

Fiche récapitulative : Java

1. Types primitifs JAVA

	Types	Exemples
Booléen	boolean	Vrai / Faux
Entier	int	1
Réel	double	3.14
Caractère	char	's'
Chaîne de caractères	String	"ceci est une ligne"

Pour déclarer une variable

nomDe Type nomDeVariable ;

Ex : int unNombre ;

Ecriture

```
System.out.println("x :"+x) ;
```

```
unEntier = Lire.i() ;
```

```
unReel = Lire.d() ;
```

```
unCaractere = Lire.c() ;
```

```
uneChaine = Lire.s() ;
```

2. Opérations

Opérations arithmétiques

+ addition

- différence

* multiplication

/ division

% modulo

++ incrémentation de 1

-- décrémentation de 1

Opérations de comparaison

< plus petit que

> plus grand que

<= plus petit ou égal à

>= plus grand ou égal à

== égal à

!= différent de

! NON

&& ET

|| OU

Ex : Echange du contenu de 2 variables

```
int x, y, aux ;  
x=128 ;  
y=256 ;  
aux=x ;  
x=y ;  
y=aux ;
```

3. Schéma

IF / ELSE

```
If (condition) {  
  action 1  
} else {  
  action 2  
}
```

Ex : maximum de 2 entiers

```
int x, y ;  
int max ;  
if (x>y) {  
  max = x ;  
} else {  
  max = y ;  
}  
System.out.println("max="+max) ;
```

SWITCH

```
Switch (v) {  
  cas v1 : A1 ;  
}  
break  
...  
cas vn : An ;  
break ;  
default : A ;  
break ;
```

WHILE

```
While (condition) {  
  actions  
}
```

Ex : tant que ... faire

```
int n ;  
int somme = 0 ;
```

```
int i=1 ;
n = Lire.i() ;
while (i<=n) {
somme = somme + i ;
i = i+1 ;
System.out.println("somme au bout de "+n+"mois :"+somme) ;
```

DO / WHILE

```
do {
actions
} while (condition)
```

Ex :

```
int n ;
int somme = 0 ;
int i = 1 ;
n = Lire.i() ;
do {
somme = somme+i ;
i=i+1 ;
} while (i<=n) ;
System.out.println("somme au bout de "+n+" mois :"+somme) ;
```

FOR

```
For (i=v1 ; i<=v2 ; i=i+h) {
actions
}
```

Ex :

```
int n ;
int somme = 0 ;
int i ;
n = Lire.i() ;
for (i=1 ; i<=n ; i=i+1) {
somme = somme+i ;
}
System.out.println("somme au bout de "+n+" mois : "+somme) ;
```

4. Création d'un tableau

```
int [] unTableau = new int [5] ;
unTableau.length
dernier élément : unTableau[unTableau.length-1]
non modifiable : unTableau.length = 5
```

5. Ossature d'une méthode

```
static typeDuResultat nomMéthode  
(typeP1 p1, typeP2 p2, ...)    {  
}
```

!! Attention !!

- Ne pas redéfinir les paramètres
- Ne pas utiliser un nom de paramètre comme nom de variable locale