

# NFA031 – TP4.2

## Exercices sur le type String (1)

24 octobre 2017

### Utiliser le type String

Créez un nouveau fichier `ExempleChaine.java` dans lequel vous allez taper tout le code donné dans les explications qui suivent. Vous comprendrez mieux comment cela fonctionne !

Le type `String` est prédéfini en Java et permet de modéliser les chaînes de caractères. Il est introduit au chapitre 6 des notes de cours, et nous aurons l'occasion de l'étudier en fin de notre module. Dans ce Tp nous en ferons usage de manière relativement simple, et les explications données ici devraient suffire. Pour aller plus loin, lisez le chapitre 6.

### Les constantes

Une constante `String` est formé de 0 ou plusieurs caractères entourés de guillemets doubles. Par exemple : `"ab"`, `"1234"`, `"34 + ("` et `"*"` sont des chaîne de caractères. La chaîne vide, autrement dit, celle qui ne contient aucun caractère s'écrit `" "`. Il ne faut pas confondre le type `String` et le type `char`. Par exemple `"a"` est de type `String` et `'a'` est de type `char` : ils ne sont pas des valeurs équivalentes ni interchangeables. On peut déclarer des variables de type `String`.

---

```
String s1 = "";  
String s2 = "Y8_>-?";  
String s3 = "q";
```

---

### Indice des caractères

Dans une chaîne, chaque caractère a une position qu'on appelle son *indice*. Le premier caractère d'une chaîne a pour indice 0, le dernier a pour indice la taille de la chaîne moins 1. Par exemple, dans la chaîne `"Y89>-?"`, le caractère `'Y'` se trouve à l'indice 0, `'8'` à l'indice 1, `'9'` à l'indice 2, `'>'` à l'indice 5.

### Les méthodes sur les chaînes

Il existe des nombreuses méthodes prédéfinies sur les chaînes (voir la documentaton sur le site d'Oracle). Dans ce Tp nous utiliserons principalement les deux méthodes suivantes :

- `s.length()` : renvoie la taille (nombre de caractères) de la chaîne `s`.
- `s.charAt(i)` : renvoie le caractère se trouvant à l'indice `i` de la chaîne `s`.

Du fait que les chaînes sont un type objet en Java, on invoque leurs méthodes avec une syntaxe particulière. On doit faire suivre la variable ou constante String à traiter du nom de la méthode plus paramètres que l'on souhaite lui appliquer. Par exemple, si on souhaite obtenir la taille de la chaîne `s2` on écrira : `s2.length()`. Voici quelques exemples (sur les variables définies plus haut).

---

```
int x = s1.length(); // x <-- 0
char a = s2.charAt(0); // a <-- 'Y'
"q".charAt(1); // erreur: l'indice 1 n'existe pas
```

---

## Parcourir une chaîne

On peut parcourir les caractères d'une chaîne à l'aide d'une boucle. Par exemple, cette boucle permet d'afficher les caractères d'une chaîne, chacun sur une ligne et précédé par son indice dans la chaîne :

---

```
String s = Terminal.lireString(); // ou autre valeur initiale
for (int i=0; i<s.length(); i++){
    System.out.print("Indice_" + i + "_:_");
    System.out.println(s.charAt(i));
}
```

---

## Consignes

Pour tous les programmes de ce TP vous devez :

- *Documenter le programme* : ajouter au moins un commentaire dans son entête qui décrit son comportement en termes des entrées et sorties (il faut répondre à la question *quoi* et non pas à la question *comment*). Ajouter également votre nom, vous êtes l'auteur. Inspirez vous des programmes fournis dans l'exercice 1 pour rédiger cette documentation.
- *Produisez un jeu de tests* : suffisamment représentatif de tous les cas des entrées et sorties du programme.
- Utilisez votre jeu de tests pour testert votre programme. N'oubliez pas de refaire **tous** vos tests après toute modification : ce que marchait avant, peut ne plus marcher après vos changements.

## Exercice 1 : Tester, modifier

Récupérez les programme fournis `ParcoursChaine.java`.

1. Testez ce programme sur plusieurs exemples.
2. Pour vous entraîner, réécrivez ce programme à l'aide d'une boucle `while`.
3. Modifiez le programme d'origine afin d'afficher les caractères de la chaîne séparés par un tiret. Par exemple, la chaîne "salut" sera affichée `s-a-l-u-t`.
4. Modifiez le programme d'origine afin d'afficher les caractères de la chaîne dans l'ordre inverse. Par exemple, la chaîne "salut" sera affichée `tulas`.

## **Exercice 2 : occurrences**

Ecrivez un programme qui lit une chaîne s et ensuite un caractère c. Votre programme doit calculer et afficher le nombre de fois qu'apparaît le caractère c dans la chaîne s.

## **Exercice 3 : fabriquer une nouvelle chaîne**

Ecrivez un programme qui lit une chaîne s et un caractère c. Votre programme doit fabriquer une nouvelle chaîne s2 qui contient tous les caractères de s dans le même ordre, sauf les occurrences de c. Par exemple, si la chaîne s est "oh la la" et le caractère c est 'l', la nouvelle chaîne s2 doit être : "oh a a".

## **Exercice 4 : tester un mot**

Ecrivez un programme qui lit une chaîne s et qui teste si cette chaîne est un mot, autrement dit, si elle ne contient que des lettres (majuscules ou minuscules).