**Zoo**

Objectives:

* Practice utilizing inheritance
* More practice with associations between classes
* Practice overriding methods
* See polymorphism in action

Imagine a game where you can create a zoo and start raising different types of animals. Say that for each zoo you create can have several different animals. Start by creating an Animal class, and then at least 3 specific animal classes that inherit from Animal. (Maybe lions and tigers and bears, oh my!) Each animal should at least have a name, an age, a health level, and happiness level. The Animal class should have a display\_info method that shows the animal's name, health, and happiness. It should also have a feed method that increases health and happiness by 10.

In at least one of the Animal child classes you've created, add at least one unique attribute. Give each animal different default levels of health and happiness. The animals should also respond to the feed method with varying levels of changes to health and happiness.

Once you've tested out your different animals and feel more comfortable with inheritance, create a Zoo class to help manage all your animals.

One way you could put together this zoo is by doing the following:

class Zoo:

*def \_\_init\_\_(self, zoo\_name):*

*self.animals = []*

*self.name = zoo\_name*

*def add\_lion(self, name):*

*self.animals.append( Lion(name) )*

*def add\_tiger(self, name):*

*self.animals.append( Tiger(name) )*

*def print\_all\_info(self):*

*print("-"\*30, self.name, "-"\*30)*

*for animal in self.animals:*

*animal.display\_info()*

*zoo1 = Zoo("John's Zoo")*

*zoo1.add\_lion("Nala")*

*zoo1.add\_lion("Simba")*

*zoo1.add\_tiger("Rajah")*

*zoo1.add\_tiger("Shere Khan")*

*zoo1.print\_all\_info()*

Hopefully this seems a little bit repetitive, and you can use your skills and knowledge to reduce some of the code above, and have fun making a zoo in the process!

This assignment is deliberately open-ended. Have fun!