

Les commandes Scilab sont écrites en **violet**. Dans les exemples, toute ligne précédée de **-->** est une commande, les autres lignes sont les retours.

Entrées

La phase 1 d'initialisation s'appelle aussi phase d'affectation des données.

On distingue :

- les données constantes, qui sont des nombres, ou du texte, ou Vrai/Faux
- les données variables, qui sont comme des boîtes dont le contenu peut être modifié en cours de programme, et qui sont nommées par des lettres

Pour donner à x la valeur 126, je peux dire :

- Mettre 126 dans x
En Scilab : **x=126**
- Lire x : ce qui permet de ne pas fixer x à priori.
En Scilab : **x=input(" Valeur de x = ");**
On affiche une phrase (toujours écrite entre guillemets) et le programme attend la valeur donnée pour continuer.

Si je veux ensuite rajouter 5 à x, qui a déjà une valeur, je dirai :

- x prend la valeur x+5
En Scilab : **x=x+5**

Sorties

La phase 3 d'affichage peut être plus ou moins sophistiquée : supposons qu'après calcul la valeur à afficher s'appelle y :

- Ecrire le nom de la variable fait afficher sa valeur, sauf si on met un point virgule ;
Si la constante calculée n'a pas de nom, elle s'appellera par défaut **ans** (answer = réponse)

```
-->y=7
```

```
y =  
7.
```

```
-->y=7;
```

Rien ne s'affiche

```
-->7+3
```

```
ans =  
10.
```

Pour afficher deux valeurs sur la même ligne, utiliser les crochets, avec une virgule :

```
-->a=4;b=6.2;
```

```
-->[a,b]
```

```
ans =  
4.    6.2
```

- La commande **afficher** permet des affichages plus jolis, combinant des phrases (chaînes de caractères), mises entre guillemets, et des valeurs changées en chaînes de caractère grâce à la commande **string**, avec des **+** entre les deux :

```
-->afficher("La valeur de y est "+string(y))
```

```
La valeur de y est 7
```

Les deux méthodes d'affectation des données vues plus haut ont un inconvénient :
On doit recommencer l'exécution du programme à chaque fois que la variable change.

Il est pratique de définir une fonction, qu'on pourra appeler f , et qui transformera x en y .

Il suffira ensuite de demander $f(x)$ pour obtenir y .

La fonction peut très bien dépendre de plusieurs variables.

Elle peut aussi donner plusieurs valeurs réelles comme image. On les écrira alors entre crochets, séparées par des virgules (voir exemple 2).

En Scilab :

```
function y=f(x)
phase 2 : calcul de y ;
endfunction
```

Exemples et exercices

Exemple 1 : Calculer le prix TTC d'un article dont on connaît le prix HT, avec un taux de TVA de 19,6 %.

Algorithme

- Lire le prix hors taxe, le mettre dans HT
- Calculer le prix toutes taxes comprise et le mettre dans TTC
- Afficher TTC

Scilab : Calcul et affichages simples

```
-->HT=540;
-->TTC=HT*1.196
TTC =
    645.84
```

Scilab : Demande de valeur et affichage plus joli

```
-->HT=input("Prix hors taxe : ");
Prix hors taxe : 3789
-->TTC=HT*1.196;
-->afficher("Le prix toutes taxes comprises est "+string(TTC))
Le prix toutes taxes comprises est 4531.644
```

Scilab : Fonction

```
-->function TTC=f(HT);
-->TTC=HT*1.196;
-->endfunction
-->f(540)
ans =
    645.84
-->f(3789)
ans =
    4531.644
```

Exemple 2 : Transformer un temps donné en secondes en heures, minutes secondes.

Algorithme

Lire le temps en secondes, le mettre dans t
Mettre dans q le quotient de t par 60 (nombre de minutes)
Mettre dans s le reste (nombre de secondes)
Mettre dans h le quotient de q par 60 (nombre d'heures)
Mettre dans m le reste (minutes restantes)
Afficher heures, minutes, secondes

Scilab : Calcul et affichages simples

```
-->t=12680;  
-->q=quotient(t,60);s=reste(t,60);  
-->h=quotient(q,60);m=reste(q,60);  
-->[h,m,s]  
ans =  
    3.    31.    20.
```

*Quotient et reste sont des
fonctions du module lycee.*

Scilab : Demande de valeur et affichage plus joli

```
-->t=input("Heure en secondes : ");  
Heure en secondes : 4586  
-->q=quotient(t,60);s=reste(t,60);  
-->h=quotient(q,60);m=reste(q,60);  
-->afficher(string(h)+" heures "+string(m)+" minutes "+string(s)+"  
secondes ")  
  
1 heures 16 minutes 26 secondes
```

Scilab : Fonction

```
-->function y=f(t)  
-->  q=quotient(t,60);s=reste(t,60);  
-->  h=quotient(q,60);m=reste(q,60);  
-->  y=[h,m,s]  
-->endfunction  
-->f(12680)  
ans =  
    3.    31.    20.  
-->f(4586)  
ans =  
    1.    16.    26.
```

Exercice 1

Utiliser les trois méthodes pour transformer des euros en dollars (taux de change 1 € = 1,4 \$)

Exercice 2

Utiliser les trois méthodes pour calculer l'hypoténuse d'un triangle rectangle dont on connaît les deux côtés de l'angle droit.