OBJECT ORIENTED PRINCIPLES.

Lab04: OOP Fundamentals: Inheritance.

On this lab a Java application for practising Inheritance is proposed.

- The folder **src** contains the code of the Java application.
  - o MyMain.java => This class tests the functionality of Lab04. <u>This class is provided completed, so you must not modify it</u>.
  - o Animal.java => Our parent class in the class hierarchy, modelling the concept of an animal. <u>You must complete this class</u>.
  - o Dog.java => Our first child class in the class hierarchy, modelling the concept of a dog. <u>You must complete this class</u>.
  - o Cat.java => Our second child class in the class hierarchy, modelling the concept of a cat. <u>You must complete this class</u>.
  - o GoldenRetriever.java => Our grandchild class in the class hierarchy, modelling the concept of, specifically, a Golden Retriever dog. <u>You must complete this class</u>.
  - o DogToy.java => Our class example for a has-a relationship. This class is not part of the class hierarchy. However, all dogs (that is, Dog.java and GoldenRetriever.java have a field representing the dog toy). <u>This class is provided completed, so you must not modify it</u>.

- The folder **javadoc** contains the documentation of the Java application.
  - o MyMain.html => This file contains the description of the class MyMain.java. As the class is provided completed use this file just as a reference.
  - o Animal.html => This file contains the description of the fields, constructor, get and set methods of the class Animal.java. Use this file to guide you in completing the code of the file Animal.java.
  - o Dog.html => This file contains the description of the fields, constructor, get methods, set methods and extra functionality of the class Dog.java. Use this file to guide you in completing the code of the file Dog.java.
  - o Cat.html => This file contains the description of the fields, constructor, get methods, set methods and extra functionality of the class Cat.java. Use this file to guide you in completing the code of the file Cat.java.
  - o GoldenRetriever.html => This file contains the description of the fields, constructor, get methods, set methods and extra functionality of the class GoldenRetriever.java. Use this file to guide you in completing the code of the file GoldenRetriever.java.

o DogToy.html => This file contains the description of the fields, constructor, get and set methods of the class DogToy.java. As the class is provided completed use this file just as a reference.

- The folder **results** contains the expected results of the Java application.
  o result_option_1.txt => This file contains the output obtained from running the lecturer's solution when using <option = 1> in the main method of MyMain.java. That is, when the Java application is used to test the functionality of the class Animal.
  o result_option_2.txt => This file contains the output obtained from running the lecturer's solution when using <option = 2> in the main method of MyMain.java. That is, when the Java application is used to test the functionality of the class Dog.
  o result_option_3.txt => This file contains the output obtained from running the lecturer's solution when using <option = 3> in the main method of MyMain.java. That is, when the Java application is used to test the functionality of the class Cat.
  o result_option_4.txt => This file contains the output obtained from running the lecturer's solution when using <option = 4> in the main method of MyMain.java. That is, when the Java application is used to test the functionality of the class GoldenRetriever.

Actions:

1. Complete the classes Animal.java, Dog.java, Cat.java and GoldenRetriever.java.
   The fields, constructor, get methods, set methods and extra functionality must follow the description given by the Javadoc files Animal.html, Dog.html, Cat.html and GoldenRetriever.html, respectively.