

## TD02 : Les chaînes de caractères

### Objectifs

- Savoir déclarer et utiliser des chaînes de caractères en Java.
- Savoir convertir une chaîne de caractères en entier.
- Savoir utiliser les méthodes de traitement de chaîne de caractères de la classe `String`.
- Différencier l'opérateur `==` et des méthodes `equals()`, et `compareTo()` de la classe `String`.
- Se familiariser avec les exceptions en Java.

### Exercice 1. Conversion de type

En Java, les chaînes de caractères sont représentées par des objets de la classe `String`<sup>1</sup>, dont les instances sont non mutables. On souhaite pouvoir convertir des chaînes de caractères contenant des nombres en valeur entières.

1. Le faire en utilisant la méthode statique `static int parseInt(String)` de la classe `Integer`. Définir une chaîne de caractères contenant un entier et obtenir sa valeur sous forme d'entier (`int`). Faire attention au fait que la méthode `parseInt` peut lever l'exception `NumberFormatException`.
2. Déclarer une chaîne de caractères contenant un nombre réel et obtenir sa valeur sous la forme d'un flottant (`double`).

### Exercice 2. Evaluation d'expression

Écrire un programme prenant sur la ligne de commande une expression mathématique en notation préfixée, de la forme `opérateur nombre nombre` et qui l'évalue.

### Exercice 3 (Références et égalités)

1. Créer un tableau `t` de 4 références sur des chaînes de caractères.
2. Créer une chaîne de caractères dont le contenu est "Bon".
3. Affecter les 2 premiers éléments du tableau avec cette chaîne et le troisième avec la chaîne "Bonjour".
4. Créer une nouvelle chaîne de caractères dont le contenu est à nouveau "Bon" et affecter cette chaîne au dernier élément du tableau `t`.
5. Comparer 2 à 2 les éléments du tableaux en utilisant les méthodes `String.equals()`, `String.compareTo()`, et l'opérateur `==`. Que constate-t-on ?
6. Concaténer `t[0]` avec la chaîne "jour" en utilisant la méthode `String.concat()` ou l'opérateur `+`. Comparer 2 à 2 les éléments du tableaux en utilisant les méthodes `String.equals()`, `String.compareTo()`, et l'opérateur `==`. Que constate-t-on ?

<sup>1</sup> <https://docs.oracle.com/javase/7/docs/api/java/lang/String.html>



### Exercice 4 (Références et égalités)

Déclarer un tableau `t1` contenant les valeurs 2, 7, 5, 21 et -7. Recopier `t1` dans un tableau `t2`.  
Comparer les tableaux `t1` et `t2` en utilisant l'opérateur `==` et la méthode `equals`.

### Exercice 5 (chaines de caractères, structures de contrôle)

1. Ecrire un programme qui teste la similarité de deux chaînes de caractères et affiche vrai ou faux.
2. Ecrire un programme qui remplace un caractère dans une chaîne de caractère par un espace.
3. Ecrire un programme qui détermine si un mot est un palindrome ou non (Un mot est dit palindrome s'il peut se lire indifféremment des deux sens).