

Exercice 1 (Fonction sans argument). Vérifier que le code :

```
for i in range(16) :
    print("*",end="")
print()

print("* Hello world! *")

for i in range(16) :
    print("*",end="")
print()
```

peut être remplacé par le code :

```
def ligneEtoilee() :
    for i in range(16) :
        print("*",end="")
    print()

ligneEtoilee()
print("* Hello world! *")
ligneEtoilee()
```

L'instruction **def** définit une fonction, c'est à dire un bloc de code réutilisable en l'appelant par son nom. Cela permet d'éviter le copier coller de code, le pire ennemi du programmeur.

Exercice 2. Modifier le code ci-dessous en utilisant une fonction pour éviter le copier coller de code :

```
for i in range(5) :
    print("/\  ", end="")
print()
for i in range(5) :
    print("  \/ ", end="")
print()
print("      Une vague.")
print()
for i in range(4) :
    for i in range(5) :
        print("/\  ", end="")
    print()
    for i in range(5) :
        print("  \/ ", end="")
    print()
print("      La mer.")
```

Exercice 3 (Utilisation d'une fonction prédéfinie avec un argument). 1. Tester ce programme afin de comprendre la fonction **len**, qui est une fonction déjà définie dans le langage **Python** :

```
print("Saisissez un texte :")
texteSaisi = input()
a = len(texteSaisi)
print(a)
```

2. Que va afficher le programme suivant :

```
a = len("Bonjour") + len("à tous.")
print(3*a)
```

Exercice 4 (Définir une fonction avec un argument). 1. Tester le code suivant :

```
def ligneEtoilee(longueur) :
    for i in range(longueur) :
        print("*", end="")
    print()
```

```
ligneEtoilee(7)
```

2. Définir et tester une fonction nommée **box** qui prend en argument un texte **message** et qui l'affiche dans une boîte d'étoiles. Ainsi le code :

```
box("Coucou !")
```

doit afficher :

```
*****
* Coucou ! *
*****
```

Exercice 5 (Fonction avec un argument et qui renvoie une valeur). Quand on définit une fonction, on peut utiliser l'instruction **return** qui aura le double effet de stopper l'exécution de la fonction et de renvoyer la valeur de l'expression qui la suit.

1. Tester le code :

```
def valeurAbsolue(x) :
    if x >= 0 :
        return x
    else :
        return -x

print("Saisissez un nombre :")
texteSaisi = input()
a = int(texteSaisi)
b = valeurAbsolue(a)
print("Vous avez saisi le nombre :", a)
print("Sa valeur absolue est :", b)
```

2. Peut-on se passer de l'instruction **else** dans ce cas¹ ?
 3. Compléter le code suivant pour que la fonction **somme(n)** renvoie la somme des entiers de 0 à n. Par exemple, **somme(5)** doit valoir 15.

```
def somme(n) :
    <À compléter>

print("Saisissez un nombre entier :")
a = int(input())
b = somme(a)
print("La somme des entiers jusqu'à ", a, " vaut ", b)
```

4. Refaire la question précédente sans boucle **for** ni boucle **while**.

1. En fait ce n'est pas conseillé de s'en passer pour des raisons de lisibilité du code.