

SNT 2^{ème}

Exercice 1

```
# Écrivez un programme qui demande à l'utilisateur de deviner un nombre secret
compris entre 1 et 10.
# Le programme doit afficher "Trop grand !" si le nombre deviné est supérieur au
nombre secret,
# "Trop petit !" si le nombre deviné est inférieur au nombre secret, et "Bravo !"
si le nombre deviné est correct.
```

```
secret = 7
guess = int(input("Devinez le nombre secret : "))
```

```
while _ _ _ _:
    if _ _ _ _:
        print("Trop grand !")
    elif _ _ _ _:
        print("Trop petit !")
    else:
        print("Bravo !")
        break
```

Exercice 2

```
# Écrivez un programme qui demande à l'utilisateur de deviner un mot secret.
# Le programme doit afficher "Félicitations !" si le mot deviné est correct,
# et "Mauvaise réponse !" si le mot deviné est incorrect.
```

```
secret_word = "python"
guess = input("Devinez le mot secret : ")
```

```
while _ _ _ _:
    if _ _ _ _:
        print("Félicitations !")
        break
    else:
        print("Mauvaise réponse !")
        guess = input("Devinez le mot secret : ")
```

Exercice 3

```
# Écrivez une fonction qui calcule et retourne la somme de deux nombres donnés.
# Utilisez cette fonction pour calculer la somme de deux nombres saisis par
l'utilisateur.
```

```
def add_numbers(a, b):
    return _ _ _ _
```

```
num1 = int(input("Entrez le premier nombre : "))
```

```
num2 = int(input("Entrez le deuxième nombre : "))
result = add_numbers(num1, num2)
print("La somme est :", result)
```

Exercice 4

```
# Écrivez un programme qui demande à l'utilisateur de saisir son nom.
# Le programme doit afficher "Bonjour, [nom de l'utilisateur] !" s'il saisit un
nom,
# et "Veuillez saisir un nom !" s'il ne saisit rien.
name = input("Entrez votre nom : ")

while _ _ _ _:
    if _ _ _ _:
        print("Bonjour,", name, "!")
        break
    else:
        print("Veuillez saisir un nom !")
        name = input("Entrez votre nom : ")
```

Exercice 5

```
# Écrivez un programme qui demande à l'utilisateur de saisir un nombre positif.
# Le programme doit afficher la table de multiplication de ce nombre jusqu'à 10.
num = int(input("Entrez un nombre positif : "))
count = 1

while _ _ _ _:
    result = _ _ _ _
    print(num, "x", count, "=", result)
    count += 1
```

Exercice 6

```
# Écrivez une fonction qui calcule et retourne le carré d'un nombre donné.
# Utilisez cette fonction pour afficher les carrés des nombres de 1 à 5.
def square(num):
    return _ _ _ _

count = 1

while _ _ _ _:
    result = _ _ _ _
    print("Le carré de", count, "est", result)
    count += 1
```

Exercice 7

```
# Écrivez un programme qui demande à l'utilisateur de saisir une phrase.
# Le programme doit afficher le nombre de caractères dans la phrase (y compris les
espaces).
sentence = input("Entrez une phrase : ")
count = 0

while _ _ _ _:
    count += _ _ _ _

print("La phrase contient", count, "caractères.")
```

Exercice 8

```
# Écrivez une fonction qui vérifie si une chaîne de caractères donnée est un
palindrome.
# Utilisez cette fonction pour vérifier si une phrase saisie par l'utilisateur est
un palindrome.
def is_palindrome(text):
    _ _ _ _

text = input("Entrez une phrase : ")

while _ _ _ _:
    if _ _ _ _:
        print("C'est un palindrome !")
        break
    else:
        print("Ce n'est pas un palindrome !")
        text = input("Entrez une phrase : ")
```

Exercice 9

```
# Écrivez un programme qui demande à l'utilisateur de saisir un nombre positif.
# Le programme doit afficher la somme de tous les nombres de 1 jusqu'au nombre
saisi.
num = int(input("Entrez un nombre positif : "))
sum = 0
count = 1

while _ _ _ _:
    sum += _ _ _ _
    count += 1

print("La somme des nombres de 1 à", num, "est", sum)
```

Exercice 10

```
# Écrivez une fonction qui renverse une chaîne de caractères donnée.  
# Utilisez cette fonction pour inverser une phrase saisie par l'utilisateur.  
def reverse_string(text):  
    _ _ _ _  
  
text = input("Entrez une phrase : ")  
reversed_text = _ _ _ _  
  
print("La phrase inversée est :", reversed_text)
```