

# Exercises 20

## constructors

### 1.20 Constructors

#### Question 20.1.1

Here is a program that illustrates the behavior of constructors with the calls for constructors of the superclass (`super()`). This program focuses on constructors, there is nothing else in the class.

```
class Premiere{
    Premiere(){ Terminal.ecrireStringln("constructeur de Premiere");}
}
class Seconde extends Premiere{
    Seconde(boolean b){ super();
        Terminal.ecrireStringln("constructeur de Seconde");}
}
class Troisieme extends Premiere{
    Troisieme(int i){super();
        Terminal.ecrireStringln("constructeur de Troisieme");}
}
class Quatrieme extends Troisieme{
    Quatrieme(double d){super(14);
        Terminal.ecrireStringln("constructeur de Quatrieme");}
}
class Exo20_1_1{
    public static void main(String[] args){
        new Premiere();
        Terminal.ecrireStringln("=====");
        new Seconde(true);
        Terminal.ecrireStringln("=====");
        new Troisieme(15);
        Terminal.ecrireStringln("=====");
        new Quatrieme(12.3);
    }
}
```

1. **draw** the inheritance graph of the program, tracing the inheritance relationships between classes without forgetting `Object`, the superclass `Premiere`.
2. **predict** what this program displays.
3. **compile** and run this program to test your predictions.
4. Is there a constructor that runs without displaying a message during the creation of certain objects?

### Question 20.1.2

```
class Cinquieme extends Premiere{
    Cinquieme(){
        Terminal.ecrireStringln("constructeur de Cinquieme");
    }
}
class Sixieme extends Cinquieme{
    Sixieme(){
        Terminal.ecrireStringln("constructeur de Sixieme");
    }
}
class Exo20_1_2{
    public static void main(String[] args){
        new Cinquieme();
        Terminal.ecrireStringln("=====");
        new Sixieme();
    }
}
```

1. **predict** what this program displays.
2. **compile** and run this program to test your predictions.

### Question 20.1.3

The following program causes a compilation error: Class Huitieme is incorrect.

```
class Septieme extends Premiere{
    Septieme(int i){
        Terminal.ecrireStringln("constructeur de Septieme");
    }
}
class Huitieme extends Septieme{
    Huitieme(){
        Terminal.ecrireStringln("constructeur de Huitieme");
    }
}
```

Try to find the error. If you do not succeed, try to compile this program: the message error can put you on track.

### Question 20.1.4

```
class Neuvieme extends Premiere{
    Neuvieme(int i){
        Terminal.ecrireStringln("premier constructeur de Neuvieme");
    }
    Neuvieme(boolean b){
        Terminal.ecrireStringln("second constructeur de Neuvieme");
    }
}
class Dixieme extends Neuvieme{
    Dixieme(double d){
        super(true);
        Terminal.ecrireStringln("premier constructeur de Dixieme");
    }
    Dixieme(int x, boolean y){
        super(x);
        Terminal.ecrireStringln("second constructeur de Dixieme");
    }
}
class Exo19_1_4{
    public static void main(String[] args){
        new Dixieme(10.5);
        Terminal.ecrireStringln("=====");
        new Dixieme(45, true);
    }
}

class Neuvieme extends Premiere{
    Neuvieme(int i){
        Terminal.ecrireStringln("premier constructeur de Neuvieme");
    }
    Neuvieme(boolean b){
        Terminal.ecrireStringln("second constructeur de Neuvieme");
    }
}
class Dixieme extends Neuvieme{
    Dixieme(double d){
        super(true);
        Terminal.ecrireStringln("premier constructeur de Dixieme");
    }
    Dixieme(int x, boolean y){
        super(x);
        Terminal.ecrireStringln("second constructeur de Dixieme");
    }
}
class Exo19_1_4{
    public static void main(String[] args){
        new Dixieme(10.5);
        Terminal.ecrireStringln("=====");
        new Dixieme(45, true);
    }
}
```

1. **prédez** ce que ce programme affiche.
2. **compilez** et **exédez** ce programme pour vérifier vos prédictions.

## Exercice 20.2 - volaille

Un éleveur de volaille reçoit d'un fournisseur de jeunes canards et de jeunes poulets qu'il élève jusqu'à ce qu'ils aient la taille nécessaire à leur commercialisation.

Une volaille est caractérisée par son poids et un numéro d'identification reporté sur une bague qu'elle porte à sa petite patte. Les volailles arrivent à l'élevage à l'âge de trois semaines. Elles sont baguées et enregistrées dans le système informatique.

Il y a deux sortes de volailles : des canards et des poulets.

Le prix du canard et celui du poulet sont deux prix différents, exprimés en euros par kilo.

En revanche, le prix est le même pour tous les individus de la même espèce.

Ce prix varie chaque jour. Le poids auquel on abat les bêtes est différents pour les canards et les poulets, mais c'est le même pour tous les poulets (respectivement, tous les canards).

**Ecrivez** une classe des volailles avec deux sous-classes des poulets et des canards.

Il faut pouvoir enregistrer les prix du jours, les poids d'abattage, le poids d'une volaille donnée.

Ecrivez une classe permettant de représenter l'ensemble des animaux de l'élevage au moyen d'un tableau. Des méthodes doivent permettre de trier les animaux à abattre et d'évaluer le prix obtenu pour ces animaux. Il faut également pouvoir enregistrer les jeunes animaux qui arrivent.