

Ex 1 :

```
1 from numpy import array
2 T=array([0]*20, dtype='i')
3 def saisirNombre(msg,deb,fin):
4     k=int(input(msg))
5     while not deb<=k<=fin:
6         k=int(input(msg))
7     return k
8 def decomposer(x,T):
9     d= 2
10    y= 0
11    r= x % d
12    q= x // d
13    while q!=0:
14        if r == 0:
15            T[y]= d
16            y= y + 1
17            x= q
18        else:
19            d = d + 1
20            r= x % d
21            q= x // d
22    return y
23 def afficher (n,y,T):
24    print(n, ' = ',end=' ')
25    if y !=1:
26        for i in range(y-1) :
27            print(T[i], ' * ',end=' ')
28            print(T[i+1])
29    else :
30        print(T[0])
31 #programme principal
32 n=saisirNombre('Donnez un entier compris entre 2 et 100',2,100)
33 nb=decomposer(n,T)
34 afficher(n,nb,T)
```

Ex 2 :

```
1 #module qui permet de saisir une chaîne non vide
2 def saisie(msg):
3     mot=input(msg)
4     while(len(mot)==0):
5         mot=input(msg)
6     return mot
7 #module qui supprime les espaces superflus
8 def superflus(x):
9     p=x.find(' ')
10    while p>0:
11        x=x[0:p]+x[p+1:]
12        p=x.find(' ')
13    return x
14 #programme principal
15 ch=saisie("Tapez une chaîne non vide")
16 print('Votre chaîne Après suppression des espaces superflus :')
17 print(superflus(ch))
```

Ex3 :

```
1 from random import randint
2 def jouer (score1,score2):
3     score1 = 0
4     score2 = 0
5     while (score1 != 10) and (score2 != 10):
6         a = randint (1,6) + 1
7         b = randint (1,6) + 1
8         print( a, ' ',b)
9         if a > b :
10            score1 = score1 + 1
11        elif a < b:
12            score2 = score2 + 1
13    print('score1',score1)
14    print('score2',score2)
15    if score1 > score2 :
16        return 1
17    elif score1 < score2:
18        return 2
19 #programme principal
20 joueur1=0
21 joueur2=0
22 print("Le joueur gagnant c'est le joueur N° ",jouer (joueur1,joueur2));
```

Ex4 :

```
1 def saisie(msg):
2     mot=input(msg)
3     while(ord(mot[0].upper()) not in range(65,91)) or (mot.find(' ')!=-1) or (mot[len(mot)-1]!=' '):
4         mot=input(msg)
5     return mot
6
7
8 def renverser (mot):
9     ch1=''
10    while (mot.find(' ') != -1):
11        p= mot.find(' ')
12        ch1= mot[0:p]+' '+ch1
13        mot=mot[p+1:len(mot)]
14    ch1= mot + ' ' + ch1
15    return ch1
16 #programme principal
17 ch=saisie("Tapez une chaîne non vide")
18 print(renverser(ch))
```

Ex5 :

```
1 #module saisie permettant de saisir deux chaînes de memes longueurs
2 def saisie (msg1,msg2):
3     ch1=input(msg1)
4     ch2=input(msg2)
5     while((ch1=="") or (ch2 == "") or (ch1 == ch2)):
6         ch1=input(msg1)
7         ch2=input(msg2)
8     return ch1,ch2
9 #module verifier permettant de vérifier si deux mots sont anagrammes ou non
10 def verifier(ch1,ch2):
11     l=len(ch2)
12     ok= True
13     ch3=""
14     if l !=len(ch1) :
15         ok= False
16     else:
17         for i in range(l):
18             ch3= ch3 + ch1[i].upper()
19         i= 1
20         while((ok) and (i < l)):
21             p= ch3.find(ch2[i].upper())
22             if p == -1 :
23                 ok= False
24             else:
25                 i= i + 1
26     return ok
27 #programme principal
28 m1,m2=saisie("taper mot1","taper mot 2")
29 if verifier(m2,m1):
30     print(m2,' est une anagramme de ',m1)
31 else:
32     print(m2," n'est pas une anagramme de ",m1)
```

Ex6 :

```
1 from numpy import array
2 T=array([0]*20, dtype='i')
3 def saisirNombre(msg,deb,fin):
4     k=int(input(msg))
5     while not deb<=k<=fin:
6         k=int(input(msg))
7     return k
8 def remplir ( T,n):
9     for i in range(n):
10        T[i]=int(input("Donnez la valeur de l'élément N°"+str(i)))
11 def diviseur (x,y):
12     d = False
13     if (y % x) == 0:
14         d = True
15     return d
16 #programme principal
17 n=saisirNombre('Donnez la taille du tableau',1,15)
18 remplir (T,n)
19 ind_sup=saisirNombre("Donnez l'indice supérieur",2,n)
20 ind_inf=saisirNombre("Donnez l'indice inférieur",1,ind_sup-1)
21 p=int(input("Tapez l'entier qu'on cherche ces diviseurs et ces multiples"))
22 div=""
23 mult=""
24 for i in range(ind_inf,ind_sup+1):
25     if T[i] < p:
26         if diviseur(T[i],p) :
27             ch = str(T[i])
28             div = div+ ch + ' '
29     else:
30         if diviseur (p,T[i]) :
31             ch=str(T[i])
32             mult = mult + ch + ' '
33 print('Les diviseurs de ',p,' sont : ',div)
34 print ('Les multiples de ',p,' sont : ',mult)
```