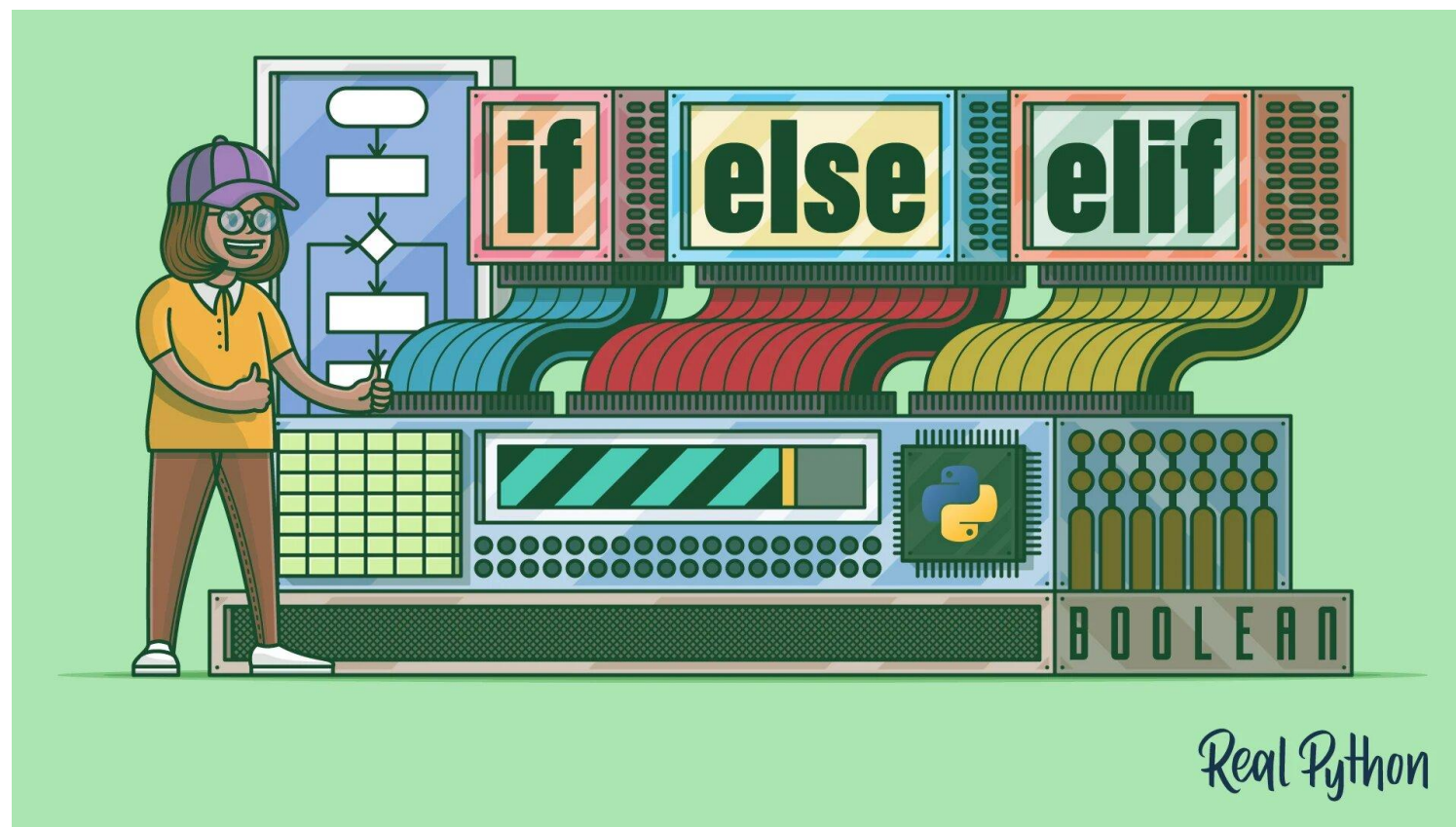




La Plateforme

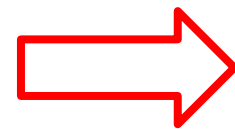
/ Les conditions et les fonctions



La Plateforme

/ Qu'est-ce qu'une instruction conditionnelle ?


