

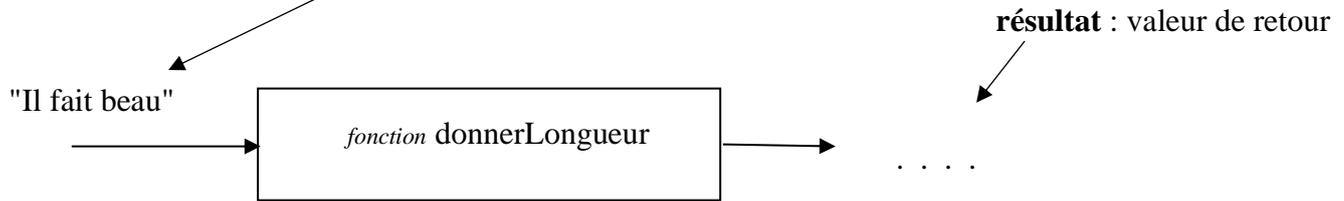
## I. Définition d'une fonction

## I.1 - Définition

Une **fonction** est un sous-programme qui fournit un service à celui qui l'utilise (l'utilisateur de la fonction) et retourne une valeur de retour.

Pour fonctionner, la fonction a besoin d'informations :

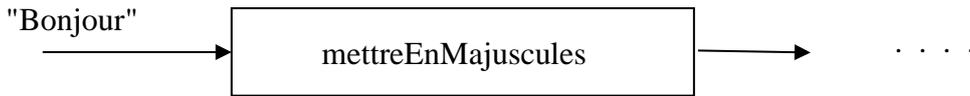
ce sont les **arguments**, ou **paramètres** : ils sont fournis en entrée à la fonction.



## I.2 - Exercice 1 : service rendu par une fonction

**Travail à faire** : Compléter les espaces vides laissés ci-dessous.

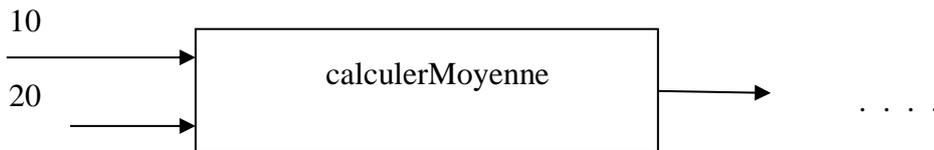
a)



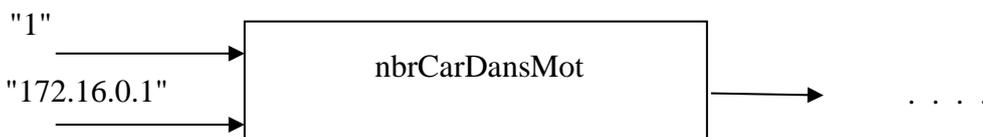
b)



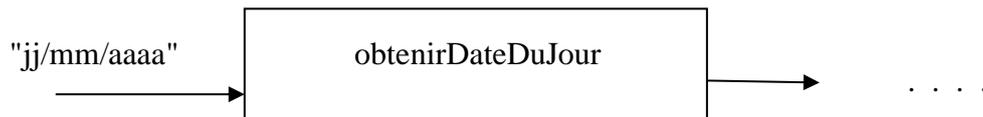
c)



d)



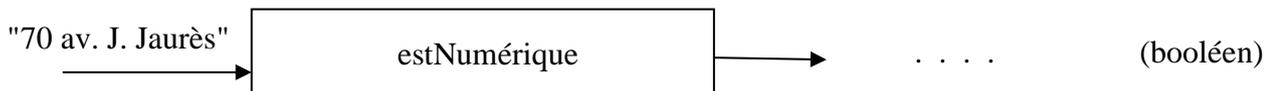
e)



f)



g)



## SIO 2 BLOC 2 – Ch. 4 – Révisions - Les fonctions

### II. Appel d'une fonction en Java

Travail à faire : Compléter les espaces vides laissés ci-dessous.

#### Exemple 1 :

```
String texte = "Il fait beau";
int longueur = getLongueur(texte);
System.out.println("longueur du texte : " + longueur);
```

	texte	longueur
String texte = "Il fait beau";	<input type="text"/>	<input type="text"/>
int longueur = <b>getLongueur</b> (texte);	<input type="text"/>	<input type="text"/>

Qu'est-ce qui s'affiche ? . . . . .

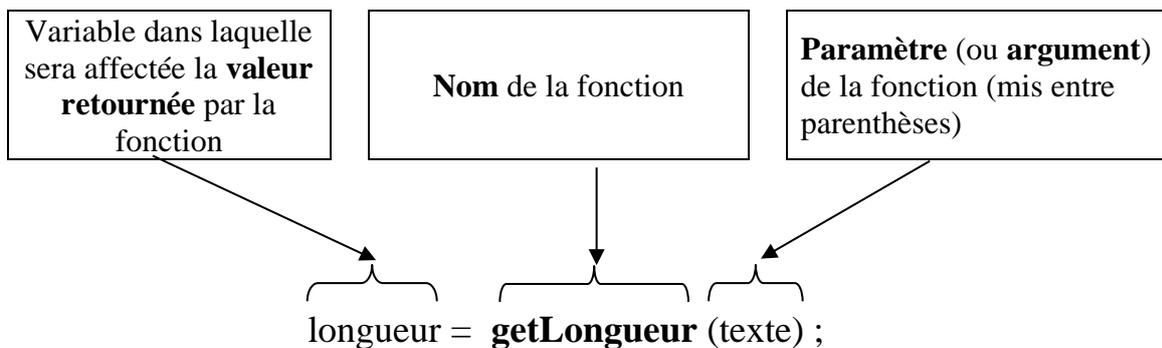
#### Exemple 2 :

```
int nb1, nb2, lemax ;
nb1 = 5;
nb2 = 50;
lemax = max(nb1, nb2);
System.out.println("le plus grand est : " + lemax);
```

