

RATTRAPAGE Langage Evolué | Nom : Prénom :Groupe :

EXERCICE 1 : QCM (10 points)

Q1 : Que va renvoyer la commande suivante :

```
>>> type ( [(1,2)] )
```

- <class 'list'>
- <class 'tuple'>
- <class 'dict'>
- <class 'int'>

Q2 : Que va renvoyer la commande suivante :

```
>>> type (len( { 1:2 , 4:8 } ))
```

- <class 'list'>
- <class 'tuple'>
- <class 'dict'>
- <class 'int'>

Q3 : Soit :

```
>>> a = [ (1 , 2 , 3, "Math") , { 1:2 } , [ 1 , 2 ] ]
>>> len(a)
```

Que va afficher le code suivant : **len(a)**

Q4 : Que va afficher le code suivant :

```
x = True
y = False
z = True
if x or (y and z):
    print ("OUI")
else:
    print ("NON")
```

Q5 : Indiquez la liste des valeurs qui seront affichées par le code suivant :

```
liste=(1,2,3,(4,5))
for x in range(len(liste)):
    print(x)
```

- 2, 4, 6
- 0,1, 4, 5
- 0, 1, 2, 3
- 0, 1, 4, 5, 6, 7, 8, 9
- 1, 2, 4, 5, 6

Q6 : Indiquez la liste des valeurs qui seront affichées par le code suivant :

```
liste=(1,2,3,(4,5))
for x in range(1, len(liste), 3):
    print(x)
```

Q7 : Indiquez ce que va afficher le code suivant :

```
try:
    print (2+len("Math-STID"))
except:
    print ("erreur")
else:
    print ("OK")
```

Q8 : Indiquez ce que va afficher le code suivant si l'utilisateur tape 14 en réponse à l'instruction « input » :

```
try:
    a = input("Donne une valeur")
except:
    print ("erreur")
else:
    print ("OK")
```

Q9 : Soit une liste initialisée comme suit :

```
>>> Villes = ['Bejaia', 'Alger', 'Oran', 'Jijel']
```

Quelle est la commande python permettant de créer une chaîne « S » composée des éléments de la liste « modules » séparés par le caractère « / ».

- S = join(Villes , "/")
- S = "/" .join(Villes)
- S = Villes + "/"
- S = Villes.split("/")

Q10 : Soit les commandes suivantes :

```
>>> pays = {}
>>> pays['Bejaia']='algérie'
>>> pays['bejaia']='algérie'
>>> pays['Alger']='algérie'
>>> pays['Alger']='Algérie'
>>> print(len(pays))
```

Que va afficher la dernière commande ?

Q11 : Soit les commandes suivantes :

```
>>> eleve = ['amal', 'said', 'mohand', 'amira']
>>> A = eleve
>>> A.append(10)
>>> print(len(A))
```

Que va afficher la dernière commande ?

Q12 : Soit les commandes suivantes :

```
>>> fruits = ['pomme', 'tomate', 'abricot']
>>> print ( fruits[-2] )
```

Que va afficher la dernière commande ?

Q13 : Soit les commandes suivantes :

```
>>> fruits = ['pomme', 'tomate', 'abricot']
>>> print ( fruits[-1][-3] )
```

Que va afficher la dernière commande ?

Q14 : Soit les commandes suivantes :

```
>>> fruits = ['pomme', 'tomate', 'abricot']
>>> print (fruits[0][0:3] )
```

Que va afficher la dernière commande ?

Q15 : Que va afficher le programme suivant :

```
>>> s = "Bejaia,Alger:Oran"
>>> r = s.split()
>>> r.append(10)
>>> print(r[1])
```

Q16 : Que va afficher le programme suivant :

```
>>> def f(x, y=2, z=5):
    return x+y+z
>>> print(f(z="2", x="BAC", y="3" ))
```

Q17 : Que va afficher le programme suivant :

```
>>> def f(x, y=2, z=5):
    return x+y+z
>>> print(f(10))
```

Q18 : Que va afficher le programme suivant :

```
>>> def f(x, y=10, z="14"):
    return x+y+z
>>> print(f("Physique","Math"))
```

Q19 : Que va afficher le programme suivant :

```
>>> equipes = ('JSK', 'MCA', 'MOB', 'JSMB', 'RSK')
>>> f = open('equipes.txt', 'w')
>>> for e in equipes:
    f.write(e+"\n")
>>> f.close()
>>> f = open('equipes.txt', 'r')
>>> texte = f.read().split("\n")
>>> print(texte[2])
```

Q20 : Que va afficher le programme suivant :

```
>>> equipes = ('JSK', 'MCA', 'MOB', 'JSMB', 'RSK')
>>> f = open('equipes.txt', 'w')
>>> for e in equipes:
    f.write(e+"\n")
>>> f.close()
>>> f = open('equipes.txt', 'r')
>>> print(f.read().strip('\n').split("\n"))
```

Nom : Prénom :Groupe :

EXERCICE 2 (5 points) : Ecrire un programme qui lit (**une seule fois**) une valeur représentant un temps exprimé en secondes. Ce programme doit convertir ce temps sous le format HMS (heure : minutes : secondes) et l'afficher.

Exemple d'exécution :

Donnez un temps en secondes : 3750
le temps en HMS est : 1h 2mn 30s

Lecture du temps

Affichage du résultat sous format HMS

Indication importante : Vous devez impérativement écrire trois fonctions permettant chacune de donner le nombre d'heures, de minutes et de secondes. Vous devez aussi écrire une fonction « *lireTemps()* » permettant de renvoyer un entier lu à partir du clavier. Attention, cette fonction doit effectuer un contrôle de validité de la valeur saisie !

Exemple : *heures(3750)* renvoie la valeur 1 pour dire une heure.

minutes(3750) renvoie la valeur 2 pour dire deux minutes ($3750 - 1*3600 = 150$, ce qui donne 2 minutes)

secondes(3750) renvoie la valeur 30 pour dire 30 secondes ($3750 - 1*3600 - 2*60 = 30$)

lireTemps() permet de renvoyer un entier lu au clavier tout en s'assurant de la validité de la saisie.

```
def heures (temps) : .....
.....
.....

def minutes (temps) : .....
.....
.....

def secondes (temps) : .....
.....
.....

def lireTemps () : .....
.....
.....
.....
.....
.....
.....
.....
.....
.....

# Ici débute le programme principal
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
```

