**Instruction for simple-python-lab**

If you are new to GitHub and working with repositories, follow these detailed steps to get started with the course project repository.

1. **Install Git on your computer**

For Windows user:

Download and install Git from [Git for Windows](https://gitforwindows.org/" \t "_new).

For macOS user:

Install Git using Homebrew by using:

brew install git

For following steps, you need to open your Terminal (macOS/Linux) or Command Prompt/PowerShell (Windows) and use the code provided to get access to the lab.

1. **Configure Git**

Configure Git by using:

git config --global user.name "Your Name"

git config --global user.email "Your Email"

1. **Clone the Repository**

git clone https://github.com/Qiushi-Xia/simple-python-lab.git

This command is for cloning the repository from GitHub. It downloads all the files from the repository into a folder named simple-python-lab on your computer. You can use it to clone all the public GitHub project as above.

1. **Navigate to the Repository Directory**

cd simple-python-lab

This command accesses the directory of the cloned repository.

1. **View the README File**

cat README.md

For more detailed project information and additional instructions, view the README.md file in the directory. This step is crucial, especially when you are working on a new project, as it helps you quickly understand the project's purpose and organization.

1. **Create and Edit Your File**

cd student\_labs

First, access to student\_labs file. Then you can use your favorite text editor or an IDE to create a file. Please name it YourName.py, replacing YourName with your actual name. You can also use this command is to create a file：

touch YourName.py

Then open the file YourName.py in your text editor or IDE. Write or paste your Python code into the file and save your changes.

1. **Upload Your Changes to the Repository**

After editing your file and saving your changes, you need to upload these changes back to the GitHub repository. This involves adding your changes to the staging area, committing them with a message, and pushing them to the repository.

This step prepares your changes for a commit. Ensure you're in the student\_labs directory when you add your file:

git add YourName.py

Commit your changes with a descriptive message:

git commit -m "Your Name-You student ID number"

Push your changes to GitHub:

git push origin main

By following these steps, you will be able to create, edit, and upload your Python files to the student\_labs directory in the repository, which will be graded by an auto-grader.