Exercises 16

exceptions

Exercise 16.1 - Predefined exceptions

There are predefined exceptions in Java:
- `ArithmeticException`: it occurs when a division by zero.
- `NullPointerException`: it occurs when trying to use an object or uninitialized array (on which we have not made `new`).
- `ArrayIndexOutOfBoundsException`: it is an error on the index of an array.
It seeks access to a box that does not exist.

The common point of these exceptions is that they are a class of exception that you do not need to be declared in the `throws` clause of methods because they are common.
Complete the following program to manage the errors that may occur until a calculation is actually carried through.

```
class Exo15_1{
    static int[] tableau = {17, 12, 15, 38, 29, 157, 89, -22, 0, 5};
    static int division(int indice, int diviseur){
        return tableau[indice]/diviseur;
    }
    public static void main(String[] args){
        int x, y;
        Terminal.ecrireString("Entrez l'indice de l'entier à diviser: ");
        x = Terminal.lireInt();
        Terminal.ecrireString("Entrez le diviseur: ");
        y = Terminal.lireInt();
        Terminal.ecrireString("Le résultat de la division est: ");
        Terminal.ecrireIntln(division(x,y));
    }
}
```

Exercise 16.2 - *reading* exercise

What is displayed by the following program if the entered integer is 3? Same question for 0, 1 and 2. Do this exercise on paper with self-correction by *running* the program.

```
class Exo15_2{
    static void methode1(int p) throws Exc1, Exc2{
        Terminal.ecrireStringln("Debut d'execution de methode1");
        if (p == 0){
            throw new Exc1 ();
        }
        Terminal.ecrireStringln("Milieu d'execution de methode1");
        if(p ==1){
            throw new Exc2 ();
        }
        Terminal.ecrireStringln("Fin d'execution de methode1");
    }
```

```
    static void methode2(int p) throws Exc1, Exc2, Exc3{
        Terminal.ecrireStringln("Debut d'execution de methode2");
        if (p == 2){
            throw new Exc3();
        }
        Terminal.ecrireStringln("Milieu d'execution de methode2");
        methode1(p);
        Terminal.ecrireStringln("Fin d'execution de methode2");
    }
    static void methode3(int p) throws Exc1, Exc3{
        Terminal.ecrireStringln("Debut d'execution de methode3");
        try{
            methode2(p);
        }catch(Exc2 ex){
            Terminal.ecrireStringln("Debut d'execution de methode3");
        }
        Terminal.ecrireStringln("Fin d'execution de methode3");
    }
    public static void main(String[] args) throws Exc1, Exc3{
        Terminal.ecrireString("Entrez un nombre: ");
        methode3(Terminal.lireInt());
    }
}
class Exc1 extends Exception{}
class Exc2 extends Exception{}
class Exc3 extends Exception{}
```

Exercise 16.3 - Menu

This is an entry to a menu of choices.

Question 16.3.1 - thrown exception

We build a method `saisirChoix` that takes parameter **n** and returns an integer value between 1 and **n**, typed at the keyboard by the user.
The various errors that can occur are:
- **n** is less than or equal to
- the user enters a number which is not between 1 and n
- user did not enter a number

Each error must be detected by the program and be identified by a specific exception.
So we need three different classes of exception.
For simplicity, you can limit n to n <10, which allows us to enter a single character using `Terminal.lireChar ()`.

Question 16.3.2 - menu display

Write a method that takes as a parameter an array of strings. Each string describes a choice of menu. The method must display these choices.

16.3.3 Question - Question and Answer

Write a method that uses two methods already written to display a menu, and enter the choice of the user.
This method will handle the three exceptions defined in question 1. It takes as input an array of options (String) and at output the choice made (as an integer).

Question 16.3.4 - Class Menu

Write a Menu class that has a constructor to initialize an array of strings representing different choices. This class will provide the input of a choice by an integer, using the necessary exception.
This class will include the features developed in the previous question, but adapted to the class structure.
Write a program that uses this class menu with three different menus for three different objects. This program will capture the exceptions that can be raised.