

Déclaration	<pre>myList=list() # déclarer un tableau vide autrement : myList=[] l2=[2,3,True,'Hello'] print(l2)</pre>																				
Taille de la liste	<pre>myList=list() # déclarer un tableau vide autrement : myList=[] l2=[2,3.8,True,'Hello'] dim=len(l2) #la taille de la liste print("la taille de la liste est : {}".format(dim))</pre>																				
Indexing	<pre>myList=list() # déclarer un tableau vide autrement : myList=[] l2=[2,3.8,True,'Hello',10,11,12] #indexing to get value in position index print(l2[4]) # nomList[index] =&gt; index compris entre 0 et taille-1 #indexing to set value in position index l2[3]='bonjour' print(l2)</pre>																				
La commande del	<pre>#la commande del l=list(range(2,25,2)) l2=[i for i in range(2,18) if i%2==0] #compréhension list #del l #remove print("""*10,"before", ""*10,sep="") print(l) del l[5] print("""*10,"after", ""*10,sep="") print(l)</pre>																				
Indexation négative	<pre>L=list(range(1,11)) L = -10      -9      -8      -7      -6      -5      -4      -3      -2      -1</pre> <table border="1"><tr><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td></tr><tr><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td></tr></table>	1	2	3	4	5	6	7	8	9	10	0	1	2	3	4	5	6	7	8	9
1	2	3	4	5	6	7	8	9	10												
0	1	2	3	4	5	6	7	8	9												
Exemple pratique	<pre>#Exemple pratique : #déclarer une liste myList=[i for i in range(1,6)] #list(range(1,6)) #Supprimer l'element du milieu print(myList) index=len(myList)//2 #del myList[index] newValue=input("saier la nouvelle valeur :") myList[index]=newValue</pre>																				

	<b>print(myList)</b>
Methode append	<pre> # méthodes pour manipuler les listes # append =&gt; ajouter un élément à la fin de la liste l=[i for i in range(30) if i%3==0] #l.append(30) ajouter une valeur à la fin de la liste  """ <b>tableau l[] : entier</b> <b>var i :entier</b>      pour i=0 : 29         l[i]=alea()*(30-1)+1     FinPour     redim(31)     lire l[30]  <b>Fin</b> """ </pre>
Methode insert(pos,new Value)	<pre> #insert l=[i for i in range(10)] <b>print(l)</b> l.insert(1,2222) <b>print(l)</b> </pre>
Parcourir les elts de la liste	<pre> <b>fruits=['pomme','banane','orange','kiwi']</b> """for f in fruits:     print(f,end="****")"""  """"for i in range(len(fruits)):     if i&lt;len(fruits)-1:         print(fruits[i],end="****")     else:         print(fruits[i])""" <b>for i in range(len(fruits)):</b>     <b>print(fruits[i],end="****") if i&lt;len(fruits)-1 else print(fruits[i])</b> </pre>
Méthode extend	<pre> <b>fruits=['pomme','banane','orange','kiwi']</b>  <b>fruitsB=['avocat','fraise']</b> """for f in fruitsB:     fruits.append(f) print(fruits)""" #autrement <b>fruits.extend(fruitsB)</b> <b>print(fruits)</b> </pre>

Count	<pre> from random import randint  """for i in range(10):#range(0,10,1)     print(randint(1,10))""" """myList=list() for i in range(10):     myList.append(randint(1,10))""" myList=[randint(1,10) for i in range(8)] print(myList) x=2 """c=0 x=6 for l in myList:     if l==x:c+=1 print(c)""" c=myList.count(x) print(c) </pre>
index	<pre> myList=[2,4,3,2,6,3,7] print(myList)  x=3 print(myList.index(2))  pos=None for i in range(len(myList)):     if myList[i]==x:         pos=i;break print("la position de {} est {}".format(x,pos)) </pre>
Remove	<pre> myList=[2,4,3,2,6,3,7] print(myList) myList.remove(3) print(myList) </pre>
Deep Copy	<pre> myList=[2,4,3,2,6,3,7] l=myList # deepCopy =&gt; myList et l deux noms différents d'une même variable print(myList,l,sep="\n") myList.pop() print(myList,l,sep="\n") </pre>

Shallow Copy using slicing	<pre> myList=[2,4,3,2,6,3,7] #extract slice  =&gt; listName[posIn:posF:step]  slice from posIn to posF-step print(type(myList[1:5]))  # slice from list =&gt; list l=myList[:]  # vi = 0 vf=len(list) et step=1 print("***10","before","***10") print(myList,l,sep="\n") myList.pop() print("***10","after","***10") print(myList,l,sep="\n") </pre>

1								