****

**Assignment 1**

**Submitted To**

**Faculty Name :** **Puja Chakraborty**

**Submitted By**

**Name : Sheikh Sarafat Hossain**

**Id No : 2022-3-60-109**

**Course Title :** Statistics for Data Science

**Course Code :** CSE303

**Section :** 07

**Semester :** Fall 24

**Date of Submission:** 21 December 2024

1. **Describe what is python script, python module and library?**

**Answer:**

* **Python Script:**

A Python script is a file containing Python code that can be executed directly. It typically has a .py extension. Scripts are usually used to automate tasks, run programs, or execute logic when they are run.

In short python script is a standalone program.

* **Python Module:**A Python module is a file containing Python code (like a script) that can be imported and reused in other Python scripts or programs. It can define functions, classes, and variables.  
  In short python module is Reusable code that can be imported.
* **Python Library:**

A Python library is a collection of modules and packages that provide specific functionality. Libraries like NumPy or Pandas simplify tasks like data analysis or scientific computing.

In short python library is Collection of modules/packages.

1. **Point out the difference and use of .py and .ipynb files.**

**Answer:**

|  |  |  |
| --- | --- | --- |
| **Criteria** | **Python File (.py)** | **Jupyter Notebook File (.ipynb)** |
| **File Format** | Plain text file | JSON-based interactive file |
| **Usage** | For scripting, coding and automation | For interactive coding, data analysis and visualization |
| **Environment** | Runs in python interpreters (Example: VS code) | Runs in Jupyter Notebook environment |
| **Interactivity** | No interactivity, only sequential execution | Interactive cells for code, text, images, etc. |
| **Extension** | .py | .ipynb |
| **Code Organization** | Written in one continuous block of code | Divided into cells (code, text, images, plot) |
| **Best for** | Scripting, production-ready code | Data science, machine learning, education, and analysis |
| **Execution** | Run using python filename.py in terminal | Run in jupyter Notebook (Browser-based) |

1. **What is VsCode and conda?**

**Answer:**

**VS Code:** A lightweight, open-source code editor with support for multiple languages, debugging, and extensions.

**Conda:** A package manager and environment management tool for Python, used to manage libraries, dependencies, and virtual environments.

1. **What is a python environment? How to create multiple python environments? How to activate a particular environment with CLI (Command Line Interface)?**

**Answer:**

A Python environment is an isolated space where specific versions of Python, libraries, and dependencies are installed, ensuring project-specific compatibility and avoiding conflicts between projects.  
After Installed Anaconda we can use VS code terminal to create multiple environments.

**Command to create conda environment:**

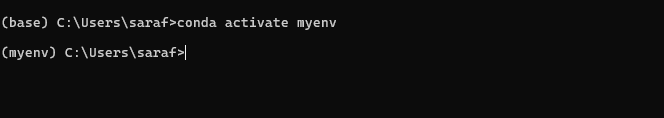
conda create --name env\_name python=3.13.1

A black screen with white dots

Description automatically generated

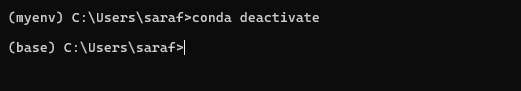
**Command to activate conda environment:**

**conda activate env\_name**



**Command to deactivate conda environment:**

**conda deactivate**



1. **How to integrate conda with VS code?**

**Answer:**

**First Step:**

Download and install Vs code and Anaconda from their website.

VS code link: <https://code.visualstudio.com/download>

Anaconda link: <https://www.anaconda.com/download>

**Second Step:**

Install Necessary extension in VS code such as python extension pack.

**Third Step:**

Set Conda as the Python Interpreter. To do it we need to press (Ctrl + Shift + P) then type “Python: Select Interpreter” then select your conda environment

1. **What is git?**

**Answer:**

Git is a distributed version control system that tracks changes in code, enabling multiple developers to collaborate on projects. It allows you to manage versions, revert changes, and work on different branches simultaneously. Git is widely used in software development to ensure code consistency and collaboration.