Instruction if



```
name = "Alex"

if name == "Alex":
    print("Ah oui ! Je me souviens de toi !")
```

Instruction if...else



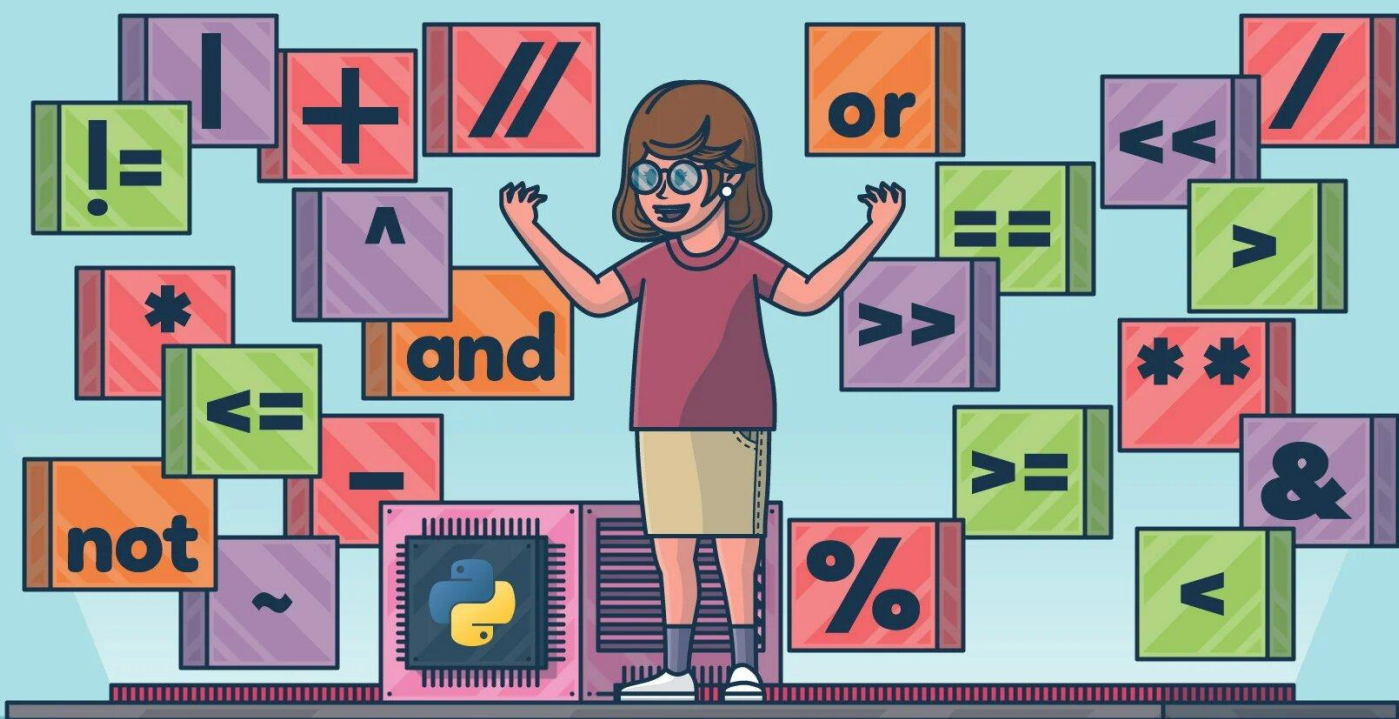
```
name = "Jim"

if name = "Alex":
    print('Ah oui ! Je me souviens de toi')
else:
    print('Qui es tu?')
```

Instruction if...elif...else

```
name = "Alex"

if name == "Alex":
    print("Bonjour Alex")
elif name == "Julie":
    print("Salut Julie")
else:
    print("Qui es-tu?")
```



