Gremaud Lucien
Paunoiu Bruno

Serie 4

IN.2028
March 19, 2019

1 Theory

(a) An abstract class is a class that cannot be instantiate. It only serves as a building block to extend from. It is mainly used when multiple classes share multiple same aspect but it does not make sense to be able to instantiate the general case.
For example, if we want to create different fruit classes, like Apple, Banana, Peach, ect..., it would make sense to create a parent class called Fruit that gather every shared properties of the fruits. It would not make sense to be able to instantiate an object that is simply a fruit, since a fruit is an abstract concept that as no tangential form or meaning. It is therefore advised to make Fruit and abstract class.

(b) An abstract method is an empty method that is meant to be extended. Those methods are used in abstract classes as building block for child classes.
If a method in the abstract class need to call another that is not the same for every child class, that is where we need to use an abstract method. An abstract method can be reference by another method of the abstract class, this abstract will then react depending on which child class it is called from.
For example, if we have an abstract class called Vehicle and inside it there is two method one called distance that returns the distance of a vehicle from its speed and time passed, and an abstract method called speed that gives the speed of the vehicle. This method is abstract because not every vehicle changes speed in the same way. The method distance will reference the method speed which will be determine depending on which vehicle it is called from.

(c) An abstract method must be declared in an abstract class.

(d) An interface is a tool use to define the way classes manage their content regarding other classes. It is essential to organize the interaction between classes of a program.
Since Java does not support multiple inheritance, we use interfaces to counter this issue.

(e) **extend**: Allow a class to extend another and use its variable and method and override them at its convenience. It makes the class using it a child of the class extended. **implement**: Allow a class to implement an interface and use its methods.

(f) A class can implements multiple interfaces.

(g) A class can only extends one class.

(h) It is not possible to instantiate an abstract class or an interface.

(i) Every static fields are by default public. Every method are by default public. We cannot modify the visibility of an interface.

(j) Every field and method of an abstract class is by default public. Their visibility can be modified by using any desired modifier.

(k) Fields of interface have to be static, you can't modify it.

(l) Fields of abstract method have no default property.

3 Debugging

(a)

```
0
8
0
5
9
```

(c)

```
13
14
15
18
13
```