1. **Create an account on github.**

**Answer:**

Already have account on github.

My github account: <https://github.com/shiekhsarafathossain>

Useful video link to create a github account: https://youtu.be/Gn3w1UvTx0A

1. **Install necessary extensions on VSCode.**

**Answer:**

Already installed python extension pack and Jupyter Notebook

1. **Define the following terms:**

**a. repo b. stage c. commit d. push e. pull f. merge g. clone h. branch i. fork j. master or main branch**

**Answer:**

1. **repo (Repository):** A repository is a storage space where your project’s files and the version history of your project are stored. It can be located locally (on your computer) or remotely (on a platform like GitHub).
2. **stage:** Staging refers to the process of preparing changes in your working directory to be committed to the repository. In Git, you "stage" files using git add before committing them.
3. **commit:** A commit is a snapshot of your staged changes in the repository. It captures the state of your project at a specific point in time and creates a record of changes, including a message describing the changes made.
4. **push:** Pushing refers to the action of sending your committed changes from your local repository to a remote repository, like GitHub. This updates the remote repository with your latest commits.
5. **pull:** Pulling refers to the action of retrieving updates or changes from a remote repository to your local repository. It incorporates the latest commits from the remote into your local working copy.
6. **merge:** Merging is the process of integrating changes from one branch into another. If two branches have diverged, merging will combine the changes into a single branch.
7. **clone:** Cloning is the process of creating a copy of a repository, including its history, from a remote location (e.g., GitHub) to your local machine. The command git clone is used for this purpose.
8. **branch:** A branch in Git is a separate line of development within a repository. It allows you to work on different features or bug fixes without affecting the main codebase. You can switch between branches as needed.
9. **fork:** Forking is creating a personal copy of someone else's repository. It allows you to make changes in your own copy without affecting the original repository, which is useful for contributing to open-source projects.
10. **master/main:** The master (or main) branch is the default branch in most Git repositories. It represents the main line of development, and most of the time, it holds the production-ready code. "Main" is now preferred over "master" in many projects for inclusivity reasons.
11. **Write the command to execute the mentioned tasks in question 9.**

**Answer:**

1. **Creating a repo:**

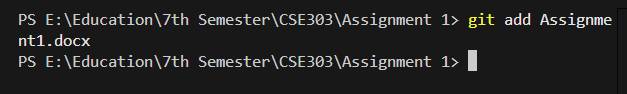
**git init**

A black screen with white text

Description automatically generated

1. **Stage Files:**

**git add file\_name**

A screen shot of a computer

Description automatically generated

In here, we can see that only Assignment1 has been added

**git add . (add all files in that folder)**

**A black background with white text

Description automatically generatedA black background with white text

Description automatically generated**

If we use add . then everything in the folder will be added.

1. **Commit changes:**

**git commit -m “Message”**

A screen shot of a computer screen

Description automatically generated

1. **Push changes into remote repo:**

Before pushing we need to link the github repository with our local repository



**git push origin branch\_name**

A screen shot of a computer

Description automatically generated

1. **Pull changes from remote repo:**

**git pull origin branch\_name**

A screenshot of a computer screen

Description automatically generated

1. **Merge a branch into the current branch**

**git merge branch\_name (This command merges the specified branch into the current branch.)**

A screenshot of a computer program

Description automatically generated

1. **Clone a remote repo:**

**git clone repo\_url**

**A computer screen shot of a black screen

Description automatically generatedA screenshot of a computer program

Description automatically generated**

1. **Create a new branch:**

**git branch branch\_name (Creating a new branch)**

**git checkout (Switching to the new branch)**

A computer screen shot of white text

Description automatically generated

1. **Fork a repo: A screenshot of a computer

   Description automatically generatedA screenshot of a computer

   Description automatically generated**

**A screenshot of a computer

Description automatically generated**

Or we can clone the project as follows:

