that contains the desired changes.



EXPLANATION

- Git allows us to display detailed changes in each commit using the `-p` flag with `git log`.
- The `git status` command shows an overview of your changes, grouped into "Changes not staged for commit", "Changes to be committed", and "Untracked files".
- We should write good commit messages that summarize your changes, explain the motivation, and describe how it differs from the previous version.
- A good commit should contain related changes from a single topic, be completed work that has been tested, and have a short descriptive message.
- The `git branch` command lists all branches, with the current HEAD branch indicated by an asterisk.
- We should never commit half-done work, instead use Git's stash feature to temporarily save work in progress.
- Branches have nothing to do with each other by default, but a local branch can be set up to "track" a remote branch using the `--track` flag with `git checkout`.
- The `git push` command uploads new commits from the current HEAD branch to its remote counterpart branch.