

Frédéric  
LURET



# Python

## Exercices série 2

Version du 5 décembre 2024



## 1 A trier

### Question 1

Écrivez un programme qui trouvera tous ces nombres qui sont divisibles par 7 mais ne sont pas un multiple de 5, entre 2000 et 3200 (les deux inclus). Les nombres obtenus doivent être imprimés dans une séquence séparée par des virgules sur une seule ligne.

**Considérez la méthode Utiliser la plage (#Begin, #end) :**

Code python :

```
1 l=[]
2 for i in range(2000, 3201):
3     if (i%7==0) and (i%5!=0):
4         l.append(str(i))
5
6 print ','.join(l)
```

q001.py

### Question 2

Écrivez un programme qui peut calculer la factorielle d'un nombre donné. Les résultats doivent être imprimés dans une séquence séparée par des virgules sur une seule ligne. Supposons que l'entrée suivante soit fournie au programme : 8 Ensuite, la sortie doit être : 40320

**En cas de données d'entrée fournies à la question, il doit être supposé être une entrée de console. :**

Code python :

```
1 def fact(x):
2     if x == 0:
3         return 1
4     return x * fact(x - 1)
5
6 x=int(raw_input())
7 print fact(x)
```

q002.py

### Question 3

Avec un numéro intégral donné N, écrivez un programme pour générer un dictionnaire qui contient (i, i \* i) tel que c'est un nombre intégral entre 1 et n (les deux inclus). Et puis le programme devrait imprimer le dictionnaire. Supposons que l'entrée suivante



soit fournie au programme : 8 Ensuite, la sortie doit être : {1 : 1, 2 : 4, 3 : 9, 4 : 16, 5 : 25, 6 : 36, 7 : 49, 8 : 64}

**En cas de données d'entrée fournies à la question, il doit être supposé être une entrée de console. Envisagez d'utiliser dict () :**

**Code python :**

```
1 n=int(raw_input())
2 d=dict()
3 for i in range(1,n+1):
4     d[i]=i*i
5
6 print d
```

q003.py

#### Question 4

Écrivez un programme qui accepte une séquence de nombres séparés par des virgules à partir de la console et générer une liste et un tuple qui contient chaque numéro. Supposons que l'entrée suivante soit fournie au programme : 34,67,55,33,12,98 Ensuite, la sortie doit être : ['34', '67', '55', '33', '12', '98'] ('34', '67', '55', '33', '12', '98')

**En cas de données d'entrée fournies à la question, il doit être supposé être une entrée de console. La méthode tuple () peut convertir la liste en Tuple :**

**Code python :**

```
1 values=raw_input()
2 l=values.split(",")
3 t=tuple(l)
4 print l
5 print t
```

q004.py

#### Question 5

Définissez une classe qui a au moins deux méthodes : getString : pour obtenir une chaîne à partir de l'entrée de la console PrintString : Pour imprimer la chaîne en haut du boîtier. Veuillez également inclure une fonction de test simple pour tester les méthodes de classe.

**Utilisez la méthode \_\_init\_\_ pour construire certains paramètres :**

**Code python :**



```
1 class InputOutString(object):
2     def __init__(self):
3         self.s = ""
4
5     def getString(self):
6         self.s = raw_input()
7
8     def printString(self):
9         print self.s.upper()
10
11 strObj = InputOutString()
12 strObj.getString()
13 strObj.printString()
```

q005.py

### Question 6

Écrivez un programme qui calcule et imprime la valeur en fonction de la formule donnée :  $Q = \text{racine carrée de } [(2 * c * d) / h]$  Voici les valeurs fixes de C et H : C est 50. H est 30. D est la variable dont les valeurs doivent être entrées à votre programme dans une séquence séparée par des virgules. Exemple Supposons que la séquence d'entrée séparée des virgules suivante est donnée au programme : 100 150 180 La sortie du programme doit être : 18,22,24

**Si la sortie reçue est sous forme décimale, elle doit être arrondi à sa valeur la plus proche (par exemple, si la sortie reçue est de 26,0, elle doit être imprimée comme 26) En cas de données d'entrée fournies à la question, il doit être supposé être une entrée de console. :**

Code python :

```
1 #!/usr/bin/env python
2 import math
3 c=50
4 h=30
5 value = []
6 items=[x for x in raw_input().split(',')]
7 for d in items:
8     value.append(str(int(round(math.sqrt(2*c*float(d)/h))))))
9
10 print ','.join(value)
```

q006.py

### Question 7

Écrivez un programme qui prend 2 chiffres, x, y comme entrée et génère un tableau bidimensionnel. La valeur de l'élément dans la i-tème ligne et la colonne J-th du tableau doit être  $i * j$ . Remarque :  $i = 0,1 \dots, x-1$ ;  $J = 0,1, \dots, Y-1$ . Exemple Supposons



que les entrées suivantes soient données au programme : 3,5 Ensuite, la sortie du programme doit être : `[[0, 0, 0, 0, 0], [0, 1, 2, 3, 4], [0, 2, 4, 6, 8]]`

**Remarque : En cas de données d'entrée fournies à la question, il doit être supposé être une entrée de console sous un formulaire séparé des virgules. :**

Code python :

```
1 input_str = raw_input()
2 dimensions=[int(x) for x in input_str.split(',')]
3 rowNum=dimensions[0]
4 colNum=dimensions[1]
5 multilist = [[0 for col in range(colNum)] for row in range(rowNum)]
6
7 for row in range(rowNum):
8     for col in range(colNum):
9         multilist[row][col]= row*col
10
11 print multilist
```

q007.py

### Question 8

Écrivez un programme qui accepte une séquence de mots séparée par des virgules en entrée et imprime les mots dans une séquence séparée par des virgules après les avoir triés de manière alphabétique. Supposons que l'entrée suivante soit fournie au programme : sans, bonjour, sac, monde Ensuite, la sortie doit être : Sac, bonjour, sans, monde