**git clone fork\_repo\_url**

**A computer screen with text on it

Description automatically generated**

1. **Switch to main/master branch:**

**git checkout main (switch to main branch)**

**git checkout master (switch to master branch)**

A black screen with white text

Description automatically generated

1. **What is the importance of ReadMe.md and requirements.txt file in a project?**

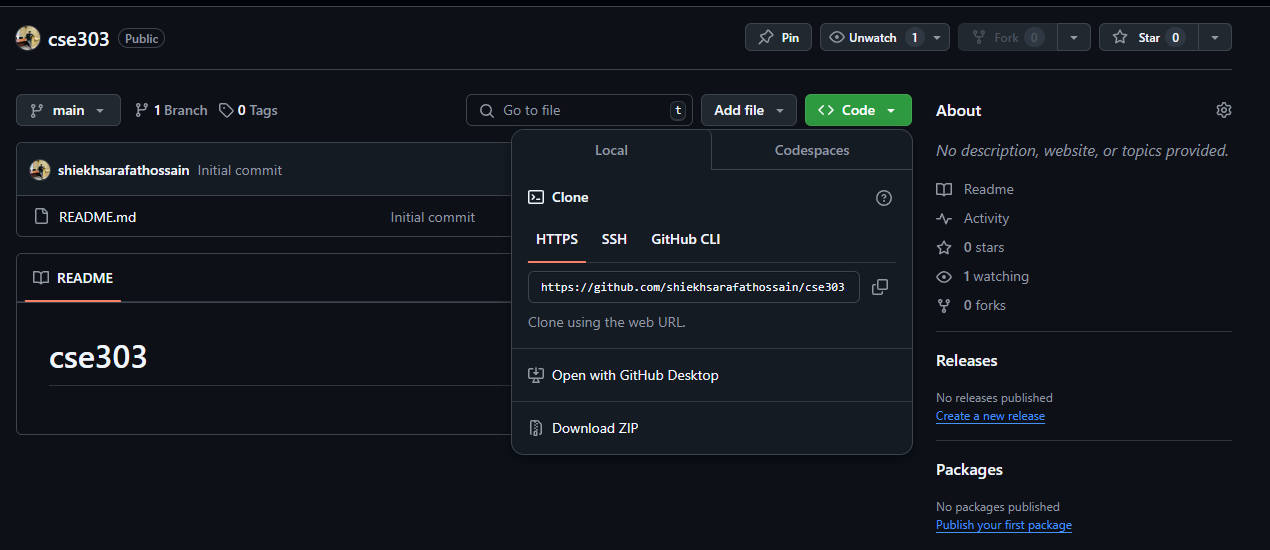
**Answer:**

**README.md**: It provides an overview of the project, including its purpose, installation instructions, usage, and other essential details, helping users and developers understand the project.

**requirements.txt**: It lists all the Python dependencies, and their versions required to run the project, enabling easy environment setup with the command pip install -r requirements.txt.

