TASK

Capstone Project II

Visit our website

Introduction

WELCOME TO THE SECOND CAPSTONE PROJECT TASK!

Welcome to the Second Capstone Project and congratulations for making it this far!

This task will consolidate the knowledge that you've learnt across various tasks. Now that you are comfortable with Python, we can start working on some complex applications. This Task will focus less on teaching you Python, and more on showing you applications and writing useful programs using text files.

But first, let us understand a very simple concept referred to Inheritance that we'll use in the compulsory task.

Get in touch

Connect for support

Remember that with our courses, you're not alone! You can contact your mentor to get support on any aspect of your course.

The best way to get help is to login to **www.hyperiondev.com/portal** to start a chat with your mentor. You can also schedule a call or get support via email.

Your mentor is happy to offer you support that is tailored to your individual career or education needs. Do not hesitate to ask a question or for additional support!

INHERITANCE

To help us understand inheritance we are going to consider an example using animals. Consider two animals – a lion and a cheetah. Although these are not the same animal, they are both animals so they have certain things in common. Let's say all animals have four legs and a head and teeth. Because all animals have these features, we can define a lion and a cheetah as animals.

Now let's put this in a programming context. If we had to create a class for a lion and one for a cheetah, we would find that we would be duplicating a lot of the same methods and fields in the two classes because the two animals are similar and therefore have features in common. Programmers are lazy! So we always want to try to do things as efficiently as possible which means we minimise the number of duplicates we create. To do this in this example we could make use of inheritance

If we created a class for animals and we defined all the common methods and fields for the two animals in here, we could then derive the lion and cheetah classes from it. We would create another two classes – one for the lion and one for the cheetah. These would both inherit all the methods and fields from the superclass – the animal class. This means the lion and cheetah classes would both have all the properties that the animal class has as well as others which would be specific to that class. Think of the lion and cheetah class as subclasses of the animal class.

INHERITANCE - HANDS ON!

Let's create the scenario above. Create an animal class with variables common to both the lion and the cheetah e.g. number of teeth, spots(Boolean), weight. We will now create the lion class. An example animal class has been included in the task folder in case you get confused.

```
class Animal(object): #Animal is inheriting from object
    '''
    Animal class docstring
    '''
    def __init__(self, numteeth,spots,weight):
        self.numteeth = numteeth
        self.spots = spots
        self.weight = weight
```

It is, however, advised that you create one on your own to practice your skills. The animal class is very similar to previous classes you have created so there should be nothing new there. The lion class will seem quite different from any other class you have created, but do not worry you will understand everything shortly.

Below is the initial setup of the lion class:

```
class Lion(Animal): #Lion is Inheriting from Animal
    '''
    An example of inheritance (see reading).
    This class inherits its constructor and the get_color method from
Animal,
    and every instance of Cow is also an instance of Animal.
    '''
def __init__(self, numteeth,spots,weight):
        super.__init__(self, numteeth,spots,weight)
```

You will notice many unusual things here. The first will be the word 'Animal within the brackets'. This tells Python that you are using inheritance and it also tells Python that the class name which follows this keyword is the class you will be inheriting from – in this case it is the 'Animal' class.

The next thing you will notice is the constructor. If the class you are inheriting from has a constructor which requires parameters, these parameters have to be passed into the subclass and the superclass object must be constructed using these parameters.

You will see that the keyword 'super' is used to denote the superclass. The constructor of the superclass is used here without defining a specific object for it. This is a special case because it is being used in inheritance and therefore it will not cause an error. When accessing methods and variables of the superclass the 'super' keyword will be used along with the dot operator.
You can now add anything to this class that is specific to a 'lion' and it will have those features in addition to all those which are specific to all animals.

Please note that when you use a field or a method of the superclass it will not change for all instances of the superclass. It will only change for that particular instance. Inheritance is not the same as using static variables or methods.

Instructions

Complete the Compulsory Tasks below for completing your second Capstone Project.

Compulsory Task 1

Follow these steps:

- From the Animal class template, we discussed in this task PDF earlier, create a file **Animal.py**.
- Using the Lion class template as shown in the task PDF, expand the class to have features specific to a lion.
- Add a field for lion type.
- Add a method in this class which sets the lion type based on its weight (note that the weight is a derived field from the superclass).
- If their weight is less than 80kg, its type should be a cub. If less than 120 kg, it should be female. And if greater than 120 kg, a male.
- Print out the type of lion.

Compulsory Task 2

Follow these steps:

- Modify the existing **Animal.py** file for this task.
- Create a class called 'Cheetah' that:
    - Inherits from the Animal class.
    - Contains a constructor.
    - Has an array as one of its fields.
- Create a Cheetah object and print it out.

Completed the task(s)?

Ask your mentor to review your work!

Review work

Things to look out for:

1. Make sure that you have installed and setup all programs correctly. You have setup **Dropbox** correctly if you are reading this, but **Python or Notepad++** may not be installed correctly.
2. If you are not using Windows, please ask your mentor for alternative instructions.

Rate us

Share your thoughts

HyperionDev strives to provide internationally-excellent course content that helps you achieve your learning outcomes.

Think that the content of this task, or this course as a whole, can be improved or think we've done a good job?

Click here to share your thoughts anonymously.