Assignment 10 – OOP (Animal Daycare)

Goals

- Practice writing classes with attributes and methods
- Creating and using instance variables
- More practice reading from CSV files
- Continued practice with functions

Requirements

You will be creating a program that simulates an animal daycare, using a class to represent an animal. Information about the animals in the daycare will be loaded in from a CSV file. The user will be allowed to interact with the animals based on selected menu options. Structure your code using the following requirements:

- Create a new Python file. It must begin with comments in the following format (replace the name and email with your actual information and write text for the description):

```
# Name, USC email
# ITP 115, Spring 2020
# Assignment 10
# Description:
# Describe what this program does.
```

- Animal Class - create a class called **`Animal`** that has the following public attributes and methods:
  - **Attributes**
    - **`hunger`**: an integer representing the animal's hunger meter
    - **`happiness`**: an integer representing the animal's happiness meter
    - **`health`**: an integer representing the animal's health meter
    - **`energy`**: an integer representing the animal's energy meter
    - **`age`**: an integer representing the animal's age
    - **`name`**: a string representing the animal's name
    - **`species`**: a string representing the animal's species
  - **Methods**
    - **Important note**: All of the following methods aside from **`__init__`** and **`__str__`** will modify at least one of the attributes of the animal. The range of the attributes hunger, happiness, health, and energy should be from 0-100, inclusive (age may go beyond 100). Therefore, it is important that before changing any of those values, you check that the change will not make that value outside of the range. If the change causes a value to go below 0, set it 0. If the change causes a value to go above 100, set it to 100 to prevent going beyond the range.

        - For example: If an animal's hunger is currently 5, and you call the **`feed`** method on the animal, its hunger attribute should be set to 0 (and not -5 or any other value).
    - **Constructor Method**
        - **`__init__`**
            - Inputs (7): the constructor should take in 7 values, one for each of the 7 attributes the animal class has (see above)
            - Return value: none
            - Set each of the animal's attributes to the corresponding inputs.
    - **Instance Methods**
        - **`play`**
            - Input: none
            - Return value: none
            - Increase the animal's happiness by 10 and increase the animal's hunger by 5.
        - **`feed`**
            - Input: none
            - Return value: none
            - Decrease the animal's hunger by 10.
        - **`giveMedicine`**
            - Input: none
            - Return value: none
            - Decrease the animal's happiness by 20 and increase the animal's health by 20.
        - **`sleep`**
            - Input: none
            - Return value: none
            - Increase the animal's energy by 20 and increase the animal's age by 1.
        - **`__str__`**
            - Input: none
            - Return value: a string
            - Create a string containing a well formatted message with all of the information about the animal. See sample output for an example of how to format this string. Be sure to include all 7 attributes.
- In addition to the Animal class, define the following functions:
    - **`loadAnimals`**

        - Input: a string representing the name of the csv file
        - Return value: a **`list`** of Animal objects
        - Open the csv file and use each line of data from the file to create new Animal objects. You will store all of the Animals in a **`list`**, which will be returned.
        - Hint: Recall that lines read from a file will be strings, however the some of the Animal class attributes are integers.
    - **`displayMenu`**
        - Input: none
        - Return value: none
        - Print out a menu with all of the possible options to the user. See sample output for an example of how to format your menu.
    - **`selectAnimal`**
        - Input: a list of Animals
        - Return value: the selected animal from the list
        - Print out a menu with each animal's name and species
        - Ask the user to make a selection from the list. Be sure to handle invalid inputs
        - Return the animal from the list that corresponds with the user's selection.
    - **`main`**
        - Input: none
        - Return value: none
        - Begin by creating a **`list`** of **`Animals`** by calling the **`loadAnimals`** function.
        - Using a **`while`** loop, display the menu to the user by calling the **`displayMenu`** function, and then ask the user to make a selection. Be sure to handle invalid menu selections.
        - Based on the user's selection, perform the corresponding actions. For options 1-5, first ask the user to select an animal from the animal list by calling the **`selectAnimal`** function. Call the correct **`Animal`** method on the selected **`Animal`**. After calling the method, print out a message to the user to let the know that method was called on the selected animal.
        - For option 6 (view all animals), print information about all of the animals in the list out to the user.

Extra Credit

- Create an additional function (**not** part of the Animal class) that saves all of the information about the animals back into a **csv** file (meaning the resulting file should

be comma separated value text file, and not a binary data file). The data should be written in the same format as the input file. When the user chooses to exit, save the Animal data to the file.

