

Atelier_1_Corr

N'oubliez pas d'utiliser des commentaires pour expliquer le fonctionnement de chaque partie de votre code.

Activité 1

Trouver et corriger l'erreur dans chaque ligne de code, **si elle existe**, avec une justification:

- a. message = Hello world
- b. 23message = 'Hello world'
- c. New message = 'Hi there'
- d. print(message.upper)
- e. print(message.lower())
- f. print(message.count())
- g. nom = input(Veuillez entrer votre nom SVP)
- h. a='c'
b=123
d=a+b

Solution

```
message = "Hello world"
23message = 'Hello world'
New_message = 'Hi there'
print(message.upper())
print(message.lower())
print(message.count('l'))
nom = input('Veuillez entrer votre nom SVP')
a='c'
b='123'
d=a+b
```

Activité 2

Ecrire un programme en Python qui demande à l'utilisateur de saisir deux nombres x et y et de lui afficher leur somme : $x + y$

Solution

```
a=int(input("veuillez saisir un nombre x: \n"))
b=int(input("veuillez saisir un nombre y: \n"))
print("La somme de x et y est:", x,'+',y,'=',x+y)
```

Les structures conditionnelles

Activité 1

Ecrire un programme en Python qui demande à l'utilisateur de saisir 3 nombre x, y et z et de lui afficher leur maximum

Solution

```
# Demander à l'utilisateur de taper 3 nombres a, b, c
a = int(input("Type a value of the number a "))
b = int(input("Type a value of the number b "))
c = int(input("Type a value of the number c "))
# définir et initialiser le maximum à zero
max = 0
if(a > b):
    max = a
else:
    max = b
if(max < c):
    max = c
else:
    max = max
print("Le maximum des trois nombre est : max(a,b,c) = ", max)
```

Activité 2

Créez un jeu où l'ordinateur choisit un nombre aléatoire entre 1 et 100, avec `random.randint(1, 100)` et l'utilisateur doit deviner ce nombre en utilisant des propositions. L'ordinateur doit donner des indices tels que "Trop grand" ou "Trop petit" pour aider l'utilisateur à se rapprocher du nombre mystère

Solution

```
import random
nombre_mystere = random.randint(1, 100)
while True:
    proposition = int(input("Devinez le nombre mystère entre 1 et 100 : "))
    if proposition < nombre_mystere:
        print("Trop petit.")
    elif proposition > nombre_mystere:
        print("Trop grand.")
    else:
        print("Bravo, vous avez trouvé le nombre mystère !")
        break
```

Activité 3

Vérification de la parité des chiffres

Demandez à l'utilisateur de saisir un nombre entier positif. Ensuite, vérifiez si tous les chiffres du nombre sont pairs, et affichez le résultat

Solution

```
nombre = int(input("Entrez un nombre entier positif : "))
# Vérifier si tous les chiffres sont pairs
tous_paires = all(int(chiffre) % 2 == 0 for chiffre in str(nombre))
```

```
if tous_paires:
    print("Tous les chiffres du nombre sont pairs.")
else:
    print("Certains chiffres du nombre ne sont pas pairs.")
```

Les boucles

Activité 1

Ecrire un programme en Python qui demande à l'utilisateur de saisir un nombre entier n et de lui afficher la valeur de la somme $1 + 2 + \dots + n =$

Solution

```
print("Donnez un nombre entier :")
n=int(input("n = "))
s=0
for i in range(n+1):
    s=s+i
print("\nLa somme de 1+2+...+",n," est", s)
```

Activité 2

Initialisez deux entiers : a = 0 et b = 10.

Écrire une boucle affichant et incrémentant la valeur de a tant qu'elle reste inférieure à celle de b.

Écrire une autre boucle décrémentant la valeur de b et affichant sa valeur si elle est impaire. Boucler tant que b n'est pas nul.

Solution

```
print("les valeurs de a :")
a=0
b=10
while(a<b):
    print (a)
    a=a+1
print("les valeurs de b :")
while (b>0):
    if (b%2!=0):
        print(b)
    b=b-1
```

```
# ecriture compacte :
plus_petit = x if x < y else y
print("\nLa plus petite des deux est", plus_petit)
print("\nAu revoir")
```

Activité 3

Ecrire un programme en Python qui demande à l'utilisateur de saisir un nombre entier n et de lui afficher n !

