

## TP 1 (POO)

### L'environnement JAVA

Ce premier TP concerne la compilation et l'exécution d'un premier programme écrit en Java et la conception de programmes Java élémentaires. Toute la documentation JAVA se trouve à l'adresse <http://docapi.iutlan.univ-rennes1.fr/>.

#### Exercice 1 : découverte de la documentation en ligne

Accédez à cette documentation en ligne. Vous pouvez observer trois parties dans l'écran qui s'affiche. Que représente chacune de ces trois parties ?  
Le menu de la partie centrale est le suivant :



#### Java™ 2 Platform Standard Edition 5.0 API Specification

Quels sont les liens actifs ici?

Cliquez sur le lien 'Help'. Comment est organisée la documentation en ligne ?

Comment savoir à quoi sert le package java.io ? A quoi sert le package java.lang ?

Dans la fenêtre principale, cliquez sur le lien java.lang.

Que constatez-vous au niveau du menu du haut de la fenêtre ?

A quoi sert la classe System ?

On s'intéresse à l'instruction suivante : `System.out.println(data)`.

Pourquoi le mot 'System' a-t-il une majuscule ? Que représente le mot 'out' ? Que représente le mot 'println' ?

On s'intéresse à l'instruction

```
BufferedReader entree=new BufferedReader  
                                (new InputStreamReader (System.in));
```

Que représente le mot `BufferedReader` ? Comment trouver rapidement les informations sur cette classe dans la documentation en ligne ?

Que représentent `System.in`, `InputStreamReader` ?

#### Exercice 2

Vous pouvez saisir le code source d'une application JAVA avec n'importe quel éditeur de texte. La compilation sous l'invite Unix se fait par la commande *javac* :

```
javac Monappli.java
```

Le fichier à compiler doit avoir l'extension .java. Son exécution se fait par la commande *java* :

```
java Monappli
```

Créez un répertoire pour les TP de POO, puis un sous-répertoire pour ce TP1.  
Ecrivez un programme qui affiche la chaîne de caractères *bonjour* à l'écran.

Compilez-le. Regardez le contenu de votre répertoire. Que constatez-vous ? Exécutez votre programme.

### Exercice 3

But de cet exercice: vous faire explorer la classe `java.lang.String` en testant ses diverses méthodes sur des chaînes et d'autres valeurs lues au clavier et de vous initier à la consultation de cette documentation.

Pour cela, ouvrez un navigateur sur la doc de l'API ; dans le volet gauche supérieur sélectionnez le paquetage `java.lang` puis dans le volet gauche inférieur sélectionnez la classe `String`.

Ouvrez une autre fenêtre avec votre éditeur et texte, et une autre avec le terminal pour compiler et tester vos programmes.

Il s'agit d'écrire une classe exécutable `TestChaines` dont la méthode principale effectue les opérations suivantes :

A. Créez une variable de type `int`, affectez-lui une valeur, puis convertissez cette variable en chaîne (ex.: le nombre 12345 devient la chaîne "12345"). Utilisez la méthode `valueOf()` de la classe `String`.

B. Au clavier, lire une chaîne entièrement composée de chiffres et la convertir dans le nombre entier qu'elle représente (ex.: la chaîne "12345" devient le nombre 12345). La solution se trouve parmi les méthodes statiques de la classe `java.lang.Integer`. Après affichage du nombre entier, vous lui ajouterez 1 et afficherez le nouveau nombre obtenu. Utilisez la méthode `parseInt()` de la classe `Integer`.

C. Même question que ci-dessus, mais avec un nombre flottant (ex.: la chaîne "0.12345e4" devient le nombre 0.12345e4). Après affichage du nombre, vous lui ajouterez 1.1 et afficherez le nouveau nombre obtenu. Utilisez la méthode `parseFloat()` de la classe `Float`.

D. Lire une chaîne représentant un nom de ville, lui enlever les éventuels blancs au début et à la fin et l'afficher entièrement en majuscules.

E. Lire deux chaînes `s1` et `s2` et afficher la réponse à la question: «ces deux chaînes commencent-elles par le même caractère?» Utilisez la méthode d'instance `charAt`.

F. Lire deux chaînes `s1` et `s2` et afficher les résultats renvoyés par les expressions «`s1==s2`», «`s1.equals(s2)`», «`s1.compareTo(s2)`» et «`s1.compareToIgnoreCase(s2)`».

Entre autres, essayer les couples "abcd" et "abcd", puis "abcd" et "AbcD"

G. Lire deux chaînes `s1` et `s2` et afficher la réponse aux questions 's1 commence-t-elle par s2 ?', 's1 finit-t-elle par s2 ?', 's1 contient-elle s2 ?' On utilisera les méthodes `startsWith()`, `endsWith()` et `contains()`.

H. Lire deux chaînes `s1` et `s2` et si `s1` contient `s2`, afficher `s1` privée de `s2` (s'intéresser à `substring` et `indexOf`), sinon afficher `s1`.

D'une façon générale, tout langage de programmation comporte des méthodes diverses et variées sur les chaînes de caractères. Sachez repérer dans la documentation, la méthode qui répond à vos besoins.

### Exercice 4 ( Calculatrice)

Dans cet exercice, comme dans tous ceux qui suivront, vous vous aiderez de la documentation en ligne. Le but de cet exercice est de réaliser une petite calculatrice qui calcule le cosinus, le sinus ou la tangente d'un angle. Vous utiliserez les méthodes de classe de la classe `Math` pour effectuer les calculs : méthodes de classe `cos(double a)`, `sin(double a)` et `tan(double a)` dont le résultat est de type primitif `double` et dont l'argument est la mesure d'un angle en radian.

Le menu demande le type d'opération à effectuer (cosinus, sinus ou tangente) et se présente sous la forme suivante :

Calculatrice

Cosinus : tapez C

Sinus : tapez S

Tangente : tapez T

Fin :

tapez F

Pour récupérer le choix de l'utilisateur ('C', 'S', 'T' ou 'F'), il faut faire appel à la méthode `charAt()` de la classe `java.lang.String`.

Puis le programme demande à l'utilisateur de saisir une valeur de la mesure d'un angle. Le programme lit alors cette valeur au clavier et affiche le résultat de l'opération (tant que l'utilisateur n'a pas quitté le menu). Après la saisie de la mesure de l'angle en radian, il faut convertir cette saisie en type primitif `double`.

Expliquez pourquoi en vous aidant de la documentation.

Voici quelques méthodes nécessaires aux conversions :

la méthode de classe `valueOf(String str)` de la classe `Double` a comme argument une instance de classe `String` (elle délivre la conversion de la chaîne en instance de `Double`),

la méthode d'instance `doubleValue()` délivre la conversion d'une instance de `Double` en type primitif `double`.

Votre programme devra suivre l'algorithme suivant:

Votre programme devra suivre l'algorithme suivant:

- Affichage du menu sous la forme précisée ci-dessus.
- Saisie du choix de l'utilisateur ('C', 'S', 'T' ou 'F')
- Récupération du choix dans une variable de type `char`, puis tant que ce choix est différent de 'F' :
- Affichage de la proposition de saisir une valeur en radian
- Saisie de la mesure en radian
- Conversion de cette mesure en type `Double`, puis `double`.
- Calcul du cosinus, du sinus ou de la tangente selon le choix de l'utilisateur
- Affichage du résultat du calcul.