Real Python

La Plateforme

/ Les opérateurs logiques et les opérateurs de comparaison

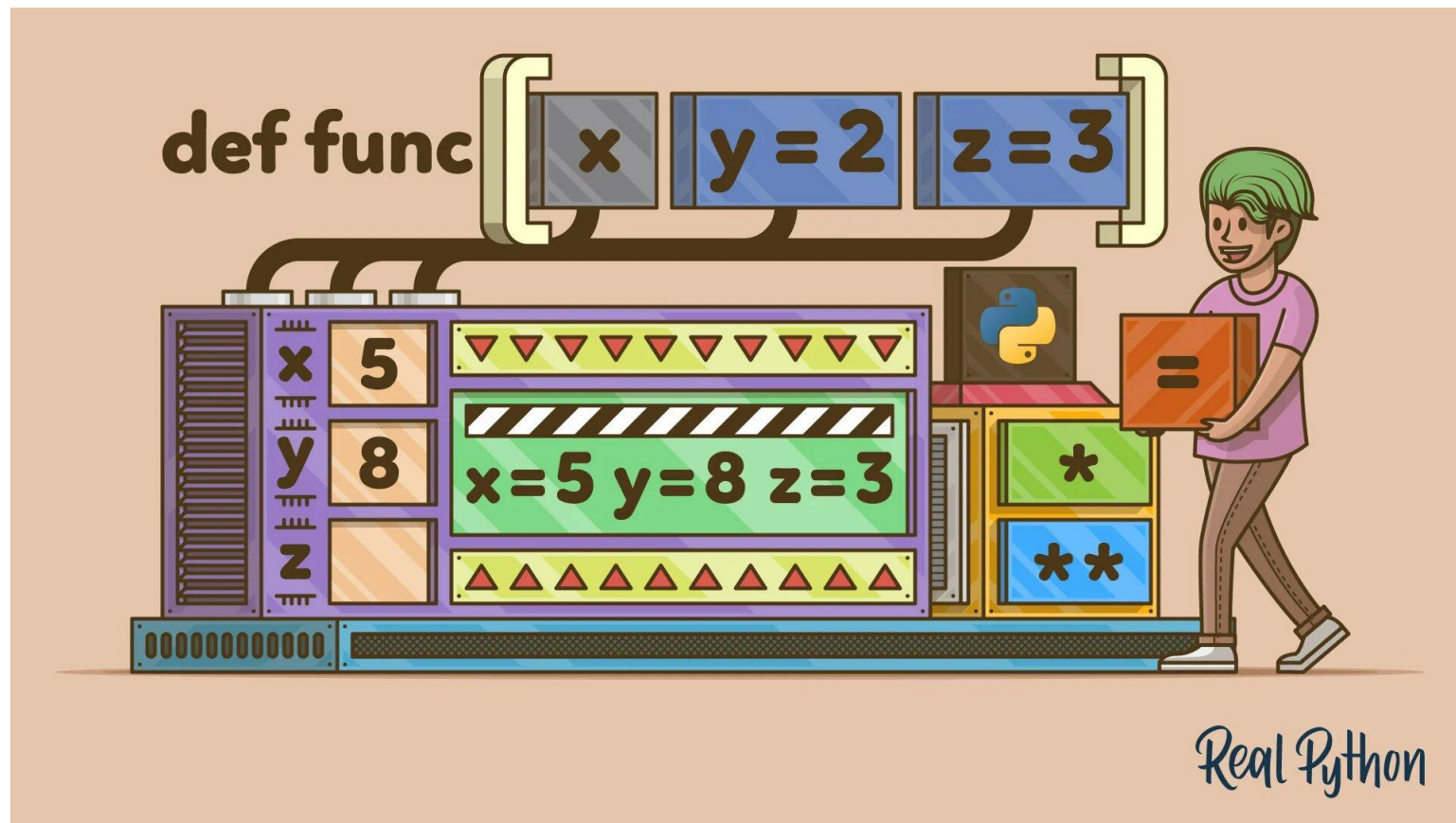
Opérateurs logiques**Définition**

and	vérifie si deux conditions sont toutes les deux vraies.
or	vérifie si au moins une condition est vraie.
not	vérifie si une condition n'est pas vraie (c'est-à-dire fausse).

```
chiffre = 2.5
if (chiffre == 2.5 and chiffre <10):
    print("top")
```

Opérateurs de comparaison**Définition**

$a == b$	Égal
$a != b$	Non égal
$a > b$	Plus que
$a < b$	Moins que
$a >= b$	Plus que ou égal
$a <= b$	Moins que ou égal



