

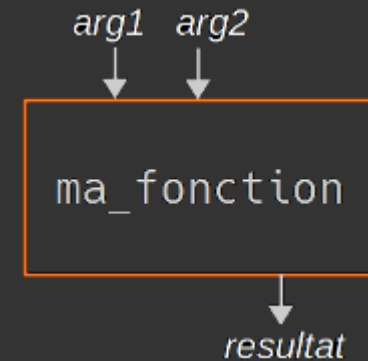
4. Fonctions

4. Fonctions

4.1 Principe

Donner un nom à un ensemble d'instructions (modularité, sémantique)

```
def ma_fonction(arg1, arg2):  
    instruction1  
    instruction2  
    ...  
    return resultat
```

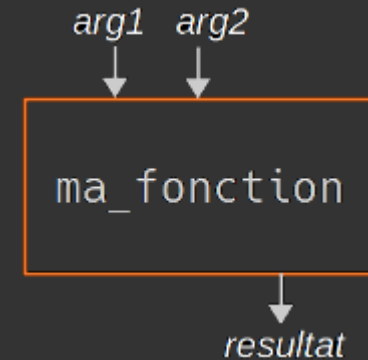


On peut ensuite utiliser la fonction et récupérer le resultat :

```
mon_resultat = ma_fonction("pikachu", "bulbizarre")  
autre_resultat = ma_fonction("salameche", "roucoups")
```

4. Fonctions

```
def ma_fonction(arg1, arg2):  
    instruction1  
    instruction2  
    ...  
    return resultat
```



4.2 Elements de syntaxe

- **def**, **:**
- indentation !!
- Des arguments
- **return**

4. Fonctions

4.3 Exemple

```
def distance(dx, dy):  
    dx_carre = dx ** 2  
    dy_carre = dy ** 2  
    return math.sqrt(dx_carre + dy_carre)
```

Utilisation :

```
D = distance(3, 5)           # -> D vaut 5.830951894845301
```

4. Fonctions

4.3 Exemple

```
def distance(dx, dy):  
    dx_carre = dx ** 2  
    dy_carre = dy ** 2  
    print(math.sqrt(dx_carre + dy_carre))
```

Utilisation :

```
D = distance(3, 5)          # -> D vaut None !
```

4. Fonctions

4.4 Variables locales

```
def distance(dx, dy):  
    dx_carre = dx ** 2  
    dy_carre = dy ** 2  
    return math.sqrt(dx_carre + dy_carre)
```

- Les variables créées dans la fonction sont **locales**
- Les noms dx, dy sont également "locaux" : ils font référence aux arguments de cette fonction, même si d'autres variables nommées dx ou dy existent ailleurs dans le code !

4. Fonctions

4.5 return

- `return` **quitte immédiatement la fonction !**
- Il peut néanmoins y avoir plusieurs `return` dans une fonction (mais seul un sera exécuté)
- Une fonction sans `return` renvoie implicitement `None`.

4. Fonctions

4.6 Arguments optionnels

Les arguments peuvent être rendu optionnels si ils ont une valeur par défaut :

```
def distance(dx, dy=0, dz=0):  
    [...]
```

Dans ce cas, tous ces appels sont valides :

```
distance(1)  
distance(1, 2)  
distance(1, 2, 3)  
distance(1, dy=2)  
distance(1, dz=3)  
distance(1, dy=2, dz=3)
```


4. Fonctions

4.7 Différence avec le concept mathématique

- Domaine généralement mal défini (e.g. type d'entrée et de sortie)
- Effets de bords (e.g. `print()`)

