VSFX 705 / Exercise 5 / Spring 2013 / Fowler*

Exercise 5 – OOP with Inheritance in Python - Interactive Zoo Database

DATE DUE: Class 20
DATE ASSIGNED: Class 18

Goals:

This assignment will focus on the student becoming familiar with python and object oriented programming using inheritance. The emphasis will be on design and implementation of the code.

Requirements:

Design and implement an interactive zoo records database. This will be an object-oriented program that processes a file of zoo animal data and interactively answers queries from a user. Externally, this program's behavior will resemble the previous OOP exercise implementing a student records database. The challenge in this assignment is to implement the database using inheritance. Proper use of inheritance reduces redundancy in code and provides a clean and clear implementation of the relationships underlying data and processing.
.

Considerations:

Specifications:

1. **File Input Specification:**

   The first line of each animal data record in the input file is in the following format:

   Name AnimalType Species Mass

   Explanation:

Name:	The animal's individual name. Contains no white space.
AnimalType:	The type of animal. One of: Mammal, Reptile, Bird
Species:	The animal's species. Contains no white space.
Mass:	The animal's mass (weight) in pounds. A whole number ≥ 1

   Examples:
   Here are examples of first lines of animal data records:

   Bob Mammal Bear 300

   Lucy Reptile Lizard 2

   Oliver Bird Ostrich 75

   The data following the first line in each animal record depends on the AnimalType:

Mammals:

The first line of data for animals of type Mammal is followed by:

LitterSize

Explanation:

LitterSize: The average number of offspring the mammal has. A whole number ≥ 1

Reptiles:

The first line of data for animals of type Reptile is followed by:

VenomousOrNot

Explanation:

VenomousOrNot: Indicates whether the reptile has a venomous bite. One of: Venomous, Nonvenomous

Birds:

The first line of data for animals of type Bird is followed by:

Wingspan TalksOrMute

Explanation:

Wingspan: The wingspan of the bird in inches. A whole number ≥ 1
TalksOrMute: Indicates whether the bird talks. One of: Talks, Mute

Finally, birds that talk have an additional line of data:

Phrase

Explanation:

Phrase: What the bird says. May contain whitespace, but is no more than one line long.

2. Example Data File:

Here is an example of a data file to be read by the program:

```
Bob Mammal Bear 300
2
Lucy Reptile Lizard 2
Nonvenomous
Carl Reptile Cottonmouth 3
Venomous
```

Oliver Bird Ostrich 75
60 Mute
Polly Bird Parrot 1
2 Talks
I want a cracker
Doug Mammal Dog 20
4

For this exercise, assume that each animal has a unique name.

3. **Program Output and Behavior Specification:**

Once the program has read in the data file, it should request and process interactive queries from the user. It should request queries in the following format:

Zoo data read in.
Query animal species[s], mass[m], litter[l], venom[v], wingspan[w], talk[t] or exit session[e]?

In a similar manner to the previous exercise, the user interactively queries the database for information on individual animals. The following example demonstrates all varieties of queries and responses, using the data provided in the above example file.

Zoo data read in.
Query animal species[s], mass[m], litter[l], venom[v], wingspan[w], talk[t] or exit session[e]? s
Animal Name? Bob
Bob species is Bear
Query animal species[s], mass[m], litter[l], venom[v], wingspan[w], talk[t] or exit session[e]? m
Animal Name? Lucy
Lucy mass is 2
Query animal species[s], mass[m], litter[l], venom[v], wingspan[w], talk[t] or exit session[e]? l
Animal Name? Bob
Bob litter size is 2
Query animal species[s], mass[m], litter[l], venom[v], wingspan[w], talk[t] or exit session[e]? v
Animal Name? Carl
Carl is Venomous
Query animal species[s], mass[m], litter[l], venom[v], wingspan[w], talk[t] or exit session[e]? v
Animal Name? Lucy
Lucy is Nonvenomous
Query animal species[s], mass[m], litter[l], venom[v], wingspan[w], talk[t] or exit session[e]? w
Animal Name? Oliver
Oliver Wing Span is 60

Query animal species[s], mass[m], litter[l], venom[v], wingspan[w], talk[t] or exit session[e]? t
Animal Name? Oliver
Oliver is Mute
Query animal species[s], mass[m], litter[l], venom[v], wingspan[w], talk[t] or exit session[e]? t
Animal Name? Polly
Polly talks.
Polly says I want a cracker
Query animal species[s], mass[m], litter[l], venom[v], wingspan[w], talk[t] or exit session[e]? e
Goodbye!

4. **Program Structure:**

   The program must use an object oriented approach with inheritance. Specifically, classes representing animals, mammals, reptiles, birds, and talking birds should be employed to store each animal's data, implementing appropriate inheritance, and providing methods to return and/or output query responses.

5. **Error Handling:**

   For this exercise, you can assume that the data input file is properly formatted. In addition, you may assume that users will only query litter size for mammals, venom status for reptiles, and wingspan and talkativeness for birds. However, the program should gracefully handle cases in which a user does not respond with one of s, m, l, v, w, t, or e to the program's query request, or provides the name of a animal not in the database.

Submissions guidelines:
Create a directory named **LastnameFirstnameExercise5**

- you should be prepared to have your program run on test data
- LastnameFirstnameExercise5.py files and any additional files named appropriately

Grading:
Functional, clean, efficient code will be the emphasis.
Meeting the minimum specifications, 80%. To move your grade above 80% go beyond the specifications, demonstrate exploration and understanding.