PYTHON diapo 9

Fonctions

Roblet [2]

dernière MAJ le 06/11/19

Diapo 9 Fonctions

Dans Python, une **fonction** est comme un sous-programme pour lequel on rentre 0, un ou plusieurs paramètres et qui peut faire une action, ou retourner une ou plusieurs valeurs.

Diapo 9 Fonctions

CE QU'IL FAUT SAVOIR

Python	syntaxe

`def fonction_somme(x1,x2):`
`  somme=x1+x2`
`  return somme`
	`def` nom de la fonction `(` paramètre 1 `,` paramètre 2 `,` ... `):`
	instruction(s)
	`return` ce que retourne la fonction

Remarques

- Concernant le nom de la fonction, on n'a pas le droit aux espaces :

`nomdelafonction` OUI `nom_de_la_fonction` OUI `nom de la fonction` NON

- La commande `return` stoppe l'exécution de la fonction. Toutes les lignes de codes qui suivent avec l'indentation ne sont pas lues.

- Toutes les variables créées dans le corps de la fonction sont des variables locales.

```
y=1
def f(x):
    x=2
    y=0
    return x,y
```

dans la console

```
>>> f(5)
(2, 0)
>>> y
1
```

PRINT ou RETURN ? 1/2

Au premier abord, ces 2 programmes semblent donner la même chose :

```
def fonction_avec_return(x):
    x=x+1
    return x
```

dans la console

```
>>> fonction_avec_return(4)
5
```

```
def fonction_avec_print(x):
    x=x+1
    print(x)
```

dans la console

```
>>> fonction_avec_print(4)
5
```

Cependant...

dans la console

```
>>> 10*fonction_avec_return(4)
50
```

dans la console

```
>>> 10*fonction_avec_print(4)
```
MESSAGE D'ERREUR

En effet, avec « print », ce qui est affiché n'est pas considéré comme un nombre par Python, contrairement avec « return ».

PRINT ou RETURN ? 2/2

CONCLUSION

Si on doit utiliser la sortie d'un sous-programme dans de nouveaux calculs, il faut donc utiliser « return », mais si ce n'est pas le cas, on peut aussi utiliser « print ».

Pour aller plus loin, on peut demander à Python le type de ce qui est affiché :

dans la console

```
>>> type(fonction_avec_return(4))
<class 'int'>
```

Dans ce cas Python considère « 5 » comme un nombre entier (« integer »).

dans la console

```
>>> type(fonction_avec_print(4))
5
<class 'NoneType'>
```

Dans ce cas Python considère « 5 » comme un objet sans étiquette.

Diapo 9 Fonctions

Que va afficher la console dans chaque cas ?

```
def f(x):
    y=3*x+1
    return y
```

dans la console

```
>>> f(2)
...
```

dans la console

```
>>> f(0)
...
```

dans la console

```
>>> f(0.)
...
```

Diapo 9 Fonctions

Que va afficher la console dans chaque cas ?

```
def f(x):
    y=3*x+1
    return y
```

dans la console

```
>>> f(2)
7
```

dans la console

```
>>> f(0)
1
```

dans la console

```
>>> f(0.)
1.0
```

réponse

« 0. » est considéré comme un réel, il en ressort donc un réel (« 1. »).

calculs effectués

$f(2) = 3 \times 2 + 1$

$f(2) = 6 + 1$

$\boxed{f(2) = 7}$

$f(0) = 3 \times 0 + 1$

$f(0) = 0 + 1$

$\boxed{f(0) = 1}$

Diapo 9 Fonctions

Que va afficher la console dans chaque cas ?

```
def f(x,y,z):
    x=x+1
    y=2
    z=x+y
    return x,y,z
```

dans la console

```
>>> f(0,1,2)
...
```

```
def f(x,y,z):
    z=x+y
    x=x+1
    y=2
    return x,y,z
```

dans la console

```
>>> f(0,1,2)
...
```

Diapo 9 Fonctions

Que va afficher la console dans chaque cas ?

```
def f(x,y,z):
    x=x+1
    y=2
    z=x+y
    return x,y,z
```

dans la console

```
>>> f(0,1,2)
(1, 2, 3)
```

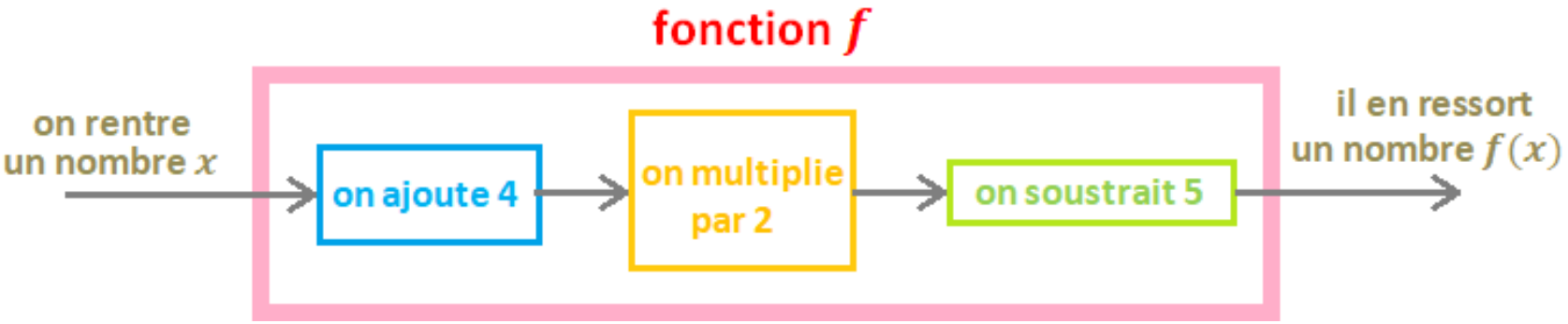
```
def f(x,y,z):
    z=x+y
    x=x+1
    y=2
    return x,y,z
```

