



# PROGRAMMER EN PYTHON

## PYZO : Le SHELL



Pour effacer le contenu du shell

Pour redémarrer le shell

Pour terminer le programme

Ne pas cliquer sur cet icône (fermeture du shell)

Shells

Python [clear] [restart] [stop] [close] [undo] [redo] [run] [debug] [exit] [menu]

```
Python 3.7.6 (default, Jan 8 2020, 20:23:39) on Windows (64 b
its).
This is the Pyzo interpreter with integrated event loop for AS
YNCIO.
Type 'help' for help, type '?' for a list of *magic* commands.

>>>
```

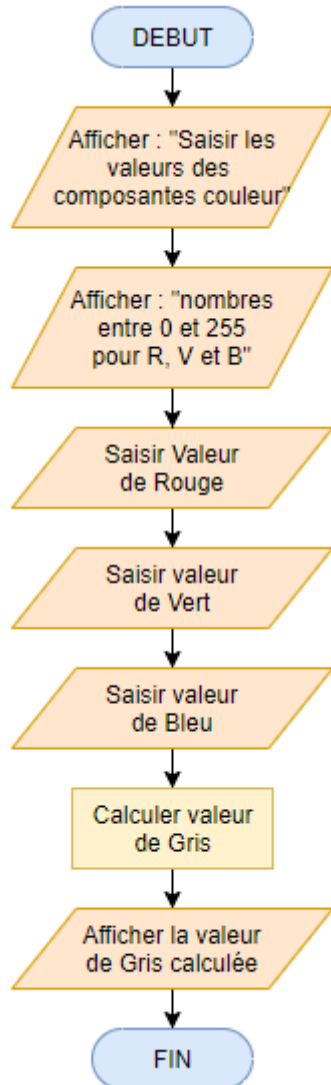
C'est à droite de ces trois signes « > » que vous saisissez les valeurs et que s'affichent les résultats



# PROGRAMMER EN PYTHON



## Exercice N°1



Chaque point d'une image couleur est codée en RVB (mélange des trois couleurs Rouge, Vert, Bleu). Pour convertir ce point en une nuance de gris, on utilise la formule suivante :

$$G = \text{int}(0.11 * R + 0.83 * V + 0.06 * B)$$

Écrire le programme qui permet de saisir les valeurs rouge, vert et bleu (nombres entiers compris entre 0 et 255) d'un pixel couleur, de convertir ce code couleur en nuance de gris puis d'afficher la valeur de Gris dans le Shell.



# PROGRAMMER EN PYTHON

## Exercice N°1

Il faut utiliser les instructions suivantes :

Saisir une valeur depuis le clavier

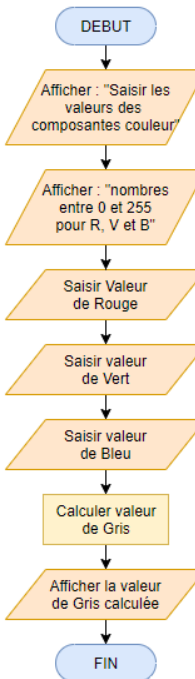
```
toto = int(input("message affiché"))
```

Travailler sur des nombres entiers

```
int( )
```

Afficher le résultat

```
print("message affiché" , toto)
```





# PROGRAMMER EN PYTHON

## Exercice N°1



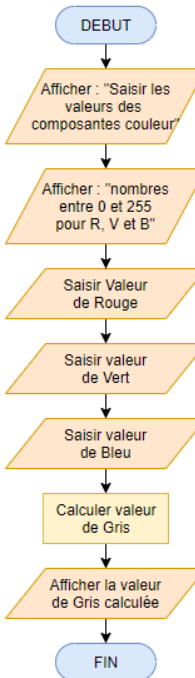
```
Rouge = int(input("Rouge (R) = "))
```

Variable dans laquelle sera stockée le nombre correspondant à la valeur de rouge saisie au clavier

Instruction qui permet de saisir une valeur depuis le clavier

Instruction qui mémorise la valeur saisie dans une variable de type nombre entier

Texte qui sera affiché à chaque fois. Il est encadré par des guillemets



Pour saisir une valeur au clavier, il faut placer le curseur dans le shell

>>>



# PROGRAMMER EN PYTHON

## Exercice N°1



```
Rouge = int(input("Rouge (R) = "))
```

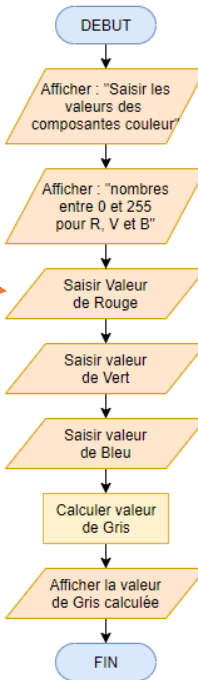
Appuyez sur la touche **F5** pour exécuter le programme

Placez le curseur dans le shell

Saisissez une valeur numérique en 0 et 255 puis validez par ENTRÉE

Si aucune erreur, cette étape est franchie avec succès

Si PYZO indique une erreur, revoyez le code de votre programme !