**En cas de données d'entrée fournies à la question, il doit être supposé être une entrée de console. :**

Code python :

```
1 items=[x for x in raw_input().split(',')]
2 items.sort()
3 print ','.join(items)
```

q008.py

### Question 9

Écrivez un programme qui accepte la séquence des lignes en entrée et imprime les lignes après avoir réalisé que tous les caractères dans la phrase sont capitalisés. Supposons que l'entrée suivante soit fournie au programme : Bonjour le monde La pratique rend parfait Ensuite, la sortie doit être : BONJOUR LE MONDE La pratique rend parfait

**En cas de données d'entrée fournies à la question, il doit être supposé être une entrée de console. :**



Code python :

```
1 lines = []
2 while True:
3     s = raw_input()
4     if s:
5         lines.append(s.upper())
6     else:
7         break;
8
9 for sentence in lines:
10    print sentence
```

q009.py

### Question 10

Écrivez un programme qui accepte une séquence de mots séparés dans l'espace en entrée et imprime les mots après avoir retiré tous les mots en double et les tris de manière alphanumériquement. Supposons que l'entrée suivante soit fournie au programme : Bonjour le monde et la pratique rend à nouveau le monde parfait et bonjour Ensuite, la sortie doit être : Encore une fois et bonjour fait un monde de pratique parfait

**En cas de données d'entrée fournies à la question, il doit être supposé être une entrée de console. Nous utilisons le conteneur SET pour supprimer automatiquement les données dupliqués, puis utilisons Trised () pour trier les données. :**

Code python :

```
1 s = raw_input()
2 words = [word for word in s.split(" ")]
3 print " ".join(sorted(list(set(words))))
```

q010.py

### Question 11

Écrivez un programme qui accepte une séquence de nombres binaires à 4 chiffres séparés par virgules comme entrée, puis vérifiez s'ils sont divisibles par 5 ou non. Les nombres divisibles par 5 doivent être imprimés dans une séquence séparée par des virgules. Exemple : 0100,0011,1010,1001 Alors la sortie doit être : 1010 Remarques : Supposons que les données soient entrées par console.

**En cas de données d'entrée fournies à la question, il doit être supposé être une entrée de console. :**

Code python :



```
1 value = []
2 items=[x for x in raw_input().split(',')]
3 for p in items:
4     intp = int(p, 2)
5     if not intp%5:
6         value.append(p)
7
8 print ','.join(value)
```

q011.py

### Question 12

Écrivez un programme, qui trouvera tous ces chiffres entre 1000 et 3000 (tous deux inclus) de sorte que chaque chiffre du nombre est un nombre pair. Les nombres obtenus doivent être imprimés dans une séquence séparée par des virgules sur une seule ligne.

**En cas de données d'entrée fournies à la question, il doit être supposé être une entrée de console. :**

Code python :

```
1 values = []
2 for i in range(1000, 3001):
3     s = str(i)
4     if (int(s[0])%2==0) and (int(s[1])%2==0) and (int(s[2])%2==0) and
5         ↪ (int(s[3])%2==0):
6         values.append(s)
7
8 print ",".join(values)
```

q012.py

### Question 13

Écrivez un programme qui accepte une phrase et calculez le nombre de lettres et de chiffres. Supposons que l'entrée suivante soit fournie au programme : Bonjour le monde!123 Ensuite, la sortie doit être : Lettres 10 Chiffres 3

**En cas de données d'entrée fournies à la question, il doit être supposé être une entrée de console. :**

Code python :



```
1 s = raw_input()
2 d={"DIGITS":0, "LETTERS":0}
3 for c in s:
4     if c.isdigit():
5         d["DIGITS"]+=1
6     elif c.isalpha():
7         d["LETTERS"]+=1
8     else:
9         pass
10 print "LETTERS", d["LETTERS"]
11 print "DIGITS", d["DIGITS"]
```

q013.py

### Question 14

Écrivez un programme qui accepte une phrase et calculez le nombre de lettres en haut et de lettres minuscules. Supposons que l'entrée suivante soit fournie au programme : Bonjour le monde ! Ensuite, la sortie doit être : ÉTABLAND 1 CAS 1 Cas du bas 9  
**En cas de données d'entrée fournies à la question, il doit être supposé être une entrée de console. :**

Code python :

```
1 s = raw_input()
2 d={"UPPER CASE":0, "LOWER CASE":0}
3 for c in s:
4     if c.isupper():
5         d["UPPER CASE"]+=1
6     elif c.islower():
7         d["LOWER CASE"]+=1
8     else:
9         pass
10 print "UPPER CASE", d["UPPER CASE"]
11 print "LOWER CASE", d["LOWER CASE"]
```

q014.py

### Question 15

Écrivez un programme qui calcule la valeur d'un  $a + aa + aaa + aaaa$  avec un chiffre donné comme valeur de a. Supposons que l'entrée suivante soit fournie au programme : 9 Ensuite, la sortie doit être : 11106

**En cas de données d'entrée fournies à la question, il doit être supposé être une entrée de console. :**

Code python :





```
1 a = raw_input()
2 n1 = int( "%s" % a )
3 n2 = int( "%s%s" % (a,a) )
4 n3 = int( "%s%s%s" % (a,a,a) )
5 n4 = int( "%s%s%s%s" % (a,a,a,a) )
6 print n1+n2+n3+n4
```

q015.py

### Question 16

Utilisez une compréhension de la liste pour carréner chaque numéro impaire dans une liste. La liste est entrée par une séquence de nombres séparés par des virgules. Supposons que l'entrée suivante soit fournie au programme : 1,2,3,4,5,6,7,8,9 Ensuite, la sortie doit être : 1,3,5,7,9

