

Un programme avec des variables

```
int i , j ;  
i=11;           //i se fait attribuer la valeur 11  
j=9;  
int somme = i+j;  
println ( "La_somme_vaut_" +somme) ;
```

On observe que les variables sont de type `int`.

Les types principaux sont : `int`, `float`, `char`, `String`, `boolean`

- 1 écrire un programme qui calcule la somme de 3 valeurs de type *int*
- 2 tracer une ellipse de taille 60×40
- 3 tracer une ellipse 10 fois plus grande sur une surface de dessin de taille 800×600
- 4 tracer 4 rectangles collés

Les instructions

Une instruction = une commande qui définit une action :

- une **déclaration** de variable :

```
int x
```

- une **affectation** : donner une valeur (ou le résultat d'une expression) à une variable déclarée

```
x=9
```

- une **conditionnelle** (un **if**) : exécuter une suite d'instructions si une expression est vérifiée

```
if (x<10)  
    println(x)
```

```
int i=4;  
int j=9;  
int somme = i+j;  
if (somme<20){  
    println("La somme vaut " +somme);  
}
```

Les instructions

Une instruction = une commande qui définit une action :

- une **déclaration** de variable :

```
int x
```

- une **affectation** : donner une valeur (ou le résultat d'une expression) à une variable déclarée

```
x=9
```

- une **conditionnelle** (un **if**) : exécuter une suite d'instructions si une expression est vérifiée

```
if (x<10)
```

```
    println(x)
```

```
int i=4;
int j=9;
int somme = i+j;
if (somme<20){
    println("La_somme_vaut_" +somme);
}
```

Le point-virgule

- une instruction de déclaration ou d'affectation sera toujours suivie d'un point-virgule ;
- on dit que le point-virgule valide l'instruction
- nous écrirons une instruction par ligne par souci de clarté.
 - il est possible d'écrire tout le programme en une seule ligne

Le if-else

```
if (a<b){  
    println ("inf");  
} else {  
    println ("sup");  
}
```

à l'intérieur de chaque branche, on peut ajouter d'autres `if`

```
if (a<b){  
    println ("inf");  
} else {  
    println ("sup");  
    if (a==b){  
        println ("=");  
    }  
}
```

Le if-else

```
if (a<b) {  
    println ("inf");  
} else {  
    println ("sup");  
}
```

à l'intérieur de chaque branche, on peut ajouter d'autres `if`

```
if (a<b) {  
    println ("inf");  
} else {  
    println ("sup");  
    if (a==b) {  
        println ("=");  
    }  
}
```

Quelques programmes en mode texte

Utiliser des conditionnelles `if (. . .)` pour :

- afficher « somme » si $a = b + c$, où a , b et c représentent 3 variables initialisées au début du programme
- afficher la valeur minimale de 3 variables a , b et c
- afficher “vrai” si 3 entiers a , b et c sont triés $a < b < c$.

Quelques programmes en mode texte

Utiliser des conditionnelles `if (. . .)` pour :

- afficher « somme » si $a = b + c$, où a , b et c représentent 3 variables initialisées au début du programme
- afficher la valeur minimale de 3 variables a , b et c
- afficher “vrai” si 3 entiers a , b et c sont triés $a < b < c$.

Quelques programmes en mode texte

Utiliser des conditionnelles `if (. . .)` pour :

- afficher « somme » si $a = b + c$, où a , b et c représentent 3 variables initialisées au début du programme
- afficher la valeur minimale de 3 variables a , b et c
- afficher “vrai” si 3 entiers a , b et c sont triés $a < b < c$.