Sample Output

```
Welcome to the Animal Daycare! Here is what we can do:

1) Play
2) Feed
3) Give Medicine
4) Sleep
5) Print an Animal's stats
6) View All Animals
7) Exit

Please make a selection: 10
*Invalid selection, please try again.

1) Play
2) Feed
3) Give Medicine
4) Sleep
5) Print an Animal's stats
6) View All Animals
7) Exit

Please make a selection: 6
Name: Ollie the Bunny
Health: 85
Happiness: 50
Hunger: 75
Energy: 60
Age: 5
******************************
Name: Murdock the French Bulldog
Health: 80
Happiness: 70
Hunger: 60
Energy: 55
Age: 3
******************************
Name: Socks the Cat
Health: 75
Happiness: 55
Hunger: 20
Energy: 70
Age: 1
******************************
Name: Peewee the Turtle
```

```
Health: 65
Happiness: 45
Hunger: 30
Energy: 15
Age: 10
******************************
Name: Milo the Labrador
Health: 70
Happiness: 75
Hunger: 40
Energy: 80
Age: 6
******************************

1) Play
2) Feed
3) Give Medicine
4) Sleep
5) Print an Animal's stats
6) View All Animals
7) Exit

Please make a selection: 1
1) Ollie the Bunny
2) Murdock the French Bulldog
3) Socks the Cat
4) Peewee the Turtle
5) Milo the Labrador

Please select an animal from the list (enter the 1-5): 0
Invalid input, please try again!

Please select an animal from the list (enter the 1-5): 1
You played with Ollie the Bunny!

1) Play
2) Feed
3) Give Medicine
4) Sleep
5) Print an Animal's stats
6) View All Animals
7) Exit

Please make a selection: 2
1) Ollie the Bunny
2) Murdock the French Bulldog
3) Socks the Cat
4) Peewee the Turtle
5) Milo the Labrador
```

```
Please select an animal from the list (enter the 1-5): 2
You fed Murdock the French Bulldog!

1) Play
2) Feed
3) Give Medicine
4) Sleep
5) Print an Animal's stats
6) View All Animals
7) Exit

Please make a selection: 3
1) Ollie the Bunny
2) Murdock the French Bulldog
3) Socks the Cat
4) Peewee the Turtle
5) Milo the Labrador

Please select an animal from the list (enter the 1-5): 3
You gave Socks the Cat some medicine!

1) Play
2) Feed
3) Give Medicine
4) Sleep
5) Print an Animal's stats
6) View All Animals
7) Exit

Please make a selection: 4
1) Ollie the Bunny
2) Murdock the French Bulldog
3) Socks the Cat
4) Peewee the Turtle
5) Milo the Labrador

Please select an animal from the list (enter the 1-5): 4
Peewee the Turtle took a nap!

1) Play
2) Feed
3) Give Medicine
4) Sleep
5) Print an Animal's stats
6) View All Animals
7) Exit

Please make a selection: 5
1) Ollie the Bunny
2) Murdock the French Bulldog
```

```
3) Socks the Cat
4) Peewee the Turtle
5) Milo the Labrador

Please select an animal from the list (enter the 1-5): 5
Name: Milo the Labrador
Health: 70
Happiness: 75
Hunger: 40
Energy: 80
Age: 6
******************************

1) Play
2) Feed
3) Give Medicine
4) Sleep
5) Print an Animal's stats
6) View All Animals
7) Exit

Please make a selection: 6
Name: Ollie the Bunny
Health: 85
Happiness: 60
Hunger: 80
Energy: 60
Age: 5
******************************
Name: Murdock the French Bulldog
Health: 80
Happiness: 70
Hunger: 50
Energy: 55
Age: 3
******************************
Name: Socks the Cat
Health: 95
Happiness: 35
Hunger: 20
Energy: 70
Age: 1
******************************
Name: Peewee the Turtle
Health: 65
Happiness: 45
Hunger: 30
Energy: 35
Age: 11
******************************
```

```
Name: Milo the Labrador
Health: 70
Happiness: 75
Hunger: 40
Energy: 80
Age: 6
******************************

1) Play
2) Feed
3) Give Medicine
4) Sleep
5) Print an Animal's stats
6) View All Animals
7) Exit

Please make a selection: 3
1) Ollie the Bunny
2) Murdock the French Bulldog
3) Socks the Cat
4) Peewee the Turtle
5) Milo the Labrador

Please select an animal from the list (enter the 1-5): 1
You gave Ollie the Bunny some medicine!

1) Play
2) Feed
3) Give Medicine
4) Sleep
5) Print an Animal's stats
6) View All Animals
7) Exit

Please make a selection: 5
1) Ollie the Bunny
2) Murdock the French Bulldog
3) Socks the Cat
4) Peewee the Turtle
5) Milo the Labrador

Please select an animal from the list (enter the 1-5): 1
Name: Ollie the Bunny
Health: 100
Happiness: 40
Hunger: 80
Energy: 60
Age: 5
******************************
```

```
1) Play
2) Feed
3) Give Medicine
4) Sleep
5) Print an Animal's stats
6) View All Animals
7) Exit

Please make a selection: 7
Goodbye!
```

Deliverables and Submission Instructions

- Create a folder on your computer called **ITP115_A10_*LastName_FirstName*** (replace *LastName* with your last/family name and FirstName with your first name).
- Inside the folder, put your python source code.
- Compress the folder (make a zip file). This cannot be done within PyCharm. Find the folder on your computer and compress it.
  a. Windows*:*
     1. Using File Explorer, select your lab file
     2. Right click
     3. Send to ->
     4. Compressed (zipped) folder
  b. Mac OSX:
     1. Using Finder, select your lab file
     2. Right click
     3. Compress "*FileName*"
- Upload the zip file to your Blackboard section:
  1. On Blackboard, click on the Assignments item in the course menu on the left.
  2. Click on the specific item for this assignment (starts with A and a number).
  3. Click on the Browse My Computer button and select your zip file.
  4. Click the Submit button.

Grading

Item	Points
Animal class	20
loadAnimals method	4
displayMenu method	4
selectAnimal method	4
main method	3
Total*	**35**

** Points will be deducted for poor code style, lack of error checking, improper submission.*