Department of Computer Science, Mathematics & Physics

COMP2232 – Object-oriented Programming Concepts

Semester 2, 2021-22: Assignment #1

Zoo Keeper

Read this entire document before starting your assignment!

This assignment is to be completed on an INDIVIDUAL basis. All work is to have been created by the student submitting the work. Any form of copying, cheating or plagiarism will result in the assessment being disqualified and awarded a grade of 0 (zero).

The purpose of this assignment is to provide exposure to classes, inheritance and data structures (array lists) and increase familiarization with their usage in Java.

For this assignment, you will be creating a Zoo Keeper program. The user of this program will carry out some zookeeping duties such as feeding animals and administering medicine to those that are sick.

The program will be command line based.

Minimum requirements are as follows.

1 - Animal Class (Animal.java):

- This stores information about an animal.
- **Attributes:** Must have attributes to store the animal's species (e.g. "Tiger"), name (e.g. "Timmy"), age (e.g. 3), hunger status (range 0-5; 0 being starving and 5 being full) and health status (range 0-10; 0 being deceased and 10 being perfect health).
- **Behaviours:**
  - **constructor:** initializes species, name, age to Java type default values. The values of hungerStatus & healthStatus should be set randomly.
  - Mutators and accessors for each data member
  - Eat Food: accepts an integer amount of food and feeds the animal, updating its hunger status appropriately.
  - Take Medicine: accepts an integer amount of medicine and heals the animal, updating its health status appropriately.
  - Speak: displays "make noise"

2 – Subclasses of Animal

- Tiger Class (Tiger.java): overrides speak method with an appropriate message.
- Lion Class (Lion.java): overrides speak method with an appropriate message.
- Peacock Class (Peacock.java): overrides speak method with an appropriate message.
- Horse Class (Horse.java): overrides speak method with an appropriate message.
- Elephant Class (Elephant.java): overrides speak method with an appropriate message.

3 - Zoo Class (Zoo.java):

- This stores and displays Animal objects
- **Attributes:** cages (an ArrayList) used store Animal objects.
- **Behaviours:**
  - Constructor: used to instantiate the animal list.
  - Add an Animal: this method accepts an Animal object and appends it to the list of animals
  - Show Animals: this method will display details about each of the Animal objects within the animal list. If no there are no animals in the list, it should display an appropriate message indicating this.
  - Get Animal: accepts an integer parameter value, using it to locate a specific animal (object) in from the cages list; the object is returned

4 – ZooKeeper Class (ZooKeeper.java):

- This class utilizes the Zoo and Animal classes to perform its duties.
- **Attributes:** name of the keeper
- **Behaviours**
  - Feed Animals: used to feed each animal in the zoo. The user must specify the correct amount of food for each animal, ensuring that they are fully fed. Animals die from overfeeding or starvation. Error checking is expected.
  - Heal Animals: used to heal each sick animal in the zoo. Sick animals are those with a health status less than 8. The user must specify the correct amount of medicine for each sick animal, ensuring that they are brought back to perfect health. Animals die from overmedication/overdosing. Error checking is expected.

R5 - ZooManager Class (ZooManager.java)

- This is the driver class containing your **main** function. At a minimum, it will:
- Welcome the user to the system and get their zoo keeper name.
- Create the appropriate zoo keeper and zoo object.
- Allow the user to carry out tasks:
  - add animals to the zoo
  - view all of the animals in the zoo
  - feed animals
  - heal animals
- A program loop must allow the user to perform tasks repeatedly until they indicate they wish to terminate the program. This will include a menu to allow task selection.
- Exit with an appropriate message

You are required to use each of the fields and implement each of the methods, as given. You may create additional fields and methods, if necessary, to complete your program.

Marks

This assignment is worth **18%** of your final course mark. In addition to marks for code, overall marks will also be allocated for: Documentation/code formatting; User-friendly interface; Successful compilation; Correct execution.

There must be appropriate use of parameter passing and returning values from methods.

Marks will be deducted for:

- Violations of the Code Conventions. *Please ask if not sure.*
- Missing/inappropriate use of access specifiers
- Violations of **Assignment Rules**.

Deliverables

1. Assignment 1 is due for submission on **Friday 18th February 2022 by 11:55pm** via the Moodle/eLearning submission tool.
2. Only **ZIP** files will be accepted. No other compression types should be used. Failure to comply with this instruction will incur a penalty.
3. Ensure that you follow the **Assignment Rules** (see course page).
4. You must submit a **Plagiarism Declaration Form** with this assignment, if you do not accept the online one. Your submission will not be marked until your declaration is received.

PLEASE NOTE: The specifications for this assignment are subject to change. You will be notified if any such changes were to occur.