dans la console

```
>>> f(0,1,2)
(1, 2, 1)
```

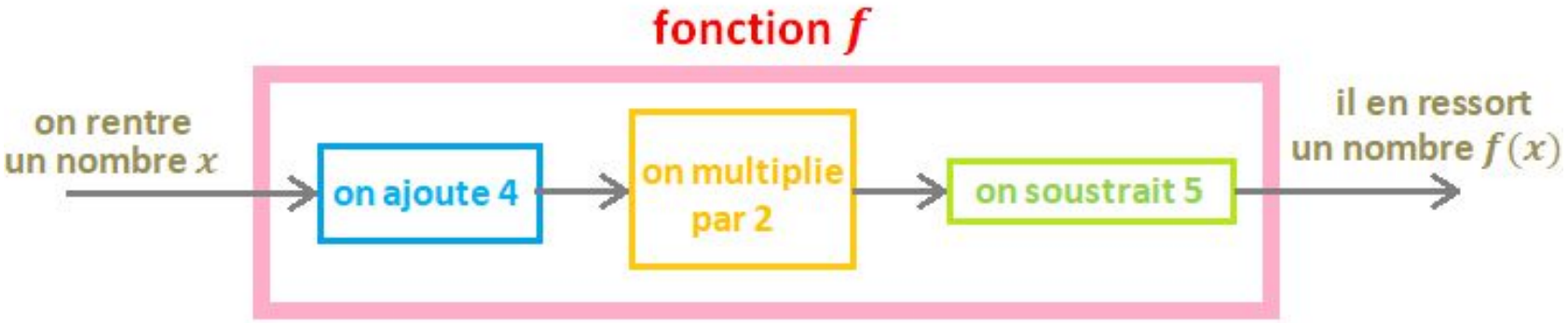
Diapo 9 Fonctions

Créer sous Python la fonction f schématisée ainsi :



Diapo 9 Fonctions

Créer sous Python la fonction f schématisée ainsi :



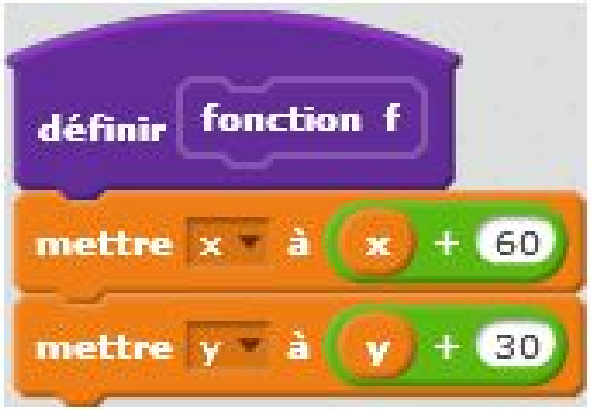
réponses possibles

```
def f(x):
    x=x+4
    x=x*2
    x=x-5
    return x
```

```
def f(x):
    x=(x+4)*2-5
    return x
```

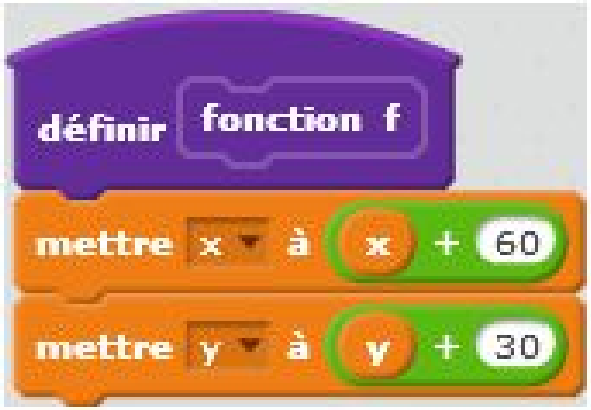
Diapo 9 Fonctions

Traduire le bloc « fonction f » suivant de Scratch à Python :



Diapo 9 Fonctions

Traduire le bloc « fonction f » suivant de Scratch à Python :



réponse

```
def f(x,y):
    x=x+60
    y=y+30
    return x,y
```

Diapo 9 Fonctions

Créer sous Python une fonction « vitesse » de paramètres « d » (pour « distance ») et « t » (pour « temps ») qui renvoie la vitesse en km/h lorsque l'on rentre une distance en km et un temps en h.

Diapo 9 Fonctions

Créer sous Python une fonction « vitesse » de paramètres « d » (pour « distance ») et « t » (pour « temps ») qui renvoie la vitesse en km/h lorsque l'on rentre une distance en km et un temps en h.

réponse

```
def vitesse(d,t):
    v=d/t
    return v
```

Modifier ce programme pour qu'il renvoie la vitesse en m/s lorsque l'on rentre une distance en km et un temps en h.

Diapo 9 Fonctions

Créer sous Python une fonction « vitesse » de paramètres « d » (pour « distance ») et « t » (pour « temps ») qui renvoie la vitesse en km/h lorsque l'on rentre une distance en km et un temps en h.

réponse

```
def vitesse(d,t):
    v=d/t
    return v
```

Modifier ce programme pour qu'il renvoie la vitesse en m/s lorsque l'on rentre une distance en km et un temps en h.

réponse

```
def vitesse(d,t):
    d=d*1000
    t=t*3600
    v=d/t
    return v
```