	nb1	nb2	lemax
int nb1, nb2, lemax ;	<input type="text"/>		
nb1 = 5;	<input type="text"/>	<input type="text"/>	
nb2 = 50;	<input type="text"/>	<input type="text"/>	
lemax = <b>max</b> (nb1, nb2);	<input type="text"/>	<input type="text"/>	<input type="text"/>

Qu'est-ce qui s'affiche ? . . . . .

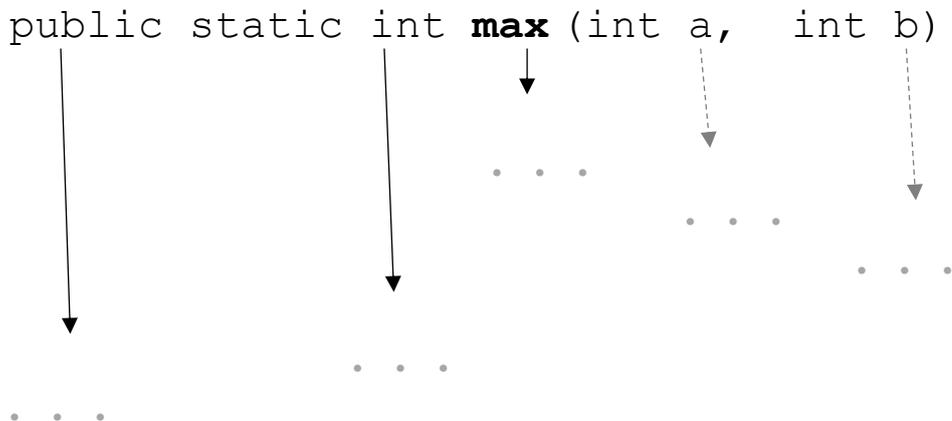
#### Pour résumer :



### III. Signature d'une fonction

La **signature d'une fonction** exprime en une ligne quel est son **contrat**, c'est-à-dire ses **conditions d'utilisation**.

Par exemple, la documentation de Java donne la signature de la fonction **max** de la classe **Math** :



**Description** : Retourne la plus grande des deux valeurs entières (int) a et b.

**Exercice 2** : compléter le programme suivant qui demande de saisir deux entiers et qui affiche le plus grand, obtenu avec la fonction **max**.

```
Scanner sc = new Scanner(System.in);

System.out.println("veuillez saisir deux nombres");

int nb1 = sc.nextInt();
int nb2 = sc.nextInt();

int lemax;           // le max

// obtenir le plus grand en utilisant Math.max

. . .
```

**Exercice 3** : compléter le programme suivant qui permet d'obtenir le nombre le plus petit de deux nombres saisis. Utilisez la fonction **Math.min** dont voici la signature :

```
public static int min (int a, int b)

Description : Retourne la plus petite des deux valeurs entières (int) a et b.

// programme
Scanner sc = new Scanner(System.in);
System.out.println("veuillez saisir deux nombres");
int nb1 = sc.nextInt();
int nb2 = sc.nextInt();

. . .
. . .
```

## SIO 2            BLOC 2 – Ch. 4 – Révisions - Les fonctions

### Exercice 4 : Simuler un dé à 6 faces



**Objectif :** créer un programme qui génère un nombre aléatoire compris entre 1 et 6.

```
// obtenir un nombre au hasard compris entre 0 et 1
// à mettre dans variable nbHasard
. . .

// multiplier ce nombre par 6
. . .

// arrondir ce nombre à l'entier inférieur
// avec la fonction floor
// et ajouter +1
. . .

// affichage
System.out.println( . . . );
```

#### **Documentation des fonctions :**

```
public static double random()
```

Retourne une valeur (double) positive, comprise entre 0.0 (inclus) et 1.0 (exclu). La valeur est générée aléatoirement.

```
public static double floor (double a)
```

Retourne le nombre entier le plus grand, qui est inférieur ou égal à l'argument.

Paramètre : a, une valeur

#### IV. Ecriture d'une fonction

Le développeur peut aussi écrire ses propres fonctions qui seront susceptibles d'être réutilisées.

Intérêts : gain de temps et meilleure lisibilité du programme principal.

```

.....
// programme principal                                note1    note2    moy
Scanner sc = new Scanner(System.in);
System.out.println("Saisissez deux notes");
double note1 = sc.nextDouble();                      
double note2 = sc.nextDouble();                      
double moy ;
moy = calculerMoyenne(note1, note2);                 
System.out.println("la moyenne est " + moy);

```

```

.....
// fonction écrite par le développeur                nb1    nb2    moyenne
public double calculerMoyenne(double nb1, double nb2)  
{
    double moyenne ;
    moyenne = (nb1 + nb2) / 2;                       
    return moyenne;
}

```

Qu'est-ce qui s'affiche ? . . . . .

Remarque sur la portée des variables :

. . . . .