

NAME : KERRY WONG SZE KAY (272135)

QUIZ 3

- What will be printed?

```
----jGRASP exec: java Lab1Stage2
Meow. I am a cat. My name is Kurre
and I am 6 years old.
Woof. I am a Dog. My name is Vilma
and I am 3 years old.

----jGRASP: operation complete.
```

- Explain how it is that the instance variable age of Animal can be used in Cat and Dog when it is declared in Animal.

The class Cat and Dog are subclasses of Animal class, they obtain all of the parent classes public and protected properties and methods through inheritance. An instance variable in a subclass which is Cat class and Dog class can shadow an instance variable of the same name in its parent class which is Animal class.

- What actually consists an instance of?

Animal object kurre contains an instance of Cat while vilma contains an instance of Dog

- And what consists a class of?

Java classes consist of variables and methods (also known as instance members).

The java files which are Animal.java, Cat.java, Dog.java and even Lab1Stage2.java consists of a classes with the same name as the file respectively. A class consists of properties and method definitions.

- What is the difference between a class and an instance?

A class is a blueprint which you use to create objects. An object is an instance of a class and it is a concrete 'thing' that you made using a specific class. So, 'object' and 'instance' are the same thing, but the word 'instance' indicates the relationship of an object to its class.

- Change the declaration of the instance variable age of Animal to a class variable using

static, in this way:

public static int age;

```
public class Animal{
    private String name;
    public static int age;

    public void introduceYourself() {
        System.out.println("Morr. I am an animal.");
        System.out.println("and I am " + this.age + " years old.");
    }
}
```

- What is the result of the output now? Why?

It may happen that you get warnings from the compiler that you should access the variable age via Animal.age, but it should be possible to run the program anyway. Otherwise, change the references to age into Animal.age.

```
----jGRASP exec: java Lab1Stage2
Meow. I am a cat. My name is Kurre
and I am 3 years old.
Woof. I am a Dog. My name is Vilma
and I am 3 years old.

----jGRASP: operation complete.
```

Fields that have the static modifier in their declaration are called static fields or class variables. The static fields are special properties that can be accessed without instantiating a class. However, only a single instance of it exists and shared between all objects of the same class. Therefore, when we set the age of Vilma to 3, age for Kurre will also be set to 3 since the age is now a static field that belongs to Animal. Therefore, when set the age of Vilma to 3, Kurre's age will also be set to 3 since the age is now a static field that belongs to Animal.

- **Where is the value of an instance variable stored?**

The instance variables are stored under the class it was defined in. For example, the kurre and vilma instance variables were defined in the `main` method of class `Lab1Stage2`. Therefore, the variables belong to the scope of `main` method inside `Lab1Stage2` class.

- **Where is the value of a class variable stored?**

The value of class variable was stored inside a property of the class' objects. For example, `age` is a property of `Animal`. Therefore, the class variable `age` will be stored in an `Animal` instance.

- **What refers the variable `this` to?**

The keyword "`this`" refers to the class itself. If two variables or methods with the same name are used, "`this`" keyword will make sure the current class version of the two will be used instead.