**En cas de données d'entrée fournies à la question, il doit être supposé être une entrée de console. :**

Code python :

```
1 values = raw_input()
2 numbers = [x for x in values.split(",") if int(x)%2!=0]
3 print ",".join(numbers)
```

q016.py

### Question 17

Écrivez un programme qui calcule le montant net d'un compte bancaire basé sur un journal de transaction à partir de l'entrée de la console. Le format de journal des transactions est affiché comme suit : D 100 W 200

D signifie dépôt tandis que w signifie le retrait. Supposons que l'entrée suivante soit fournie au programme : D 300 D 300 W 200 D 100 Ensuite, la sortie doit être : 500

**En cas de données d'entrée fournies à la question, il doit être supposé être une entrée de console. :**

Code python :



```
1 netAmount = 0
2 while True:
3     s = raw_input()
4     if not s:
5         break
6     values = s.split(" ")
7     operation = values[0]
8     amount = int(values[1])
9     if operation=="D":
10        netAmount+=amount
11    elif operation=="W":
12        netAmount-=amount
13    else:
14        pass
15 print netAmount
```

q017.py

### Question 18

Un site Web oblige les utilisateurs à saisir le nom d'utilisateur et le mot de passe pour s'inscrire. Écrivez un programme pour vérifier la validité de la saisie du mot de passe par les utilisateurs. Voici les critères de vérification du mot de passe : 1. Au moins 1 lettre entre [A-Z] 2. Au moins 1 nombre entre [0-9] 3. Au moins 1 caractère de [\$ # @] 4. durée minimale du mot de passe de transaction : 6 5. Longueur maximale du mot de passe de transaction : 12 Votre programme doit accepter une séquence de mots de passe séparés par des virgules et les vérifiera conformément aux critères ci-dessus. Les mots de passe qui correspondent aux critères doivent être imprimés, chacun séparé par une virgule. Exemple Si les mots de passe suivants sont donnés en entrée au programme : ABD1234 @ 1, A F1 #, 2W3E \*, 2WE3345 Ensuite, la sortie du programme doit être : ABD1234 @ 1 :

Code python :

```
1
```

q018.py

### Question 19

Vous devez rédiger un programme pour trier les tuples (nom, âge, hauteur) par ordre croissant où le nom est la chaîne, l'âge et la taille sont des nombres. Les tuples sont entrés par console. Les critères de tri sont : 1 : Trier basé sur le nom ; 2 : puis trier en fonction de l'âge ; 3 : Puis triez par score. La priorité est ce nom > Age > Score. Si les tuples suivants sont donnés comme entrée au programme : Tom, 19,80 John, 20,90 Jony, 17,91 Jony, 17,93 JSON, 21,85 Ensuite, la sortie du programme doit être : [('John', '20', '90'), ('Jony', '17', '91'), ('Jony', '17', '93'), ('JSON', '21', '85 '), (' Tom



```
',' 19 ',' 80 ')]  
:
```

Code python :

```
1
```

q019.py

### Question 20

Définissez une classe avec un générateur qui peut itérer les nombres, qui sont divisibles par 7, entre une plage donnée 0 et n.

**Envisagez d'utiliser le rendement :**

Code python :

```
1 def putNumbers(n):  
2     i = 0  
3     while i<n:  
4         j=i  
5         i=i+1  
6         if j%7==0:  
7             yield j  
8  
9 for i in reverse(100):  
10     print i
```

q020.py

### Question 21

Un robot se déplace dans un avion à partir du point d'origine (0,0).Le robot peut se déplacer vers le haut, le bas, la gauche et la droite avec des étapes donné.La trace du mouvement du robot est indiquée comme suit : En haut 5 Vers le bas 3 Gauche 3 À droite 2 ; Les nombres après la direction sont des étapes.Veuillez écrire un programme pour calculer la distance de la position actuelle après une séquence de mouvement et un point d'origine.Si la distance est un flottant, imprimez simplement l'entier le plus proche. Exemple : Si les tuples suivants sont donnés comme entrée au programme : En haut 5 Vers le bas 3 Gauche 3 À droite 2 Ensuite, la sortie du programme doit être : 2

**En cas de données d'entrée fournies à la question, il doit être supposé être une entrée de console. :**

Code python :



```
1 import math
2 pos = [0,0]
3 while True:
4     s = raw_input()
5     if not s:
6         break
7     movement = s.split(" ")
8     direction = movement[0]
9     steps = int(movement[1])
10    if direction=="UP":
11        pos[0]+=steps
12    elif direction=="DOWN":
13        pos[0]-=steps
14    elif direction=="LEFT":
15        pos[1]-=steps
16    elif direction=="RIGHT":
17        pos[1]+=steps
18    else:
19        pass
20
21 print int(round(math.sqrt(pos[1]**2+pos[0]**2)))
```

q021.py

### Question 22

Écrivez un programme pour calculer la fréquence des mots à partir de l'entrée. La sortie doit sortir après le tri de la clé de manière alphanumérique. Supposons que l'entrée suivante soit fournie au programme : Nouveau sur Python ou choisir entre Python 2 et Python 3 ? Lisez Python 2 ou Python 3. Ensuite, la sortie doit être : 2 : 2 3. : 1 3 ? : 1 Nouveau : 1 Python : 5 Lire : 1 et : 1 Entre : 1 Choisir : 1 ou : 2 à : 1

