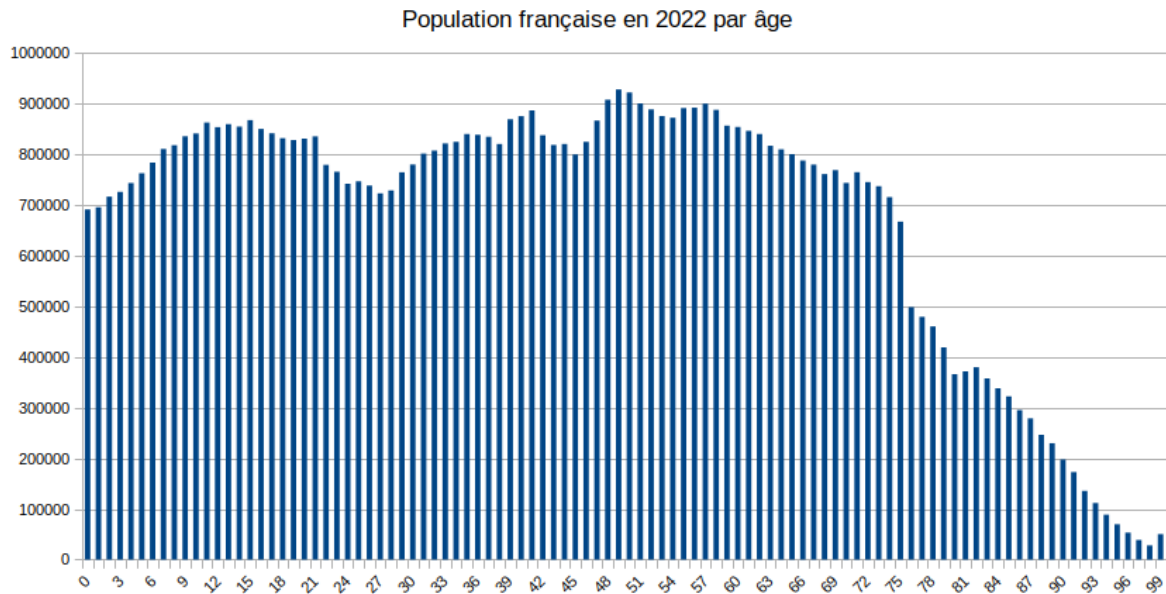


P5 - Listes - TP2

Parcours d'une liste - Corrigé

Le site internet de l'INSEE regorge de statistiques diverses. En particulier, la [page suivante](#) met à disposition la distribution de la population française par âge. Les données au 1^{er} janvier 2022 sont représentées sur le graphique suivant :



La cellule ci-dessous permet de définir une liste python `L` de taille 100 contenant ces données : `L[i]` donne le nombre de français dont l'âge est `i` sauf `L[99]` qui donne le nombre de français âgés de 99 ans ou plus.

⚠ Attention : Pensez à exécuter la cellule pour définir `L` !

```
In [1]: L = [690942, 695063, 716123, 725576, 743039, 762367, 783332,
      810097, 817631, 835242, 840895, 862319, 853243, 859175,
      854278, 867238, 849665, 841314, 831528, 827807, 830697,
      835200, 778873, 765236, 741535, 746399, 738131, 722662,
      728320, 764166, 779572, 801165, 806948, 821127, 824250,
      839830, 838306, 834328, 820145, 868806, 874869, 886274,
      837272, 818115, 820176, 799456, 824241, 866161, 907462,
      927759, 921723, 900060, 888235, 875245, 871895, 890989,
      891608, 899644, 887307, 856255, 853667, 846060, 839655,
      816587, 809514, 800032, 787325, 779454, 760616, 768507,
      743139, 764252, 744985, 736777, 715221, 666821, 498019,
      478997, 459971, 418516, 365621, 371202, 379585, 357369,
      337938, 322009, 294853, 278596, 246130, 229535, 197481,
      172622, 135469, 111515, 88537, 69657, 53102, 38627,
      27946, 50171]
```

Exercice 1 :

1. Comment obtenir la population âgée de 17 ans ?
2. Comment obtenir la liste des populations pour les âges de 20 ans à 29 ans (inclus) ?

```
In [2]: # 1
      L[17]
```

Out[2]: 841314

```
In [3]: # 2
        L[20:30]
```

```
Out[3]: [830697,
         835200,
         778873,
         765236,
         741535,
         746399,
         738131,
         722662,
         728320,
         764166]
```

Exercice 2 : Population totale

Écrire un programme qui calcule la population totale en France, que l'on notera S .

```
In [4]: # Première version - parcours sur les indices
        S = 0
        for i in range(len(L)):
            S = S + L[i]
        S

        # Seconde version - parcours sur les éléments
        S = 0
        for p in L:
            S = S + p
        S
```

```
Out[4]: 67813396
```

Exercice 3 : Population maximale et minimale

1. Écrire un programme permettant de déterminer la tranche d'âge dont la population est maximale. Le programme devra renvoyer l'âge **et** la population.
2. Faire de même pour la tranche d'âge dont la population est minimale.

```
In [5]: # 1
        imaxi = 0
        maxi = L[0]
        for i in range(1, len(L)):
            if L[i] > maxi:
                imaxi = i
                maxi = L[i]
        print(imaxi, maxi)
```

```
49 927759
```

```
In [6]: # 2
        imini = 0
        mini = L[0]
        for i in range(1, len(L)):
            if L[i] < mini:
                imini = i
                mini = L[i]
        print(imini, mini)
```

```
98 27946
```

Exercice 4 : Âge moyen

L'âge moyen d'une liste de population $[p_0, p_1, \dots, p_{99}]$ est approché par

$$m = \frac{1}{S} \sum_{i=0}^{99} ip_i,$$

avec S la population totale.

Écrire un programme qui calcule l'âge moyen des français.

```
In [7]: m = 0
for i in range(len(L)):
    m = m + i*L[i]
m/S
```

Out[7]: 41.72653094972563

Exercice 5 : Deuxième maximum

Écrire un programme qui détermine la deuxième tranche d'âge maximale.

```
In [8]: # Version 1 : on calcule le max1 et le max2 en même temps
if L[0] > L[1]:
    imaxi1, maxi1 = 0, L[0]
    imaxi2, maxi2 = 1, L[1]
else:
    imaxi1, maxi1 = 1, L[1]
    imaxi2, maxi2 = 0, L[0]
for i in range(2, len(L)):
    if L[i] > maxi1:
        imaxi2, maxi2 = imaxi1, maxi1
        imaxi1, maxi1 = i, L[i]
    elif L[i] > maxi2:
        imaxi2, maxi2 = i, L[i]
print(imaxi1, maxi1)
print(imaxi2, maxi2)
```

49 927759
50 921723

```
In [9]: # Version 2 : on calcule le max, on le "supprime" puis on recommence.
imaxi = 0
maxi = L[0]
for i in range(1, len(L)):
    if L[i] > maxi:
        imaxi = i
        maxi = L[i]

# Attention, supprimer le max avec del L[imaxi] nous fait prendre la correspondance avec l'âge
# L[i] ne sera plus toujours égal à la population d'âge i.
# On va plutôt mettre L[imaxi] à zéro.

L[imaxi] = 0
# Attention, on a changé L.
# C'est la dernière question, ça va, mais sinon ça peut être problématique pour la suite.

imaxi = 0
maxi = L[0]
for i in range(1, len(L)):
    if L[i] > maxi:
        imaxi = i
        maxi = L[i]
print('second max :', imaxi, maxi)
# Ceci est moins efficace car on parcourt 2 fois la liste.
```

second max : 50 921723

In []: