Object Oriented Programming, Exercise 6

Topics: Inheritance, UML class diagram

Make a Git commit at least after every coding task.

Code in Python3 and follow the style guide.

1. Draw a UML class diagram of task 2. **Use UML syntax and see how data attributes and methods are presented in the example presented at class (about Automobile), example can be found on lecture slides. You can use any software or e.g. draw by hand and take a picture.** Example in lectures is drawn using MS Visio.
2. Use the Mammal class (that you created in Exercise 4). Inherit **species of animals** from it (e.g. Dogs, Cats, etc.). Species of animals mean that you do not inherit an individual animal, but the whole species (such as Dogs, Cats or Elephants). The individual animals are then the individual objects. Add data attribute for the noise the species of animals make and the diet they have and the animals name and owner. Create a few objects (= individual animals) from the inherited species and display them on screen (= Print out the state of each object (use str-method)).
3. Draw a UML class diagram of exercise 4. See further instructions from task 1.
4. Change your task 2 code like this: Inherit a domestic animal from Mammal. Also inherit a wild animal from Mammal. Then inherit a few species of domestic and wild animals from those classes, create a few objects (= individual animals) and print them out. Each mammal should make a unique noise and have a certain diet as additional data attributes. Add some relevant attributes (such as name, color, owner; name and owner may make sense only for domestic animals). Display your objects on screen.
5. Draw a UML class diagram of tasks 6. See further instructions from task 1.
6. Inherit a student and teacher from a participant of OOP course. Think of a few proper data attributes that are
    a. common for both teachers and students and
    b. different between teachers and students.
7. Each participant of Task 6 has also at 1 domestic animal and 1 wild animal (= let's say, you are the benefactor of some wild animal). Create a few objects (= individual teachers and students) and display them and their information (use str-method). Update the UML diagram.
8. Change task 7 so that each teacher and student can have multiple domestic and wild animals (e.g. use a list). Print out the objects and their information (use str-method). Update the UML diagram.
9. Draw a UML class diagram of some scenario, where you can use inheritance (see task 10 as well).
10. Code your UML diagram. You shall use inheritance, have a few data attributes in both parent and child classes (think, which should be in parent class and which in child class). You shall somehow utilize dictionary in your code.