Overview:

In this engendering project you will collaborate with your team to develop an Eclipse app and go on a scavenger hunt.

General Requirements:

1. You have been provided a text file, UCEnglish.txt. It contains an extensive list of words and symbols.
2. You have been provided a location on the UC campus. In the interest of national security and FERPA, the location data is encrypted inside a JSON file, EncryptedGroupHints. Your first job is to write a Python function to extract the location data for your project team and decrypt the location. The format of the encrypted data is simple: each element in the encrypted data is an index into the file English.txt, which has been provided as part of this project. Use these numbers to index into the file and build the decrypted string. For example, [“20842”, “46853”] is *hello world*. ‘hello’ is on line 20842 in the text file. ‘world’ is on line 46853 in the text file.
3. Download the TeamsAndEncryptedMessagesForDistribution file, provided with the assignment, and find your team. Your team has been assigned a popular movie, but the title is encrypted. It’s a grown-up bullet-proof commercial-grade encryption that’s different than the simplistic file mentioned above. Decrypt the movie name and make a sign with a famous quote from your movie. Hold up the sign when you take your group photo, below. LPT: You don't have everything you need to perform this decryption. Your team should ask me for the additional information after you study the problem and figure out what you need.
4. Use this library to decrypt your message from the previous step: cryptography.fernet
5. Gather your team and meet at the location you decrypted. Take a group photo ‘proving’ you were there. Every team member must be recognizable in the photo. Don’t get yourself in trouble. Don’t break any rules. Just have fun with it.
6. Add more logic to your Eclipse project. Write a function that loads and displays the photo you took in the previous step.
7. Add more logic to your Eclipse project. Write an entry point (our main.py) that invokes each of the functions described above, displays the photo, and prints the two decrypted messages. No user input should be required.
8. Plan to demo your work during the final exam period. All group members must be present. Attendance for the entire exam period, from beginning to end, is not optional. The demo will consist of executing both features of the project for the class to enjoy.

Add our standard documentation to your project.

Deliverables:

Submit your submission to Canvas as a URL pointing to the GitHub repo containing your project. Be sure the repo is public. Name the Eclipse project **{*Group Name*}\_FinalProject**. Do not include the {} in the name. Document and architect the project according to our requirements. Your instructor will clone the project onto his laptop for the in-class demo. Every team member should have at least one substantial commit to the repo.

# Grading:

|  |  |
| --- | --- |
| Category | Grade Percentage |
| Presentation by the entire group during the final exam period including a successful execution of the Eclipse project cloned from the GitHub URL submitted to Canvas. | 20 |
| Data imported from the JSON Group Hints file, decrypted, and printed to the console | 20 |
| Movie name decrypted from the encrypted messages file and printed to the console. | 10 |
| Photograph loaded and displayed | 20 |
| Project architected according to our requirements | 10 |
| Project named , submitted, and documented according to our requirements. | 10 |
| Entry point code that invokes the functions detailed in the General Requirements, above. | 10 |

All group members will receive the same grade for their project unless extenuating circumstances warrant.

# Miscellany:

This is group work. Collaboration between groups is prohibited. Adapting algorithms and Python code from third-party sources (other than competing groups) is, in fact, permissible, as long as all sources are properly cited. A proper citation is any format that allows the reader to easily identify, locate, and obtain a specific third-party product, algorithm, or Python code fragment used in the project.

Test your work. I highly recommend you exchange your project file/URL with everyone in your group and verify that your work runs on multiple computers. This ‘proves’ that your program will probably run on my machine. If you don’t know anyone to test your program, or you don’t like anyone you do know, feel free to ask me for help.