

TP7 : Les exceptions**Enseignants : Madhi Kamel, Majdoub Manel****Groupes : PrepaA02****Objectif : manipuler les exceptions****Exercice1 : exceptions prédéfinies**

Il existe en java des exceptions prédéfinies :

- ArithmeticException : elle se manifeste lors d'une division par zéro.
- NullPointerException : elle se manifeste lorsqu'on cherche à utiliser un objet ou un tableau non initialisé (sur lequel on n'a pas fait de new).

- ArrayIndexOutOfBoundsException : il s'agit d'une erreur sur l'indice d'un tableau. On cherche à accéder à une case qui n'existe pas.

Complétez le programme suivant pour que les erreurs susceptibles de se produire soient gérées jusqu'à ce qu'un calcul soit effectivement mené à bout.

```
public class Exercice1 {
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;
System.out.println("Entrez l'indice de l'entier à diviser :");
Scanner sc=new Scanner(System.in)
x = sc.nextInt();
System.out.println("Entrez le diviseur :");
y= sc.nextInt();
System.out.println("Le résultat de la division est :");
System.out.println(division(x,y)); } }
```

Exercice2 : exceptions personnalisées

On désire gérer la sélection des candidats dans un concours. Pour cela, on définit les informations nécessaires des candidats autorisés à participer à ce concours :

- Nom : Nom du candidat
- Prénom : Prénom du candidat
- Numéro : Numéro du candidat
- Age : L'age du candidat. Il ne doit pas dépasser 30 ans.

Les fonctionnaires, eux aussi, peuvent participer à ce concours s'ils remplissent les conditions de participation, en particulier, la condition d'age.

1. Créez la classe Candidat qui contient les 4 attributs cités ci-dessus.
 - Ajoutez un constructeur qui a quatre arguments ;
 - Ajoutez une méthode afficher() qui affiche toutes les informations d'un candidat.
2. Créez une classe Fonctionnaire qui hérite de la classe Candidat et qui :
 - ajoute aux propriétés d'un candidat le numéro de somme (Num_Somme en entier).
 - redéfinit la méthode afficher().

3. Modifiez la classe Candidat en ajoutant une exception qui sera déclenchée si l'âge associé au candidat est supérieur à 30.
4. Créez une classe Concours qui contient une méthode main(), dans laquelle :
 - déclarez (instanciez) un candidat ;
 - instanciez un fonctionnaire ;
 - déclarez une collection de type ArrayList et ajoutez le candidat et le fonctionnaire à cette collection ;
 - afficher les informations du candidat et du fonctionnaire en utilisant cette collection.

Exercice 3 : exceptions d'E/S

On veut écrire une application qui lit un fichier texte supposé exister, affiche son contenu, ainsi que le nombre de lignes. Le chemin d'accès au fichier à ouvrir est demandé à l'utilisateur.

1. Écrire l'application demandée sans chercher à capturer les exceptions susceptibles d'être générées. Pour créer un fichier texte, vous pouvez utiliser n'importe quel éditeur de texte. Vous pouvez aussi écrire une autre application qui crée un fichier texte.
 Voir le programme suivant qui vous donnent un exemple de lecture et écriture de fichiers textes.
 Testez en particulier le cas où le nom (ou chemin) entré par l'utilisateur ne correspond pas à un fichier existant.
2. On veut maintenant gérer l'exception "FileNotFoundException" en capturant cette exception, en signalant l'erreur à l'utilisateur, et en lui demandant d'entrer un nouveau nom de fichier, et ce, jusqu'à ce que l'ouverture aboutisse ou que l'utilisateur abandonne la tentative d'ouverture.

```

/* Exemple de programme lisant un fichier texte nommé "essai.txt" */
import java.io.*;
public class Text2 { public static void main(String args []) throws IOException {
  BufferedReader lectureFichier = new BufferedReader( new FileReader
    ("essai.txt"));
  System.out.println("Lecture du fichier essai.txt");
  String s = lectureFichier.readLine();
  /* readLine() retourne :
  -la ligne lue jusqu'au retour chariot (lu mais non retourné) donc une chaîne vide si
    la ligne ne comporte qu'un RC
  -la valeur null s'il n'y a rien à lire (fin du flux de données) */
  while (s != null) {
    System.out.println(s);
    s = lectureFichier.readLine();
  }
  lectureFichier.close();
  System.out.println("Fin du fichier"); }

```