**En cas de données d'entrée fournies à la question, il doit être supposé être une entrée de console. :**

Code python :

```
1 freq = {} # frequency of words in text
2 line = raw_input()
3 for word in line.split():
4     freq[word] = freq.get(word,0)+1
5
6 words = freq.keys()
7 words.sort()
8
9 for w in words:
10    print "%s:%d" % (w,freq[w])
```

q022.py

**Question 23**

Écrire une méthode qui peut calculer la valeur carrée du nombre

Utilisation de l'opérateur **\*\*** :

Code python :

```
1 def square(num):  
2     return num ** 2  
3  
4 print square(2)  
5 print square(3)
```

q023.py

**Question 24**

Python possède de nombreuses fonctions intégrées, et si vous ne savez pas comment l'utiliser, vous pouvez lire un document en ligne ou trouver des livres. Mais Python a une fonction de document intégrée pour toutes les fonctions intégrées. Veuillez écrire un programme pour imprimer certains documents de fonctions intégrées Python, telles que `ABS ()`, `int ()`, `brut_input ()` Et ajouter un document pour votre propre fonction

La méthode du document intégré est `__doc__` :

Code python :

```
1 print abs.__doc__  
2 print int.__doc__  
3 print raw_input.__doc__  
4  
5 def square(num):  
6     '''Return the square value of the input number.  
7  
8     The input number must be integer.  
9     '''  
10    return num ** 2  
11  
12 print square(2)  
13 print square.__doc__
```

q024.py

**Question 25**

Définissez une classe, qui a un paramètre de classe et a un même paramètre d'instance. Définissez un paramètre d'instance, Besoin de l'ajouter dans la méthode `__init__` Vous pouvez initier un objet avec un paramètre de construction ou définir la valeur plus tard :



Code python :

```
1 class Person:
2     # Define the class parameter "name"
3     name = "Person"
4
5     def __init__(self, name = None):
6         # self.name is the instance parameter
7         self.name = name
8
9 jeffrey = Person("Jeffrey")
10 print "%s name is %s" % (Person.name, jeffrey.name)
11
12 nico = Person()
13 nico.name = "Nico"
14 print "%s name is %s" % (Person.name, nico.name)
```

q025.py

### Question 26

Définissez une fonction qui peut calculer la somme de deux nombres.

**Définissez une fonction avec deux nombres comme arguments. Vous pouvez calculer la somme dans la fonction et renvoyer la valeur. :**

Code python :

```
1 def SumFunction(number1, number2):
2     return number1+number2
3
4 print SumFunction(1,2)
```

q026.py

### Question 27

Définissez une fonction qui peut convertir un entier en une chaîne et l'imprimer dans la console.

**Utilisez STR () pour convertir un nombre en chaîne. :**

Code python :

```
1 def printValue(n):
2     print str(n)
3
4 printValue(3)
```

q027.py

### Question 28



Définissez une fonction qui peut recevoir deux nombres entéraux sous forme de chaîne et calculer leur somme, puis l'imprimer dans la console.

Utilisez `int ()` pour convertir une chaîne en entier. :

Code python :

```
1 def printValue(s1,s2):
2     print int(s1)+int(s2)
3
4 printValue("3","4") #7
```

q028.py

### Question 29

Définissez une fonction qui peut accepter deux chaînes en entrée et les concaténer, puis l'imprimer dans la console.

Utiliser `+` pour concaténer les cordes :

Code python :

```
1 def printValue(s1,s2):
2     print s1+s2
3
4 printValue("3","4") #34
```

q029.py

### Question 30

Définissez une fonction qui peut accepter deux chaînes en entrée et imprimer la chaîne avec une longueur maximale dans la console. Si deux chaînes ont la même longueur, la fonction doit imprimer les chaînes al l ligne par ligne.

Utilisez la fonction `Len ()` pour obtenir la longueur d'une chaîne :

Code python :

```
1 def printValue(s1,s2):
2     len1 = len(s1)
3     len2 = len(s2)
4     if len1>len2:
5         print s1
6     elif len2>len1:
7         print s2
8     else:
9         print s1
10        print s2
11
12
13 printValue("one","three")
```

q030.py

**Question 31**

Définissez une fonction qui peut accepter un numéro entier en entrée et imprimer le "c'est un numéro pair" si le nombre est uniforme, sinon imprime "c'est un numéro impair".

Utilisez un opérateur % pour vérifier si un nombre est uniforme ou impair. :

Code python :

```
1 def checkValue(n):
2     if n%2 == 0:
3         print "It is an even number"
4     else:
5         print "It is an odd number"
6
7
8 checkValue(7)
```

q031.py

**Question 32**

Définissez une fonction qui peut imprimer un dictionnaire où les touches sont des nombres entre 1 et 3 (les deux inclus) et les valeurs sont carrés de clés.

Utilisez un modèle de valeur dict [key] = pour mettre l'entrée dans un dictionnaire. Utilisez \*\* Opérateur pour obtenir la puissance d'un nombre. :

Code python :

```
1 def printDict():
2     d=dict()
3     d[1]=1
4     d[2]=2**2
5     d[3]=3**2
6     print d
7
8
9 printDict()
```

q032.py

**Question 33**

Définissez une fonction qui peut imprimer un dictionnaire où les touches sont des nombres entre 1 et 20 (les deux inclus) et les valeurs sont carrés de clés.

Utilisez un modèle de valeur dict [key] = pour mettre l'entrée dans un dictionnaire. Utilisez \*\* Opérateur pour obtenir la puissance d'un nombre. Utilisez la plage () pour les boucles. :





Code python :

```
1 def printDict():
2     d=dict()
3     for i in range(1,21):
4         d[i]=i**2
5     print d
6
7
8 printDict()
```

q033.py

### Question 34

Définissez une fonction qui peut générer un dictionnaire où les touches sont des nombres entre 1 et 20 (les deux inclus) et les valeurs sont du carré de clés. La fonction doit simplement imprimer les valeurs uniquement.

