

Développement I - Laboratoire de développement Java

TD2 - Alternatives

Dans ce TD vous trouverez xxx



Ce document est distribué sous licence Creative Commons Paternité - Partage à l'Identique 2.0 Belgique
(<http://creativecommons.org/licenses/by-sa/2.0/be/>).
Les autorisations au-delà du champ de cette licence peuvent être obtenues à www.heb.be/esi - pbt@he2b.be.

1 Entiers, réels et booléens

Au td1 nous avons joué avec des entiers, nous allons maintenant manipuler des nombres réels et des booléens.

En java les *réels* se représentent avec un point. Par exemple 12,3 s'écrira 12.3.

En informatique, on appelle *booléens* les deux valeurs 'vrai' et 'faux'. En java les booléens se représentent par **true** et **false**.

Dans l'exemple ci-dessous, la classe **Expression** se trouve dans le package **esi.dev1.td2**.

code/Expression.java

```
1 package esi.dev1.td2;
2
3 public class Expression {
4
5     public static void main(String[] args) {
6         System.out.println("12,3 + 13,5 = ");
7         System.out.println(12.3 + 13.5);
8
9         System.out.println(" 17 <= 20 ? ");
10        System.out.println(17 <= 20);
11    }
12 }
```

À la ligne 6, le programme affichera le texte entre guillemets : 12,3 + 13,5 =

À la ligne 7, le programme affichera le résultat de la somme de 12,3 et de 13,5 c'est-à-dire 27.8

À la ligne 9, le programme affichera le texte entre guillemets : 17 <= 20 ?

À la ligne 10, le programme affichera la valeur de l'expression $17 \leq 20$ c'est-à-dire **true**.

En java les *opérateurs* permettant de manipuler les réels sont :

- l'addition : +
- la soustraction : -
- la multiplication : *
- la division : /

On pourra donc écrire :

- 12.3-13.5
- 12.3*13.5
- 12.3/13.5

Le opérateurs pour comparer les entiers et les réels sont :

- plus petit : <
- plus petit ou égal : <=
- plus grand : >

- plus grand ou égal : `>=`
- égal : `==`

On pourra donc écrire :

- `12.3 <= 13.5`
- `12.3 > 13.5`
- `17 == (15+3)`

Ces expressions valent respectivement `true`, `false` et `false`.

Exercice 1

Créez un package `esi.dev1.td2`. Dans ce package créez une classe `Exercice1`. Dans cette classe créer un programme qui affiche la valeur des expressions suivantes :

- `12.3-13.5`
- `12.3*13.5`
- `12.3/13.5`
- `17 <= 20`
- `12.3 > 13.5`
- `17 == (15+3)`
- `12.3*13.5 <= 13.4*12.4`
- `12.3 * 13.5 >= 6*(13.4 + 12.4)`

2 Variables entières, réelles et booléennes

Au `td1` nous avons vu que les variables permettant de manipuler des entiers se *déclarent* avec le type `int`. Par exemple l'instruction :

```
int longueur = 12;
```

déclare une variable de type entier nommée `longueur` à laquelle on *assigne* la valeur 12.

En java les variables de type *réel* se déclarent avec le type `double`. Par exemple :

```
double taille = 1.92;
```

déclare une variable de type réel nommée `taille` à laquelle on *assigne* la valeur 1,92.

En java les variables de type *booléen* se déclarent avec le type `boolean`. Par exemple :

```
boolean jeSuisContent = true;
```

déclare une variable de type booléen nommée `jeSuisContent` à laquelle on *assigne* la valeur `true`. Ou encore :

```
boolean majeur = age > 18;
```

déclare une variable de type booléen nommée `majeur` à laquelle on *assigne* la valeur de l'expression `age > 18`, `majeur` vaudra `true` si la variable `age` a une valeur supérieure à 18 (au moment où s'exécute cette instruction).

On peut également stocker du texte dans des variables. En java les variables contenant du texte sont de type `String`.

Par exemple :

```
String nom = "John";
```

déclare une variable de type texte nommée `nom` à laquelle on *assigne* la valeur `"John"`.

Dans l'exemple ci-dessous, la classe `Variables` se trouve dans le package `esi.dev1.td2`.

code/Variables.java

```
1 package esi.dev1.td2;
2
3 public class Variables {
4
5     public static void main(String[] args) {
6         String nom = "John"; // déclare une variable de type texte, assigne la valeur "John"
7         double taille = 1.92; // déclare une variable de type réel, assigne la valeur 1.92
8         int année = 1958;
9
10        System.out.print("Bonjour"); // affiche le texte entre guillemet: Bonjour
11        System.out.println(nom); // affiche la valeur de la variable nom: John
12
13        System.out.print("Vous mesurez: ");
14        System.out.print( taille ); // affiche la valeur de la variable taille : 1.92
15        System.out.println(" mètres");
16
17        System.out.print("Vous avez: ");
18        System.out.print(2018-année); // affiche la valeur de l'expression 2018-année: 60
19        System.out.println(" ans");
20    }
21 }
```

Notez l'utilisation de `print` et `println` cette dernière instruction affiche le texte entre guillemet *et* passe à la ligne alors que le `print` simple ne passe pas à la ligne.