Python project
Git guidelines

Instructions

1. The project is to be carried out in pairs (only 1 project team of 3 is allowed for odd-numbered groups).
2. You must use **Git and Github** to collaborate on your project and keep track of the different versions of your code.
3. For a quick start with Git, please visit https://douga19.github.io/git-immersion-python/ and **follow the indicated steps**.
4. Each student must create his or her own account on Github.
5. One of the project team members will initiate the project by creating the main repository on his Github account. The other team member will be added as a collaborator on this repository.
6. The repository name must be **"pychatbot-student1Name-student2Name-LabGroup"**. For example, for the pair team of INT3 Durant and Dupont, their main repository will be named **"pychatbot-durant-dupont-int3".**
7. You must have a README.md file (this is a text file) at the root of your project. In this README file, you must enter:
   - o names of project team members,
   - o instructions for use,
   - o any dependencies to be installed,
   - o list of known bugs
8. Once the project repository has been created, **you need to add your two Lab teachers as repository collaborators**. This will give them direct access to your code at any time during the project.
9. Each stage/part of the project is assigned a **release** on the main repository. In other words:
   - o a first **release** will contain Part I (preferably functional) of the project
   - o a second **release** will contain the final version of the project (part I + part II).
10. Make sure you work as efficiently as possible: if tasks can be done in parallel, do them in parallel.

Evaluation criteria

1. Each new feature of your project will be developed on a new branch.
2. The code must be thoroughly tested for each feature to be implemented.
3. **Once each feature has been tested, the developer will push the branch to the main repository and then request a pull request from the main repository.**
4. You can have as many **commits** as you like.
5. The contribution of each project team member should be reflected in the **commits** history.