1. **Create a project on github and clone it to your device.**

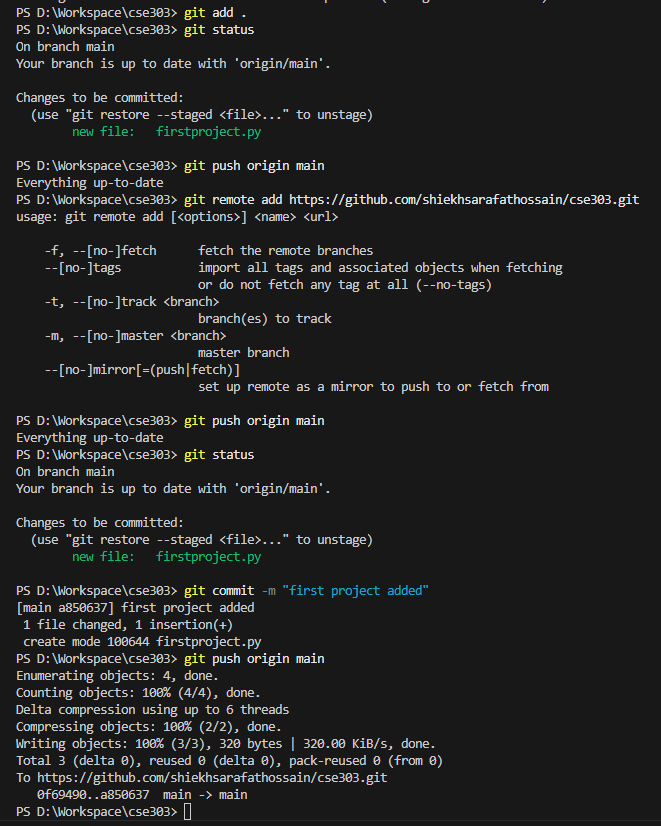
**Answer:**

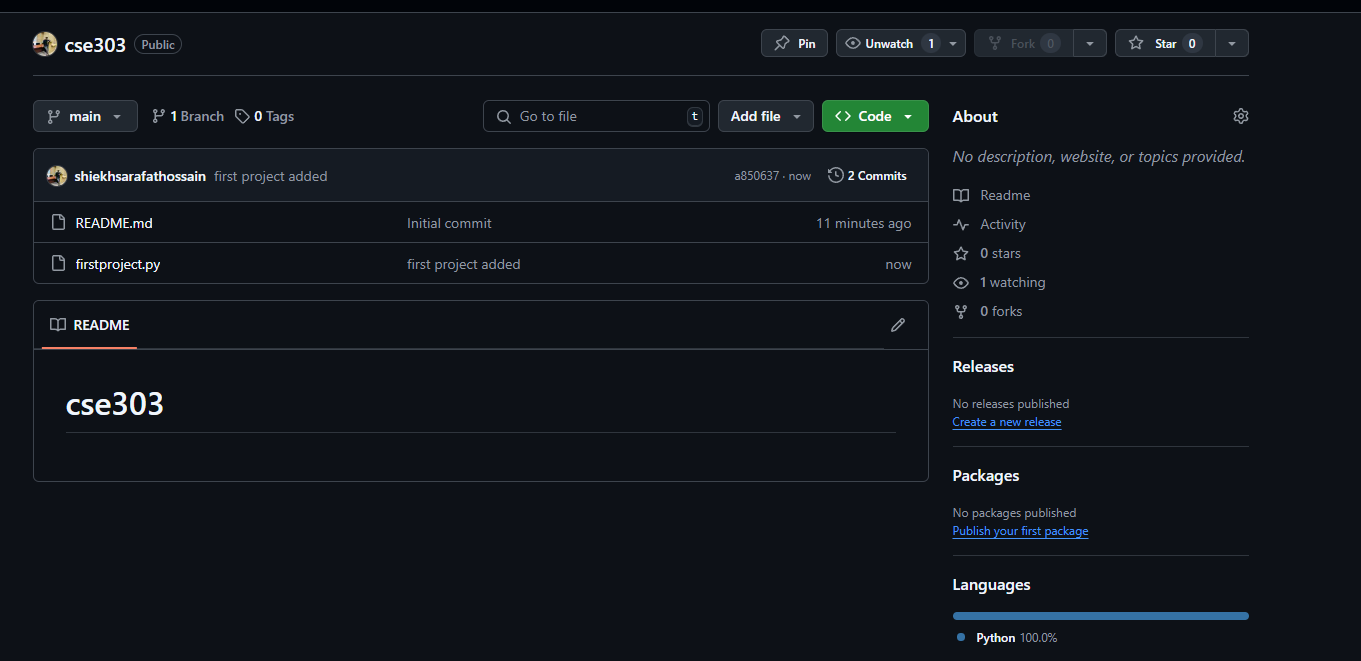
To Create a project, we need to go in github then create new repository after that we need to copy the repository link. 

Then copy the link and paste it in the terminal in the following manners: **git clone repo\_url.**

In my case it is **git clone** [**https://github.com/shiekhsarafathossain/cse303.git**](https://github.com/shiekhsarafathossain/cse303.git)

1. **Modify your cloned project and commit it back to your previously created github account.**

****

****