

11/15/21

INST326 Project Check-in

Mandy Chan, G Goodwin, Leilani Martinez, Hanna Zakharenko

User class - Leilani and Mandy:

- Mandy
  - Wrote the summary and pre\_visit methods and docstrings, helped Leilani navigate for parts of her code
  - Problems trying to solve - how to log the user's new experiences and summaries into the text files and how to have the user interact with features of the zoo
  - Main things I figured out - How to write the user's goals to the text files and how to keep track of the usernames
- Leilani
  - **Problem:** We wanted to allow the user to create an account for the virtual zoo so that when a returning user visits the zoo again, they can see what they have done at their previous visit as well as the previous quiz questions they've seen. In order to do this we needed to make sure that the user's usernames were unique, which was accomplished.
  - I was able to accomplish the problem by opening the text file of the user's usernames and seeing if the username as a potential new user has entered already exists, if it does exist in the text file, I prompt the user to choose a new, unique username.

- My work was in the Human class and the method called “account” in the User subclass. The account method will need to run with an accompanying file containing usernames that will be titled usernames.txt.

Animal class - Hanna:

- **Problem:** read a file listing the animals in the zoo and all the information about them, and organize it in a dictionary/list that will be able to be accessed in other classes/methods.
- I figured out that the easiest way to organize the information about the animal would be in a list of dictionaries where each dictionary has an animal as a key and a list as the values with each information piece about the animal (e.g. type, fact, what it eats, how long it sleeps, etc.). I used the `__init__` method of this class to accomplish this by opening a sample text file I made in the same format that the final text file will be in.
- My work is in the main text file and needs to be run using the Animal class with the attached file, zoo.txt, as an attribute
  - `Animal(“zoo.txt”)`

Quiz method (Zookeeper class) - G:

- **Problem:** In order to create an interactive quiz for the user, I needed to read a file of quiz questions and correct answers and organize the questions and answers into separate dictionaries. I wanted the quiz questions to be random and not repeat.
- I figured out that tuple unpacking is very useful for separating lines (strings) from a file. After unpacking the relevant information, I changed my original plan of using only sets

to a mix of lists, dictionaries, and one set. These different container data types allow me to store different kinds of information in useful ways. For example, I used dictionaries for the questions and correct answers to be able to number them and keep track of what answer corresponds to what question. I did end up using my initial difference set operator (-) plan to avoid repeat quiz questions.

- My work includes the “quiz\_questions.txt” file and my script in “project\_checkin.py”.
  - As of right now, the method runs by creating an instance of the User class by defining the user.age attribute and then creating an instance of the Zookeeper class with the Zookeeper's age, a user object, and specifying a text file with quiz questions. These instances are written on the global scale for testing purposes at this moment. They will not be run like that in the final submission. An example to test the method is below:

```
■ h = User(19)
■ z = Zookeeper.quiz(9, user = h, file = "quiz_questions.txt")
```