La Plateforme

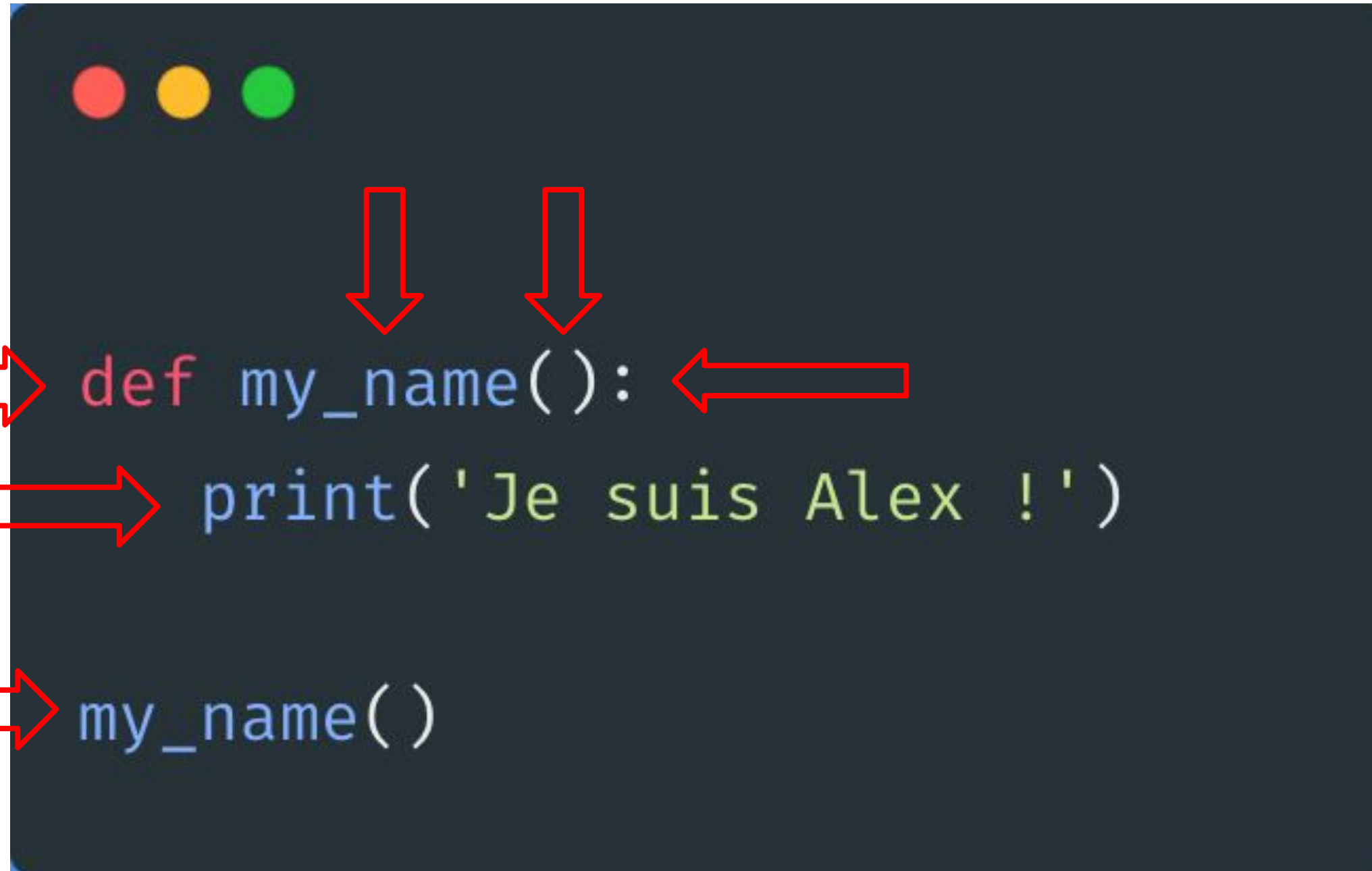
/ Qu'est-ce qu'une fonction ? Et quel est son rôle