Solution

```
# Demander à l'utilisateur de saisir la valeur de l'entier n
n = int(input("Type a value of the integer n "))

# define et initialiser une variable auxiliaire j
j = 1
for i in range(1,n+1):
    j = j*i
print("Factorielle de n est : ",n,"! = : ", j)
```

Activité 4

Ecrire un programme en Python qui demande à l'utilisateur de saisir une chaîne de caractère s et de lui renvoyer un message indiquant si la chaîne contient la lettre 'a' tout en indiquant sa position sur la chaîne. Exemple si l'utilisateur tape la chaîne s = 'langage' le programme lui renvoie : La lettre 'a' se trouve à la position : 1 La lettre 'a' se trouve à la position : 4

Solution

```
# Demander à l'utilisateur de taper la valeur de s
s = input("Tapez la valeur de s : ")
# Obtenir la longueur de la chaîne s
n = len(s)
# Parcourir la chaîne s tout en recherchant le caractère 'a'
for i in range(0,n):
    # Tester si le caractère rencontré est égale à 'a'
    if(s[i] == 'a'):
        print("Le caractère 'a' se trouve à la position : ", i , " dans la chaîne s")
```

Activité 5

Ecrire un programme en Python, qui permet de renvoyer le premier mot d'un texte donné. Exemple pour le texte : t='Python est un merveilleux langage de programmation', le programme doit renvoyer Python

Solution

```
# initialiser le premier mot à une chaîne vide
premierMot = ""
# initialiser le compteur
i = 0
# chercher le premier espace sur la chaîne
while (s[i] != " "):
    premierMot = premierMot + s[i]
    i = i + 1
print("Le premier mot de la chaîne s est : ", s[:i])
```

Activité 6

Ecrire un programme qui demande à l'utilisateur de saisir un texte et de lui renvoyer tous les mots commençant par la lettre b.

Solution

```
# Lire la chaine s
s = input("Tapez une chaine de caractères s : ")
# convertir la chaine s en une liste
s = s.split()
# chercher tous les éléments de la liste qui commencent par la lettre 'a'
for x in s:
    if(x[0] == 'a'):
        print("Le mot : ", x, " commence par la lettre 'a'")
```

Activité 7

Demander à l'utilisateur d'entrer le coût du prêt, le taux d'intérêt et le nombre d'années pour le prêt
Calculez les paiements mensuels avec la formule suivante :

$$M = L [i (1 + i)^n] / [(1 + i)^n - 1]$$

M = paiement mensuel
L = Montant du prêt
i = taux d'intérêt (pour un taux d'intérêt de 5%, i = 0,05)
n = nombre de paiements

Solution

```
#Declare and initialize the variables
monthlyPayment = 0
loanAmount = 0
interestRate = 0
interestOfPayment = 0
numberOfPayments = 0
loanDurationInYears = 0

#Ask the user for the values needed to calculate the monthly payments
strLoanAmount = float(input("How much money will you borrow? "))
strInterestRate = float(input("what is the interest rate on the loan? "))
strLoanDurationInYears = float(input("How many years will it take you to pay off the loan? "))

#Since payments are once per month, number of payments is number of years * 12
numberOfPayments = loanDurationInYears * 12

#Calculate the monthly payment based on the formula
monthlyPayment = loanAmount * interestRate * (1 + interestRate) * numberOfPayments \
/ ((1 + interestRate) * numberOfPayments - 1)

#provide the result to the user
print ("Your monthly payment will be " + str(monthlyPayment))

#Extra credit
print("Your monthly payment will be $%.2f" %monthlyPayment)
```

Activité 8

Ecrire un programme en langage python table_multiplication avec trois variables saisies par l'utilisateur : mul (multiplicateur), bornInf et bornSup.
Ce programme doit afficher la table de multiplication avec les trois paramètres.
Exemple:
Pour mul=5 et bornInf=3 et bornSup=9 le programme doit afficher:

```

3 x 5 = 15
4 x 5 = 20
5 x 5 = 25
6 x 5 = 30
7 x 5 = 35
8 x 5 = 40
9 x 5 = 45

```

Solution

```

print("les valeurs de a :")
a=0
b=10
while(a<b):
    print (a)
    a=a+1
print("les valeurs de b :")
while (b>0):
    if (b%2!=0):
        print(b)
    b=b-1
'''
#Saisie de données
'''mul = int(input("Veuillez saisir la valeur du multiplacateur : "))
bornInf = int(input("Veuillez saisir la valeur de la borne inf : "))
bornSup = int(input("Veuillez saisir la valeur de la borne sup : "))
for i in range(bornInf, bornSup+1):
    print(i, ' * ', mul, ' = ', i*mul)

```

Activité 9

Ecrire Vous devez écrire un programme qui demande à l'utilisateur de saisir une phrase, puis effectue les actions suivantes :

Compte le nombre total de mots dans la phrase.

