

## Construire un tableau simple

à l'aide d'une boucle bornée	comprehension de liste
<pre>T = [] for i in range(100)     T.append(i)</pre>	

## Relever des valeurs

programmer la saisie de ces valeurs par une interface textuelle:

- une boucle principale demande une valeur à l'expérimentateur (`input`)
- la valeur saisie est ajoutée à la liste
- tant que le caractère de fin n'est pas saisi, continuer (par ex "S")

```
1. .
2. .
3. .
4. .
5. .
6. .
7. .
8. .
9. .
10. .
```

## Tracer un graphique

1. quelle instruction permet de créer la même liste `Ec`, mais cette fois, par compréhension de liste:
2. télécharger le fichier [data\\_parabolique.csv](#). Compléter le programme python suivant qui va lire les données du fichier, et stocker les valeurs dans des listes. Tracer alors `x` en fonction de `t`.

```
1. .
2. .
3. fichier = open('data_parabolique.csv','r') # ouverture du fichier de
données
4. lignes = fichier.readlines() # parcours du fichier par ligne
5.
6. lignes = lignes[1:] # eliminer la premiere ligne qui contient
les labels
7. x=[]
8. t=[]
9.
10. for ligne in lignes:
11.     t.append(float(ligne.split(';')[0]))
12.     x.append(float(ligne.split(';')[1]))
13. .
14. .
15. .
```

## Listes et algorithmes application aux sciences physiques et chimiques

```
16. .  
17. .  
18. .
```

3. Ajouter une nouvelle liste calculée:  $Ec$ . Prendre  $m = 1$ .

```
19. .  
20. .  
21. .
```

4. Tracer le graphique  $Ec$  en fonction de  $x$

```
22. .  
23. .  
24. .  
25. .  
26. .  
27. .
```

## Parcours d'une liste

1. adapter le script pour calculez la moyenne des valeurs de `Temperature`

```
1. Temperature=[18.5, 19.0, 20.4, 20.2, 19.8]  
2. .  
3. .  
4. .  
5. .  
6. .  
7. .
```

2. adapter le script pour déterminer la valeur maximale de la liste

```
1. Temperature=[18.5, 19.0, 20.4, 20.2, 19.8]  
2. .  
3. .  
4. .  
5. .  
6. .  
7. .
```

3. Utilisez votre script pour déterminer la valeur maximale de la liste `z` dans le fichier `data_parabolique.csv`

## Recherche du nombre d'occurrences dans une liste

```
1. molecule = ['H','H','H','C','C','O','C','H','H','H']  
2. def occurrences(molecule,c):  
3.     n = 0  
4.     for atome in molecule:  
5. .  
6. .  
7. .  
8. .
```

## Matrices

Ecrire un programme qui calcule la valeur moyenne sur toutes les valeurs RVB de cette matrice. Cela donnera l'intensité moyenne de cette image (sa luminosité).

```
1. im = [[(128,30,40), (130,32,44), (128,30,40)],  
2.      [(30,40,128), (44,32,130), (40,30,128)],  
3.      [(128,30,40), (130,32,44), (128,30,40)]]  
4. .  
5. .  
6. .  
7. .  
8. .  
9. .  
10. .  
11. .
```