Exercise: Zoo Simulation

Classes and Interfaces:

Animal (Base Class):

Attributes: name, age, species

Methods: makeSound()

Mammal (Subclass of Animal):

Additional Attributes: furColor

Additional Methods: giveBirth()

Bird (Subclass of Animal):

Additional Attributes: wingSpan

Additional Methods: fly()

Aquatic (Interface):

Methods: swim()

Reptile (Interface):

Methods: crawl()

Instructions:

Create a base class Animal with attributes name, age, and species. Implement a method makeSound() that prints a generic sound.

Create a subclass Mammal that inherits from Animal. Add an attribute furColor and implement a method giveBirth() that prints a message indicating the mammal is giving birth.

Create a subclass Bird that inherits from Animal. Add an attribute wingSpan and implement a method fly() that prints a message indicating the bird is flying.

Create an interface Aquatic with a method swim().

Create an interface Reptile with a method crawl().

Create a class Dolphin that implements both the Mammal subclass and the Aquatic interface. The Dolphin class should have additional attributes specific to dolphins, and it should implement both giveBirth() and swim().

Create a class Snake that implements both the Reptile interface and the Animal class. The Snake class should have additional attributes specific to snakes, and it should implement both crawl() and makeSound().

Create instances of the Dolphin and Snake classes, and demonstrate their attributes and behaviors

Then

Additional Features:

Animal Enclosure:

Create a class Enclosure that can hold multiple animals. It should have a method to add animals to the enclosure and a method to display details of all animals in the enclosure.

Flyable Interface:

Create an interface Flyable with a method takeOff() and land(). Modify the Bird class to implement the Flyable interface. Update the demonstration to include flying actions.

Amphibian Class: Create a class Frog that inherits from Animal and implements the Aquatic and Reptile interfaces. Frogs can both swim and crawl.

Visitor Interaction: Create a class Visitor with a method observe(animal) that prints a message indicating the visitor is observing a specific animal. Demonstrate visitor interaction with both dolphins and snakes.

Challenges:

Age Limit for Dolphins:

Add a rule that dolphins in the zoo cannot be older than 10 years. Implement a check in the Dolphin class constructor to enforce this rule. If the age is invalid, raise an exception.

Custom Sound for Snakes:

Allow the Snake class to override the makeSound() method to produce a unique hissing sound. Update the demonstration to showcase the custom sound.

Random Animal Generator:

Create a function that generates a random animal (either a dolphin or a snake) with random attributes. Add the generated animal to an enclosure and demonstrate its details and behaviors.

Final part:

Create a class zoo that has various animals, and various type of enclosures with those animals. Save your zoo to a file. Reload the file and restore your zoo.