Compte le nombre de lettres dans chaque mot et affiche le nombre de lettres pour chaque mot.

Trouve le mot le plus long dans la phrase et affiche sa longueur.

Affiche la phrase en inversant l'ordre des mots.

Saisissez une phrase : Bonjour à tous les utilisateurs de Python

Le nombre total de mots dans la phrase est : 7

Le nombre de lettres pour chaque mot :

- Bonjour : 7 lettres
- à : 1 lettre
- tous : 4 lettres
- les : 3 lettres
- utilisateurs : 12 lettres
- de : 2 lettres
- Python : 6 lettres

Le mot le plus long dans la phrase est "utilisateurs" avec une longueur de 12 lettres.

La phrase inversée est : Python de utilisateurs les tous à Bonjour

Solution

```

# Saisie de la phrase
phrase = input("Saisissez une phrase : ")

# Comptage du nombre total de mots dans la phrase
mots = phrase.split()
nombre_mots = len(mots)
print("Le nombre total de mots dans la phrase est :", nombre_mots)

# Comptage du nombre de lettres pour chaque mot
lettres_par_mot = {}
for mot in mots:
    lettres_par_mot[mot] = len(mot)

# Affichage du nombre de lettres pour chaque mot
print("Le nombre de lettres pour chaque mot :")
for mot, nb_lettres in lettres_par_mot.items():
    print("-", mot, ":", nb_lettres, "lettres")

# Recherche du mot le plus long dans la phrase
mot_plus_long = max(mots, key=len)
longueur_mot_plus_long = len(mot_plus_long)
print("Le mot le plus long dans la phrase est", mot_plus_long, "avec une longueur de",
longueur_mot_plus_long, "lettres.")

# Inversion de l'ordre des mots dans la phrase
mots_inverse = mots[::-1]
phrase_inverse = " ".join(mots_inverse)
print("La phrase inversée est :", phrase_inverse)

mot = "Bonjour"
mot_inverse = ""
for i in range(len(mot) - 1, -1, -1):
    mot_inverse += mot[i]

print("Mot :", mot)
print("Mot inversé :", mot_inverse)

```

Activité 10

Vous devez écrire un programme qui demande à l'utilisateur de saisir une liste de nombres entiers séparés par des virgules. Ensuite, votre programme doit effectuer les actions suivantes :

Calculer et afficher la somme de tous les nombres de la liste.

Trouver et afficher le nombre le plus grand de la liste.

Vérifier si la liste est symétrique, c'est-à-dire si les nombres sont les mêmes lorsqu'ils sont lus de gauche à droite et de droite à gauche.

Solution

```

# Saisie de la liste de nombres
liste_nombres = input("Saisissez une liste de nombres entiers séparés par des virgules : ")

# Conversion de la liste en une liste d'entiers

```

```

nombres = [int(nombre.strip()) for nombre in liste_nombres.split(",")]

# Calcul de la somme des nombres dans la liste
somme = sum(nombres)
print("La somme des nombres dans la liste est :", somme)

# Trouver le nombre le plus grand dans la liste
plus_grand = max(nombres)
print("Le nombre le plus grand dans la liste est :", plus_grand)

# Vérification de la symétrie de la liste
est_symetrique = (nombres == nombres[::-1])
if est_symetrique:
    print("La liste est symétrique.")
else:
    print("La liste n'est pas symétrique.")

```

Les dictionnaires

Activité 4

Ecrivez un script en utilisant le dictionnaire qui permet de réaliser un menu avec 4 options :

```

menu['1']="Ajouter Etudiant."
menu['2']="Supprimer Etudiant."
menu['3']="Chercher Etudiant"
menu['4']="Quitter"

```

A chaque fois que l'utilisateur choisi un entier le programme affiche l'option à faire sans la réaliser.

Solution

```

menu = {}
menu['1']="Ajouter Etudiant."
menu['2']="Supprimer Etudiant."
menu['3']="Chercher Etudiant"
menu['4']="Quitter"
options=menu.keys()
for entry in options:
    print (entry, menu[entry])
for entry in options:
    selection=input("Selectionner SVP:")
    if selection == '1':
        print ("ajout")
    elif selection == '2':
        print ("suppression")
    elif selection == '3':
        print ("recherche" )
    elif selection == '4':
        break
    else:
        print ("Option Inconnue!")

```


Bon travail