

XX.2 Static et Exception: Récapitulatif

On considère la classe `CompteException` suivante:

```
1  import java.util.Scanner;
2
3  public class CompteException {
4      static int appelsMethodesClasse;
5          int appelsMethodesInstance;
6      private int[] tab;
7
8      public void setTable(String[] stringTab) {
9          appelsMethodesClasse++;
10         appelsMethodesInstance++;
11
12         tab = new int[stringTab.length];
13         for (int i = 0; i < stringTab.length; i++) {
14             tab[i] = Integer.parseInt(stringTab[i]);
15         }
16     }
17
18     public int readInt() {
19         appelsMethodesClasse++;
20         appelsMethodesInstance++;
21         System.out.print("Veuillez entrer un entier: ");
22         Scanner scan = new Scanner(System.in);
23         int i = scan.nextInt();
24         return i;
25     }
26
27     public void setTable(int n) {
28         appelsMethodesClasse++;
29         appelsMethodesInstance++;
30
31         tab = new int[n];
32         for (int i = 0; i < n; i++) {
33             tab[i] = readInt();
34         }
35     }
36
37     public String toString() {
38         String str = "[";
39         for (int i = 0; i < tab.length; i++) {
40             str += tab[i];
41             if (i < tab.length - 1) {
42                 str += ", ";
43             }
44         }
45         str += "]";
46         return str;
47     }
48 }
```

Question 2.1

Combien la classe `CompteException` a-t-elle d'attribut(s) ?

Question 2.2

Que fait la méthode + `readInt()`: `int` ?

Question 2.3

Que fait la méthode + `setTable(n: int)`: `void` ?

Question 2.4

Que fait la méthode + `setTable(stringTab: String[])` ?

On considère à présent le programme ci-dessous:

```

1 public class TestCompteException {
2
3     public static void main(String[] args) {
4         System.out.println("Liste des arguments: ");
5
6         for (int no = 0; no < args.length; no++) {
7             System.out.println("argument numéro " + no + ": " + args[no]);
8         }
9
10        System.out.println("---- Premier Tableau ----");
11        CompteException compte01 = new CompteException();
12        compte01.setTable(args);
13        System.out.println("Tableau: " + compte01.toString());
14        System.out.println("appelsMethodesClasse: " + CompteException.
15            appelsMethodesClasse);
16        System.out.println("appelsMethodesInstance: " + compte01.
17            appelsMethodesInstance);
18
19        System.out.println("---- Deuxième Tableau ----");
20        CompteException compte02 = new CompteException();
21        compte02.setTable(3);
22        System.out.println("Tableau: " + compte02.toString());
23        System.out.println("appelsMethodesClasse: " + CompteException.
24            appelsMethodesClasse);
25        System.out.println("appelsMethodesInstance: " + compte01.
26            appelsMethodesInstance);
27    }
28 }

```

Question 2.5

De quel type est le paramètre de la méthode `main`

Question 2.6

Dessinez le diagramme APO de ce qui se passe à l'exécution lorsque l'on lance ce le programme ci-dessus de la façon suivante et que l'on entre ensuite les 3 nombres 30 40 et 50.

```

1 java TestCompteException 10 15 20

```

Question 2.7

En déduire ce qui est affiché à l'exécution.

Question 2.8

Que se passe-t-il si l'on exécute l'instruction suivante ?:

```

1 java TestCompteException 10 15 20

```

Question 2.9

Que se passe-t-il si l'on exécute l'instruction suivante ?:

```

1 java TestCompteException bonjour

```

Question 2.10

Corrigez le code pour que le programme ne *plante* pas dans ce cas, mais inscrive un message indiquant que l'utilisateur doit rentrer un ou plusieurs entiers.

Question 2.11

Que se passe-t-il si, lorsque l'on demande à l'utilisateur d'entrer un entier (deuxième partie du programme), celui-ci entre autre chose ?

Question 2.12

Corrigez le code de la deuxième partie du programme pour que l'application ne *plante* pas non plus dans ce cas...