Utilisez un modèle de valeur dict [key] = pour mettre l'entrée dans un dictionnaire. Utilisez \*\* Opérateur pour obtenir la puissance d'un nombre. Utilisez la plage () pour les boucles. Utilisez des clés () pour itérer les clés du dictionnaire. Nous pouvons également utiliser item () pour obtenir des paires de clés / valeur. :

Code python :

```
1 def printDict():
2     d=dict()
3     for i in range(1,21):
4         d[i]=i**2
5     for (k,v) in d.items():
6         print v
7
8
9 printDict()
```

q034.py

### Question 35

Définissez une fonction qui peut générer un dictionnaire où les touches sont des nombres entre 1 et 20 (les deux inclus) et les valeurs sont du carré de clés. La fonction doit simplement imprimer les touches uniquement.

Utilisez un modèle de valeur dict [key] = pour mettre l'entrée dans un dictionnaire. Utilisez \*\* Opérateur pour obtenir la puissance d'un nombre. Utilisez la plage () pour les boucles. Utilisez des clés () pour itérer les clés du dictionnaire. Nous pouvons également utiliser item () pour obtenir des paires de clés / valeur. :



Code python :

```
1 def printDict():
2     d=dict()
3     for i in range(1,21):
4         d[i]=i**2
5     for k in d.keys():
6         print k
7
8
9 printDict()
```

q035.py

### Question 36

Définissez une fonction qui peut générer et imprimer une liste où les valeurs sont du carré de nombres entre 1 et 20 (les deux inclus).

Utilisez **\*\*** Opérateur pour obtenir la puissance d'un nombre. Utilisez la **plage ()** pour les boucles. Utilisez **list.append ()** pour ajouter des valeurs dans une liste. :

Code python :

```
1 def printList():
2     li=list()
3     for i in range(1,21):
4         li.append(i**2)
5     print li
6
7
8 printList()
```

q036.py

### Question 37

Définissez une fonction qui peut générer une liste où les valeurs sont du carré de nombres entre 1 et 20 (les deux inclus). Ensuite, la fonction doit imprimer les 5 derniers éléments de la liste.

Utilisez **\*\*** Opérateur pour obtenir la puissance d'un nombre. Utilisez la **plage ()** pour les boucles. Utilisez **list.append ()** pour ajouter des valeurs dans une liste. Utilisez **[N1 : N2]** pour trancher une liste :

Code python :



```
1 def printList():
2     li=list()
3     for i in range(1,21):
4         li.append(i**2)
5     print li[-5:]
6
7
8 printList()
```

q037.py

### Question 38

Définissez une fonction qui peut générer et imprimer un tuple où la valeur est un carré de nombres entre 1 et 20 (les deux inclus).

Utilisez **\*\* Opérateur** pour obtenir la puissance d'un nombre. Utilisez la **plage ()** pour les boucles. Utilisez **list.append ()** pour ajouter des valeurs dans une liste. Utilisez **Tuple ()** pour obtenir un tuple d'une liste. :

Code python :

```
1 def printTuple():
2     li=list()
3     for i in range(1,21):
4         li.append(i**2)
5     print tuple(li)
6
7 printTuple()
```

q038.py

### Question 39

Écrivez un programme pour générer et imprimer un autre tuple dont les valeurs sont même des nombres dans le tuple donné (1,2,3,4,5,6,7,8,9,10).

Utilisez **"pour"** pour itérer le tuple Utilisez **Tuple ()** pour générer un tuple à partir d'une liste. :

Code python :

```
1 tp=(1,2,3,4,5,6,7,8,9,10)
2 li=list()
3 for i in tp:
4     if tp[i]%2==0:
5         li.append(tp[i])
6
7 tp2=tuple(li)
8 print tp2
```

q039.py

**Question 40**

Écrivez un programme qui accepte une chaîne en entrée pour imprimer "oui" si la chaîne est "oui" ou "oui" ou "oui", sinon imprimez "non".

Utilisez si la déclaration pour juger l'état. :

Code python :

```
1 s= raw_input()
2 if s=="yes" or s=="YES" or s=="Yes":
3     print "Yes"
4 else:
5     print "No"
```

q040.py

**Question 41**

Écrivez un programme qui peut filtrer même les nombres dans une liste en utilisant la fonction filtrante. La liste est : [1,2,3,4,5,6,7,8,9,10].

Utilisez Filter () pour filtrer certains éléments dans une liste. Utilisez Lambda pour définir des fonctions anonymes. :

Code python :

```
1 li = [1,2,3,4,5,6,7,8,9,10]
2 evenNumbers = filter(lambda x: x%2==0, li)
3 print evenNumbers
```

q041.py

**Question 42**

Écrivez un programme qui peut mapper () pour faire une liste dont les éléments sont un carré d'éléments dans [1,2,3,4,5,6,7,8,9,10].

Utilisez Map () pour générer une liste. Utilisez Lambda pour définir des fonctions anonymes. :

Code python :

```
1 li = [1,2,3,4,5,6,7,8,9,10]
2 squaredNumbers = map(lambda x: x**2, li)
3 print squaredNumbers
```

q042.py

**Question 43**

Écrivez un programme qui peut map () et filter () pour faire une liste dont les éléments sont carrés de nombre uniforme dans [1,2,3,4,5,6,7,8,9,10].