Déclarer la fonction `my_name`

mot clé

indentation

appelle/exécute
la fonction



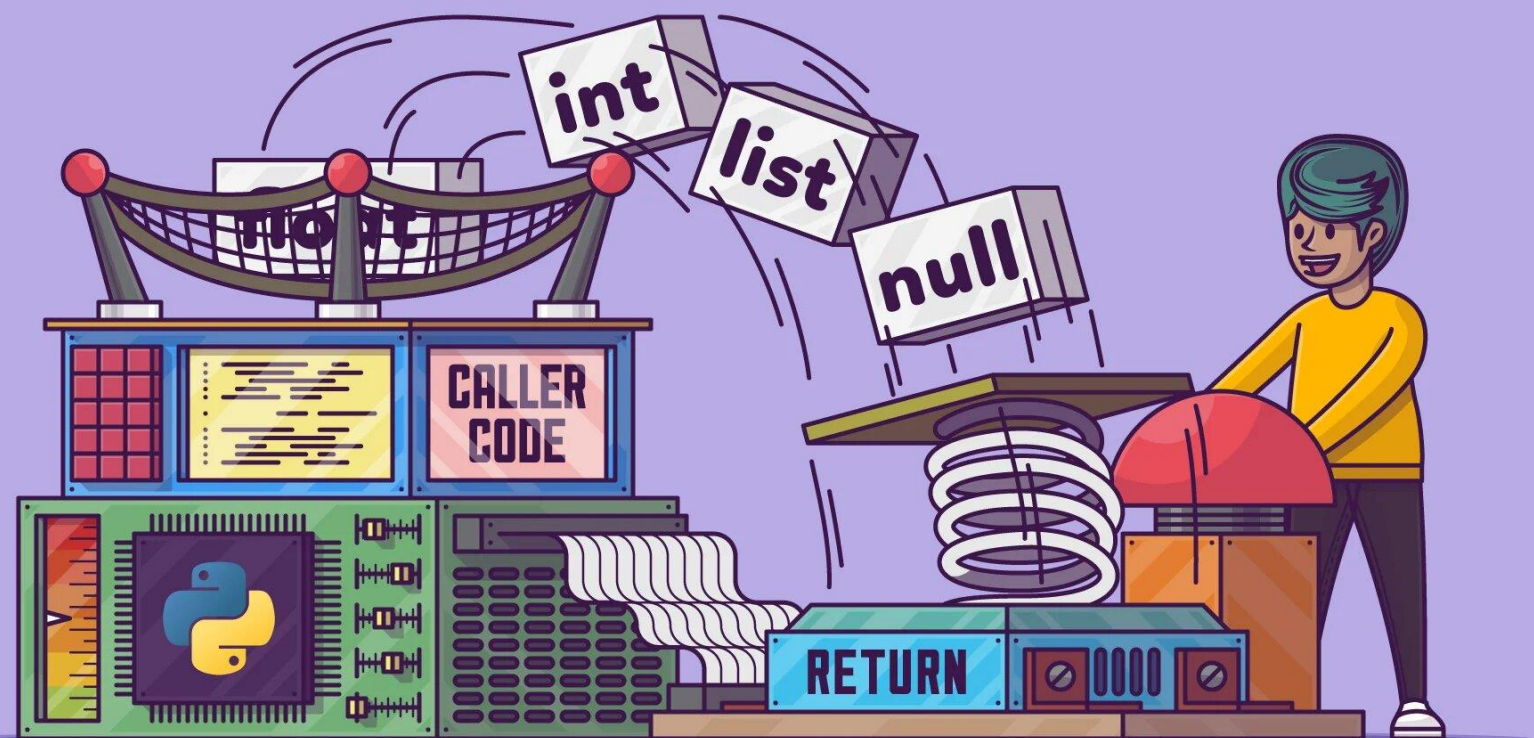
```
def my_name():  
    print('Je suis Alex !')  
  
my_name()
```

The image shows a code editor window with three colored window control buttons (red, yellow, green) at the top left. The code inside the editor is as follows:

```
def my_name():  
    print('Je suis Alex !')  
  
my_name()
```

Red arrows point from the labels on the left to the corresponding parts of the code:

- A red arrow points from "mot clé" to the `def` keyword.
- A red arrow points from "indentation" to the `print` statement, which is indented under the `def` statement.
- A red arrow points from "appelle/exécute la fonction" to the `my_name()` function call.



Real Python

La Plateforme

/ À quoi sert le return?

Le return dans une fonction

```
def getHello():  
    return 'Hello tout le monde'  
  
message = getHello()  
print(message)
```





/ Questions