OOP - Lesson 5 - Strengthening Our Programming Skills

More practice with functions and Objects

https://deepnote.com/workspace/Bassie%20Witkin-511f62db-d864-49b6-b1b3-17e7caebc742/project/Course-2022-85b2d840-de20-4fd8-8c72-2d6b125bac85/notebook/4%20-%20OOP%20-%20Practice-5f1a8a1c55e247ce80a9a3311fcdf159

```
class Car:
    #class attributes
    num_wheels = 4
    has_doors = True

    def __init__(self, brand, number_seats, clr, model, gas_tank_size = 14):
        #instance attributes
        self.brand = brand
        self.num_seats = number_seats
        self.color = clr
        self.model = model
        self.gas_tank_size = gas_tank_size
        self.fuel_level = 0  #every single instance of a car will start out at fuel level 0, it
will not remain that way once we start using our cars

    def add_fuel_and_show_message(self, amount, type_fuel = "regular"):
        old_fuel_level = self.fuel_level
        new_fuel_level = self.add_fuel_and_get_fuel_level(amount)
```

Assignment

- Define a class that is a type of Animal (such as Dog or Cat). meaning, your class should not be called Animal, it should be called Dog or Cat or something else creative
- Include 1 class attribute called species
- Include __init__() method
- Include 2 instance attributes
- Include __str__() method
- Include 1 additional method of your choice
- In another file, instantiate your object.
- Show how you use the __str__() method (i.e. print an instance)
- Show how you use the additional method
- Bonus: write another function outside of the class that uses the instance of your object that adds some aspect of functionality
- Include comments! Make your variables and function names self-documenting = name them clearly illustrating what they will do, or what datatype they will hold.