Utilisez Map () pour générer une liste. Utilisez Filter () pour filtrer les éléments d'une liste. Utilisez Lambda pour définir des fonctions anonymes. :

Code python :

```
1 li = [1,2,3,4,5,6,7,8,9,10]
2 evenNumbers = map(lambda x: x**2, filter(lambda x: x%2==0, li))
3 print evenNumbers
```

q043.py

#### Question 44

Écrivez un programme qui peut filtrer () pour faire une liste dont les éléments sont même un nombre entre 1 et 20 (les deux inclus).

Utilisez Filter () pour filtrer les éléments d'une liste. Utilisez Lambda pour définir des fonctions anonymes. :

Code python :

```
1 evenNumbers = filter(lambda x: x%2==0, range(1,21))
2 print evenNumbers
```

q044.py

#### Question 45

Écrivez un programme qui peut mapper () pour faire une liste dont les éléments sont un carré de nombres entre 1 et 20 (les deux inclus).

Utilisez Map () pour générer une liste. Utilisez Lambda pour définir des fonctions anonymes. :

Code python :

```
1 squaredNumbers = map(lambda x: x**2, range(1,21))
2 print squaredNumbers
```

q045.py

#### Question 46

Définissez une classe nommée American qui a une méthode statique appelée Print-nationalité.

Utilisez @StaticMethod Decorator pour définir la méthode statique de classe. :

Code python :



```
1 class American(object):
2     @staticmethod
3     def printNationality():
4         print "America"
5
6 anAmerican = American()
7 anAmerican.printNationality()
8 American.printNationality()
```

q046.py

### Question 47

Définissez une classe nommée American et sa sous-classe Newyorker.  
Utilisez la sous-classe de classe (parentClass) pour définir une sous-classe. :

Code python :

```
1 class American(object):
2     pass
3
4 class NewYorker(American):
5     pass
6
7 anAmerican = American()
8 aNewYorker = NewYorker()
9 print anAmerican
10 print aNewYorker
```

q047.py

### Question 48

Définissez une classe nommée cercle qui peut être construite par un rayon. La classe Circle a une méthode qui peut calculer la zone.

Utilisez Def MethodName (Self) pour définir une méthode. :

Code python :



```
1 class Circle(object):
2     def __init__(self, r):
3         self.radius = r
4
5     def area(self):
6         return self.radius**2*3.14
7
8 aCircle = Circle(2)
9 print aCircle.area()
```

q048.py

### Question 49

En supposant que nous avons des adresses e-mail au format "username@companyname.com", veuillez écrire un programme pour imprimer le nom d'utilisateur d'une adresse e-mail donnée. Les noms d'utilisateurs et les noms d'entreprise sont composés de lettres uniquement.

Exemple : Si l'adresse e-mail suivante est donnée comme entrée au programme :

John@google.com

Ensuite, la sortie du programme doit être :

John

En cas de données d'entrée fournies à la question, il doit être supposé être une entrée de console.

Utilisez `\{}` w pour faire correspondre les lettres. :

Code python :

```
1 import re
2 emailAddress = raw_input()
3 pat2 = "(\w+)@((\w+\.)+(com))"
4 r2 = re.match(pat2, emailAddress)
5 print r2.group(1)
```

q049.py

### Question 50

Écrivez un programme qui accepte une séquence de mots séparés par l'espace comme entrée pour imprimer les mots composés uniquement de chiffres.

Exemple : Si les mots suivants sont donnés en entrée au programme :

2 chats et 3 chiens.

Ensuite, la sortie du programme doit être :

['2', '3']

En cas de données d'entrée fournies à la question, il doit être supposé être une entrée de console.



Utilisez `re.findall()` pour trouver tous les sous-chaînes à l'aide de regex. :

Code python :

```
1 import re
2 s = raw_input()
3 print re.findall("\d+",s)
```

q050.py

### Question 51

Écrivez un commentaire spécial pour indiquer qu'un fichier de code source Python est dans Unicode.

:

Code python :

```
1 # -*- coding: utf-8 -*-
```

q051.py

### Question 52

Écrivez un programme pour calculer :

$f(n) = f(n-1) + 100$  quand  $n > 0$  et  $f(0) = 1$

avec une entrée  $n$  donnée par console ( $n > 0$ ).

Exemple : Si le  $n$  suivant est donné en entrée au programme :

5

Ensuite, la sortie du programme doit être :

500

En cas de données d'entrée fournies à la question, il doit être supposé être une entrée de console.

Nous pouvons définir une fonction récursive dans Python. :

Code python :

```
1 def f(n):
2     if n==0:
3         return 0
4     else:
5         return f(n-1)+100
6
7 n=int(raw_input())
8 print f(n)
```

q052.py



**Question 53**

La séquence Fibonacci est calculée en fonction de la formule suivante :

$$f(n) = 0 \text{ si } n = 0 \quad f(n) = 1 \text{ si } n = 1 \quad f(n) = f(n-1) + f(n-2) \text{ si } n > 1$$

Veillez écrire un programme pour calculer la valeur de  $F(n)$  avec une entrée  $N$  donnée par console.

Exemple : Si le  $n$  suivant est donné en entrée au programme :

7

Ensuite, la sortie du programme doit être :

13

En cas de données d'entrée fournies à la question, il doit être supposé être une entrée de console.

**Nous pouvons définir une fonction récursive dans Python. :**

**Code python :**

```
1 def f(n):
2     if n == 0: return 0
3     elif n == 1: return 1
4     else: return f(n-1)+f(n-2)
5
6 n=int(raw_input())
7 print f(n)
```

q053.py

**Question 54**

La séquence Fibonacci est calculée en fonction de la formule suivante :

$$f(n) = 0 \text{ si } n = 0 \quad f(n) = 1 \text{ si } n = 1 \quad f(n) = f(n-1) + f(n-2) \text{ si } n > 1$$

Veillez écrire un programme en utilisant la compréhension de la liste pour imprimer la séquence Fibonacci sous forme de virgule séparée avec une entrée  $N$  donnée par console.

