Final Project

Module – 1 Git & python

- You can choose either github or gitlab
- Trainer handle : lidorg-dev {github} lidorlg1{gitlab}
- Fork the code from : https://github.com/lidorg-dev/final-project.git
- Notice the python code (python ver 3.2+)

- now lock the master branch from commits
- make sure you make one approval in pull request/merge requests

- create a new branch : dev
- create a new branch from dev: <yourname_sol>

- fix the code in yourname_sol branch
- commit your change
- and test the new code in your Linux VM
- then create a pull request to dev branch (attach your test screenshot to the pull request)– add the trainer as approval
- after trainer approved – then merge the Pull Request (PR) to dev branch
  then tag your code to 1.1

Module – 2 Git & JAVA-MAVEN

- First take the code from here https://github.com/lidorg-dev/spring-boot-examples -----from master branch
- The code in Github is written in Java
- Upload or fork it to your own repo in github
- Notice there is In pom.xml there is a version : currently 0.0.1-SNAPSHOT
- now lock the master branch from commits
- make sure you make one approval in pull request/merge requests

- create a new branch : dev
- create a new branch from dev: <yourname_sol>
- fix the code in yourname_sol branch

- now try to build the code in linux VM -- make sure you have maven 3 installed and JDK 1.8 -- (the build command is: mvn compile) – make sure to run that maven command in the pom folder

- if mvn passed then run mvn test command. (it’s failing on the test) – fix it on your branch and increment the pom file to 0.0.2-SNAPSHOT and do PR to trainer for approval with the screenshot of the fix

- after successful merge -> tag the release to 0.0.2