Description:

The class is animals and stores data on different animals and their characteristics. It uses class and private data variables with get and set methods for them which is a kind of encapsulation.

Variables:

thing: This is a class variable that says what the object we are describing is which is an animal. It is the same for all objects since all objects in this class are meant to be animals.

species: This is a private data variable that stores what kind of animal our object is

weight: This is a private data variable that stores the weight of the animal.

Diet: This is a private data variable that stores the kind of food the animal eats

Methods:

__init__ : This creates a new class object and assigns values to its data variables. The arguments it uses are species, weight, and diet and are assigned to data variables

get_species: This is the getter for the data variable __species and returns the value of the data variable __species. It has no arguments

set_species: This function sets a new species for the animal which updates the data variable species. The argument it uses is the animal's species

get_ weight: This is the getter for the data variable __weight and returns the value of the data variable __weight. It has no arguments

set_ weight: This function sets a new weight for the animal which updates the data variable diet. The argument it uses is the animal's weight.

get_diet: This is the getter for the data variable __diet and returns the value of the data variable __diet. It has no arguments

set_ diet: This function sets a new diet for the animal which updates the data variable diet. The argument it uses is the animal's diet

isscary: This checks if the animal is scary based on its weight. If the animal weighs 400 or more, it prints that the animal is powerful and big. If the animal weighs less than 400 the function prints that the animal is just average. This function does not take any argument and does not return anything.

fight: This compares 2 animal objects and sees which one will win in a fight. It uses another animal as an argument and compares the data variable weight. The animal that weighs more wins. If both animals weigh the same, the animal with the diet of meat wins. If they weigh the same and have the same diet it's a draw.

Description of the demo program:

We first create 3 objects in the animal class each with a different species, weight, and diet. The program then checks the size of each animal with the isscary method to see if each animal is powerful. Then it makes all 3 animals fight each other to see who would beat who. Next, it changes the weight of one animal and makes it fight again to see if the outcome is different.

Instructions:

Download the file and open a terminal window in the same directory as the file. Next type py class.py in the terminal and press enter. If you want to change the data of the animals or make new animals open the file in visual studio code and edit the variable data.