

DEV1 – Laboratoires Java I**TD 4 – Boucles for**

Ce TD introduit l'instruction `for` ainsi que la notion de chaîne de caractères et le type associé `String`.

Les codes sources de ce TD se trouvent à l'adresse :

<https://git.esi-bru.be/dev1/labo-java/tree/master/td04-boucles-for/code/>

Les solutions se trouvent à l'adresse :

<https://git.esi-bru.be/dev1/labo-java/tree/master/td04-boucles-for/solutions/>

Table des matières

1	Boucle - for	2
2	Les chaînes de caractères	2
3	Exercices Récapitulatifs	4

1 Boucle - for

```
1 package esi.dev1.td4;
2
3 public class BoucleFor {
4
5     public static void main(String[] args) {
6         for(int i=1; i<=10; i=i+1) {
7             System.out.println(i);
8         }
9     }
10 }
```

BoucleFor.java

Le code ci-dessus affiche les nombres de 1 à 10. Ce programme s'exécute comme suit :

- ▷ le programme commence à l'instruction **for** de la ligne 6 :
 - ▷ l'initialisation **int i = 1** est d'abord exécutée : la variable *i* de type entier est déclarée et initialisée à 1 ;
 - ▷ ensuite, la condition du **for** **i<=10** est évaluée, sa valeur est vrai car *i* vaut 1 et est donc inférieur à 10 ;
 - ▷ puisque la condition est vraie le *corps* de la boucle s'exécute, la ligne 7 affiche la valeur de l'entier *i* : 1. Et l'incrément **i = i + 1** est exécuté, *i* vaut 2 maintenant ;
 - ▷ la condition est à nouveau évaluée avec la nouvelle valeur de *i* qui vaut maintenant 2 ;
 - ▷ etc...
 - ▷ le programme continue ainsi jusqu'à ce que *i* atteigne la valeur 11. À ce moment la condition est évaluée à faux et l'instruction **for** prend fin.
- ▷ Comme aucune instruction ne suit ce **for**, le programme se termine.

Exercice 1 Suites d'entiers

Dans un package `g12345.dev1.td4` créez une classe `Exercice1`. Dans cette classe écrivez un programme qui demande à l'utilisateur un nombre entier *n* et affiche à l'aide d'une boucle **for** :

- ▷ les nombres de 1 à *n* ;
- ▷ les nombres pairs qui sont compris entre 1 et *n* ;
- ▷ les nombres de -*n* à *n* ;
- ▷ les multiples de 5 qui sont compris entre 1 et *n* ;
- ▷ les multiples de *n* compris entre 1 et 100.

2 Les chaînes de caractères

Un caractère se représente en Java avec des guillemets simples. Par exemple, `'a'` représente le caractère a et `'1'` représente le caractère 1 (et non pas l'entier 1). Une variable de type `char` permet de manipuler les caractères. On écrira par exemple `char lettre = 'a';`

En Java, le texte est représenté par une chaîne de caractères (une suite de caractères). Le type associé est `String`. Une variable de type `String` permet de stocker et manipuler une chaîne de caractères : `String mot = "Bonjour";`

```

1 package esi.dev1.td4;
2
3 import java.util.Scanner;
4
5 public class Texte {
6
7     public static void main(String[] args) {
8         Scanner clavier = new Scanner(System.in);
9         String texte = clavier.nextLine(); //lit du texte au clavier
10
11         System.out.println(texte);
12         System.out.println("Le texte entré est: "+texte);
13
14         int longueur = texte.length(); //le nombre de caractères
15         System.out.println("La longueur du texte est: "+longueur);
16
17         char premièreLettre = texte.charAt(0);
18         char dernièreLettre = texte.charAt(longueur-1);
19         System.out.println(premièreLettre);
20         System.out.println(dernièreLettre);
21
22
23         for(int i=0; i< texte.length(); i=i+1) { // parcours la chaine de caractères
24             System.out.print(texte.charAt(i));
25         }
26
27         System.out.println(); //passe à la ligne
28
29         for(int i=texte.length()-1; i>=0; i=i-1) { // parcours la chaine de caractères à
            ↪ l'envers
30             System.out.print(texte.charAt(i));
31         }
32
33         System.out.println(texte.equals("Bonjour")); //affiche true si le texte entré est
            ↪ "Bonjour"
34     }
35 }

```

Texte.java

Exercice 2 **Voyelle**

Écrire un programme qui demande à l'utilisateur un mot et affiche si la première lettre est une voyelle ou non.

Exercice 3 **Consonne**

Écrire un programme qui demande à l'utilisateur un mot et affiche si la première lettre est une consonne ou non.

Astuce : une lettre est une consonne si ce n'est pas une voyelle.

Exercice 4 **Première == dernière ?**

Écrire un programme qui demande à l'utilisateur un mot et affiche si la première lettre est la même que la dernière ou non.

Par exemple, si l'utilisateur entre "java", le programme affiche que la première et la dernière lettre ne sont pas égales.

Exercice 5 **Miroir**

Écrire un programme qui demande à l'utilisateur un mot et affiche son miroir.

Par exemple, si l'utilisateur entre "java", le programme affiche

avaj

Exercice 6 **Voyelles**

Écrire un programme qui demande à l'utilisateur un mot et affiche les voyelles.

Par exemple, si l'utilisateur entre "programmation", le programme affiche

oaaio

3 Exercices Récapitulatifs

Exercice 7 **Une phrase**

Écrire un programme qui demande à l'utilisateur une phrase et affiche si la première lettre est une majuscule et la dernière un point.

Par exemple si l'utilisateur entre "Bonjour Marco." votre programme affichera que c'est une phrase. Consulter Internet pour savoir comment vérifier qu'un caractère est une majuscule.

Exercice 8 **Nombre de voyelles et de consonnes**

Écrire un programme qui demande à l'utilisateur un mot et affiche le nombre de voyelles et de consonnes dans ce mot.

Par exemple, si l'utilisateur entre "programmation", le programme affiche "5 voyelles et 8 consonnes".

Exercice 9 **Palindrome**

Un palindrome est un mot dont la succession de lettres est la même de gauche à droite et de droite à gauche. Les mots été, ressasser ou kayak sont des palindromes.

Écrire un programme qui demande à l'utilisateur un mot et affiche si ce mot est un palindrome ou non.

Exercice 10 **Transformer un for en while**

Modifier vos solutions aux deux exercices précédents afin d'utiliser des **while** à la place des **for**.