Exemple : Si le  $n$  suivant est donné en entrée au programme :

7

Ensuite, la sortie du programme doit être :

0,1,1,2,3,5,8,13

**Nous pouvons définir une fonction récursive dans Python. Utilisez la compréhension de la liste pour générer une liste à partir d'une liste existante. Utilisez String.Join () pour rejoindre une liste de chaînes. En cas de données d'entrée fournies à la question, il doit être supposé être une entrée de console. :**

**Code python :**



```
1 def f(n):
2     if n == 0: return 0
3     elif n == 1: return 1
4     else: return f(n-1)+f(n-2)
5
6 n=int(raw_input())
7 values = [str(f(x)) for x in range(0, n+1)]
8 print ",".join(values)
```

q054.py

### Question 55

Veillez écrire un programme à l'aide du générateur pour imprimer les nombres pair entre 0 et N sous forme de virgule séparée tandis que N est entré par console.

Exemple : Si le n suivant est donné en entrée au programme :

10

Ensuite, la sortie du programme doit être :

0,2,4,6,8,10

**Utilisez le rendement pour produire la valeur suivante dans le générateur. En cas de données d'entrée fournies à la question, il doit être supposé être une entrée de console. :**

**Code python :**

```
1 def EvenGenerator(n):
2     i=0
3     while i<=n:
4         if i%2==0:
5             yield i
6             i+=1
7
8
9 n=int(raw_input())
10 values = []
11 for i in EvenGenerator(n):
12     values.append(str(i))
13
14 print ",".join(values)
```

q055.py

### Question 56

Veillez écrire un programme à l'aide du générateur pour imprimer les nombres qui peuvent être divisibles par 5 et 7 entre 0 et N sous forme séparée de virgules tandis que N est entrée par console.

Exemple : Si le n suivant est donné en entrée au programme :



100

Ensuite, la sortie du programme doit être :

0,35,70

Utilisez le rendement pour produire la valeur suivante dans le générateur. En cas de données d'entrée fournies à la question, il doit être supposé être une entrée de console. :

Code python :

```
1 def NumGenerator(n):
2     for i in range(n+1):
3         if i%5==0 and i%7==0:
4             yield i
5
6 n=int(raw_input())
7 values = []
8 for i in NumGenerator(n):
9     values.append(str(i))
10
11 print ",".join(values)
```

q056.py

### Question 57

Veillez rédiger des instructions d'affirmation pour vérifier que chaque numéro de la liste [2,4,6,8] est uniforme.

Utilisez "Affirmer l'expression" pour faire l'affirmation. :

Code python :

```
1 li = [2,4,6,8]
2 for i in li:
3     assert i%2==0
```

q057.py

### Question 58

Veillez écrire une fonction de recherche binaire qui recherche un élément dans une liste triée. La fonction doit renvoyer l'index de l'élément à rechercher dans la liste.

Utilisez si / elif pour gérer les conditions. :

Code python :



```
1 import math
2 def bin_search(li, element):
3     bottom = 0
4     top = len(li)-1
5     index = -1
6     while top>=bottom and index==-1:
7         mid = int(math.floor((top+bottom)/2.0))
8         if li[mid]==element:
9             index = mid
10        elif li[mid]>element:
11            top = mid-1
12        else:
13            bottom = mid+1
14
15    return index
16
17 li=[2,5,7,9,11,17,222]
18 print bin_search(li,11)
19 print bin_search(li,12)
```

q058.py

### Question 59

Veillez générer un flotteur aléatoire où la valeur se situe entre 10 et 100 à l'aide du module mathon mathon.

Utilisez `random.random ()` pour générer un flotteur aléatoire dans `[0,1]`. :

Code python :

```
1 import random
2 print random.random()*100
```

q059.py

### Question 60

Veillez écrire un programme pour produire un nombre pair aléatoire entre 0 et 10 inclus en utilisant le module aléatoire et la compréhension de la liste.

Utilisez `Random.CHOICE ()` à un élément aléatoire d'une liste. :

Code python :

```
1 import random
2 print random.choice([i for i in range(11) if i%2==0])
```

q060.py

**Question 61**

Veillez rédiger un programme pour générer une liste avec 5 nombres aléatoires entre 100 et 200 inclusifs.

Utilisez `random.sample ()` pour générer une liste de valeurs aléatoires. :

Code python :

```
1 import random
2 print random.sample(range(100), 5)
```

q061.py

**Question 62**

Veillez écrire un programme pour générer de manière aléatoire une liste avec 5 nombres, qui sont divisibles par 5 et 7, entre 1 et 1000 inclusifs.

Utilisez `random.sample ()` pour générer une liste de valeurs aléatoires. :

Code python :

```
1 import random
2 print random.sample([i for i in range(1,1001) if i%5==0 and i%7==0], 5)
```

q062.py

**Question 63**

Veillez écrire un programme pour imprimer au hasard un numéro entier entre 7 et 15 inclusif.

Utilisez `Random.RandRange ()` à un entier aléatoire dans une plage donnée. :

Code python :

```
1 import random
2 print random.randrange(7,16)
```

q063.py

**Question 64**

Veillez écrire un programme pour compresser et décompresser la chaîne "Hello World! Hello World! Hello World! Hello World!".