DEBUT

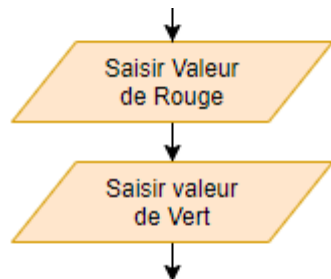
# PROGRAMMER EN PYTHON

## Exercice N°1



```
Rouge = int(input("Rouge (R) = "))
```

```
Vert =
```



FIN



DEBUT

# PROGRAMMER EN PYTHON

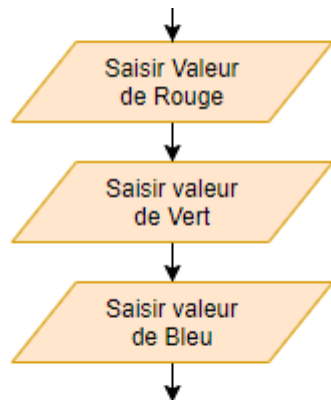
## Exercice N°1



**Rouge** = **int**(**input**("Rouge (R) = "))

**Vert** =

**Bleu** =



FIN



DEBUT

# PROGRAMMER EN PYTHON

## Exercice N°1

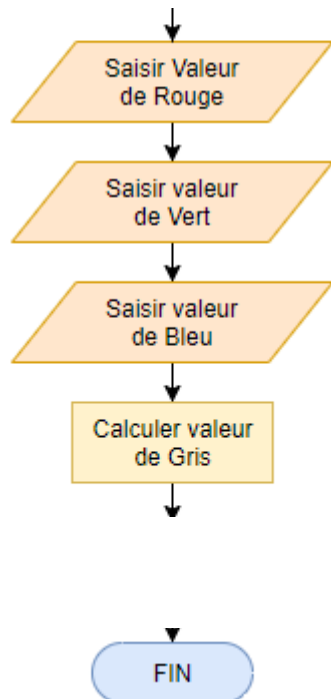


**Rouge** = `int(input("Rouge (R) = "))`

**Vert** =

**Bleu** =

**Gris** =







DEBUT

# PROGRAMMER EN PYTHON

## Exercice N°1



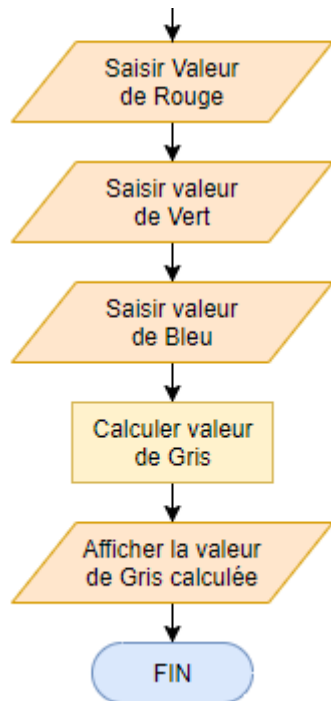
**Rouge** = **int**(**input**("Rouge (R) = "))

**Vert** =

**Bleu** =

**Gris** =

**print**("\_\_\_\_\_", \_\_\_\_\_)





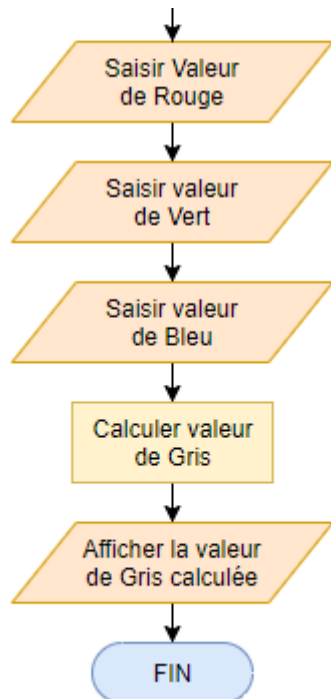
DEBUT

# PROGRAMMER EN PYTHON



## Exercice N°1

```
Rouge = int(input("Rouge (R) = "))  
Vert =  
Bleu =  
Gris =  
print (_____)
```



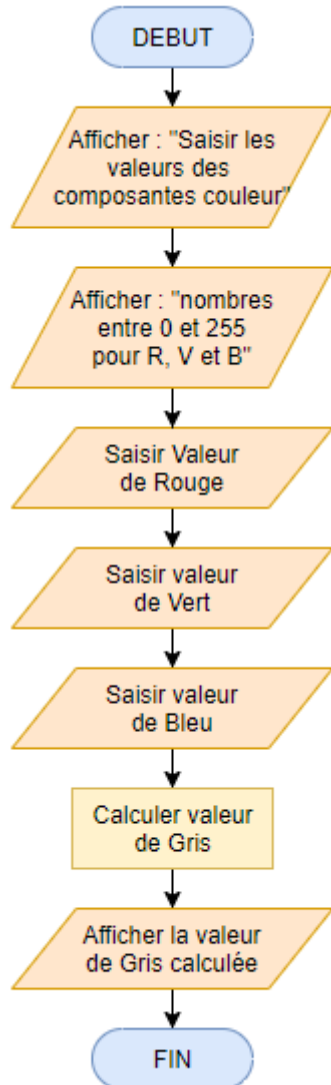
Après exécution du programme, on comprend que l'information n'est pas suffisante pour que l'utilisateur comprenne ce que réalise le programme ! Il faut afficher plus d'informations au début du programme...



# PROGRAMMER EN PYTHON



## Exercice N°1



```
print("Ce programme permet de convertir")
print("une couleur RVB en nuance de gris.")
print("Saisir la valeur des composantes couleur")
print("soit 3 nombres entre compris 0 et 255")
Rouge = int(input("Valeur de la composante rouge : "))
Vert =
Bleu =
Gris =
print(_____)
```

L'information est maintenant suffisante !