Utilisez `zlib.compress ()` et `zlib.decompress ()` pour compresser et décompresser une chaîne. :

Code python :



```
1 import zlib
2 s = 'hello world!hello world!hello world!hello world!'
3 t = zlib.compress(s)
4 print t
5 print zlib.decompress(t)
```

q064.py

### Question 65

Veillez rédiger un programme pour mélanger et imprimer la liste [3,6,7,8].

Utilisez la fonction `Shuffle ()` pour mélanger une liste. :

Code python :

```
1 from random import shuffle
2 li = [3,6,7,8]
3 shuffle(li)
4 print li
```

q065.py

### Question 66

Veillez écrire un programme pour générer toutes les phrases où le sujet se trouve dans ["I", "vous"] et le verbe est dans ["Play", "Love"] et l'objet est dans ["hockey", "football"].

Utilisez la notation de la liste [index] pour obtenir un élément d'une liste. :

Code python :

```
1 subjects=["I", "You"]
2 verbs=["Play", "Love"]
3 objects=["Hockey", "Football"]
4 for i in range(len(subjects)):
5     for j in range(len(verbs)):
6         for k in range(len(objects)):
7             sentence = "%s %s %s." % (subjects[i], verbs[j],
8                                     ↪ objects[k])
9             print sentence
```

q066.py

### Question 67

En utilisant la compréhension de la liste, veuillez écrire un programme pour imprimer la liste après avoir supprimé les numéros de suppression qui sont divisibles par 5 et 7 dans [12,24,35,70,88,120,155].



Utilisez la compréhension de la liste pour supprimer un tas d'éléments d'une liste. :

Code python :

```
1 li = [12,24,35,70,88,120,155]
2 li = [x for x in li if x%5!=0 and x%7!=0]
3 print li
```

q067.py

### Question 68

En utilisant la compréhension de la liste, veuillez écrire un programme générer un tableau 3 \* 5 \* 8 3D dont chaque élément est 0.

Utilisez la compréhension de la liste pour faire un tableau. :

Code python :

```
1 array = [[ [0 for col in range(8)] for col in range(5)] for row in
  ↳ range(3)]
2 print array
```

q068.py

### Question 69

En utilisant la compréhension de la liste, veuillez écrire un programme pour imprimer la liste après avoir supprimé la valeur 24 dans [12,24,35,24,88,120,155].

Utilisez la méthode de suppression de la liste pour supprimer une valeur. :

Code python :

```
1 li = [12,24,35,24,88,120,155]
2 li = [x for x in li if x!=24]
3 print li
```

q069.py

### Question 70

Définissez une personne de classe et ses deux classes enfants : hommes et femmes. Toutes les classes ont une méthode "Getgender" qui peut imprimer "masculin" pour la classe masculine et "féminine" pour la classe féminine.

Utilisez la sous-classe (parentClass) pour définir une classe d'enfants. :

Code python :



```
1 class Person(object):
2     def getGender( self ):
3         return "Unknown"
4
5 class Male( Person ):
6     def getGender( self ):
7         return "Male"
8
9 class Female( Person ):
10    def getGender( self ):
11        return "Female"
12
13 aMale = Male()
14 aFemale= Female()
15 print aMale.getGender()
16 print aFemale.getGender()
```

q070.py

### Question 71

Veillez écrire un programme qui accepte une chaîne de la console et l'imprimez dans l'ordre inverse.

Exemple : Si la chaîne suivante est donnée en entrée au programme :

Rise pour voter Sir

Ensuite, la sortie du programme doit être :

ris etov ot esir

Utilisez la liste [ : : - 1] pour itérer une liste dans un ordre inverse. :

Code python :

```
1 s=raw_input()
2 s = s[::-1]
3 print s
```

q071.py

### Question 72

Veillez écrire un programme qui accepte une chaîne de la console et imprime les caractères qui ont même des index.

Exemple : Si la chaîne suivante est donnée en entrée au programme :

H1E2L3L4O5W6O7R8L9D

Ensuite, la sortie du programme doit être :

Bonjour le monde

Utilisez la liste [ : : 2] pour itérer une liste par étape 2. :





Code python :

```
1 s=raw_input()  
2 s = s[::2]  
3 print s
```

q072.py

### Question 73

Veillez écrire un programme qui imprime toutes les permutations de [1,2,3]  
Utilisez `itertools.permutations()` pour obtenir des permutations de liste. :

Code python :

```
1 import itertools  
2 print list(itertools.permutations([1,2,3]))
```

q073.py

### Question 74

Écrivez un programme pour résoudre un ancien puzzle chinois classique : Nous comptons 35 têtes et 94 jambes parmi les poulets et les lapins dans une ferme. Combien de lapins et combien de poulets avons-nous ?

Utilisez pour la boucle pour itérer toutes les solutions possibles. :

Code python :

```
1 def solve(numheads,numlegs):  
2     ns='No solutions!'  
3     for i in range(numheads+1):  
4         j=numheads-i  
5         if 2*i+4*j==numlegs:  
6             return i,j  
7     return ns,ns  
8  
9 numheads=35  
10 numlegs=94  
11 solutions=solve(numheads,numlegs)  
12 print solutions
```

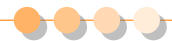
q074.py



## 2 Fonctions notions avancées



### Question 1



Question ????????

Code python :

```
1 #a corriger
```

pascorrige.py

## 3 Les compréhensions



### Question 1



Question ????????

Code python :

```
1 #a corriger
```

pascorrige.py

## 4 Gestion des fichiers



### Question 1



Question ????????

Code python :

```
1 #a corriger
```

pascorrige.py

## 5 Les exceptions



### Question 1



Question ????????



Code python :

```
1 #a corriger
```

pascorrige.py

## 6 Les sets



### Question 1



Question ????????

Code python :

```
1 #a corriger
```

pascorrige.py