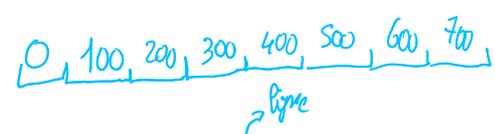
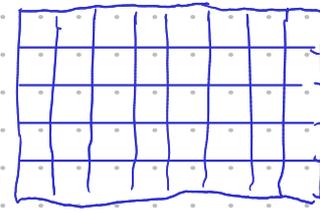


```

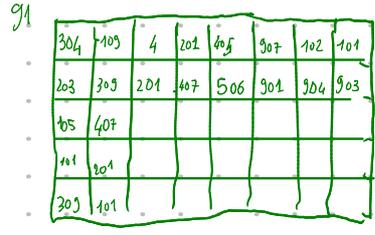
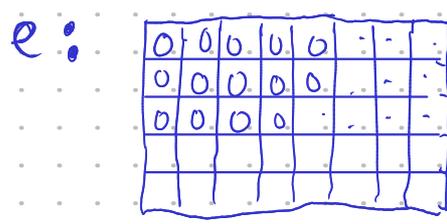
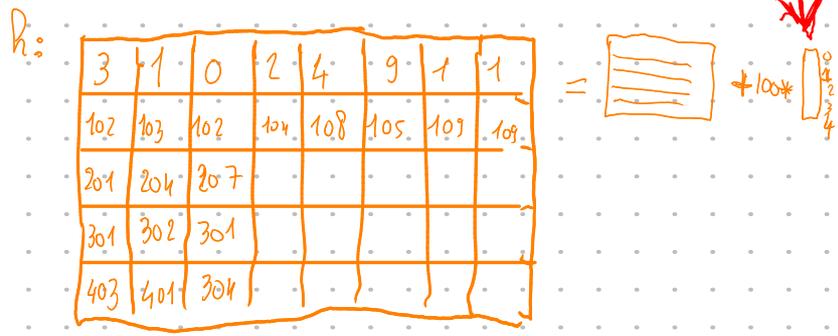
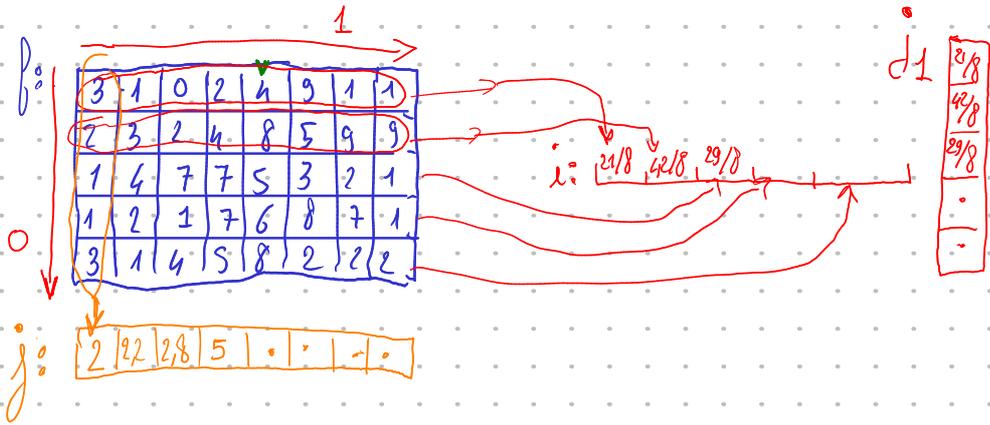
1 import numpy as np
2 def magic(arr):
3     return arr * 100 + np.random.randint(0, 10, (5, 8))
4
5 e = np.zeros((5, 8))
6 f = magic(e)
7 g1 = magic(f)
8 g2 = f + np.arange(0, 800, 100).reshape( (1, -1) )
9 g3 = f + 100*np.arange(8).reshape( (1, -1) )
10 h = f + 100*np.arange(5).reshape( (-1, 1) )
11 i = np.mean(f, axis=1)
12 j = np.mean(f, axis=0, keepdims=True)

```

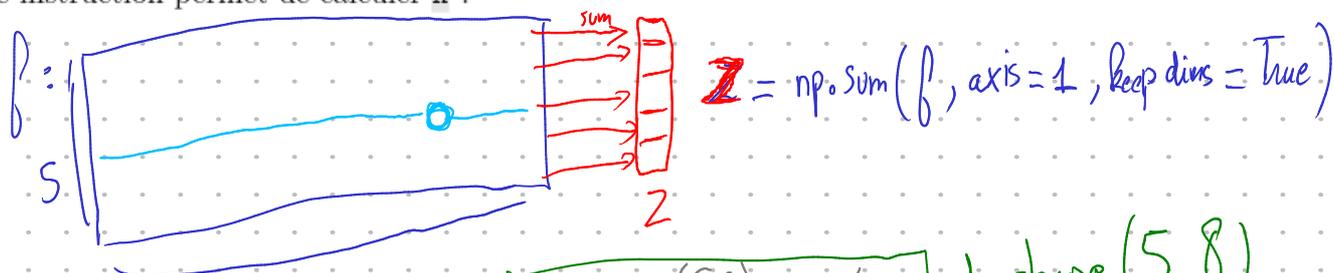


→ autour de colonne que nécessaire

axis=0



Q9) On veut « normaliser » f par ligne, c'est à dire que l'on veut que les valeurs de chaque ligne somment à 1. En pratique, nous voulons calculer k en divisant chaque valeur de f par la somme des valeurs de sa ligne. Quelle instruction permet de calculer k ?



$k = f / \text{np.sum}(f, \text{axis}=1, \text{keepdims}=\text{True})$

$R = f / z$ de shape (5,8)

Q10) Comment calculer un tableau z pour qu'il contiennent f , « normalisé » par colonne ?

$z = \text{np.sum}(f, \text{axis}=0, \text{keepdims}=\text{True})$

↑ shape (1,8)

$R = f / z$ → broadcasté en (5,8)

(5,8) (5,8) (1,8)

Q11) On veut maintenant faire que toutes les valeurs soit comprises entre 0 et 1, en travaillant à nouveau par ligne. Pour cela, à chaque valeur on va soustraire le minimum de la ligne et diviser par la différence entre le maximum et le minimum (selon la formule mathématique $\frac{v-\min}{\max-\min}$). Comment calculer dans une variable m , cette opération appliquée à f ?

Q12) Écrire trois fonctions `norm1_ligne`, `norm1_colonne` et `normmax_ligne` qui reprennent les 3 questions précédentes. Chaque fonction doit recevoir un tableau en paramètre et renvoyer le nouveau tableau.

$$m = \frac{f - \min}{\max - \min} \Rightarrow \frac{f - \min}{\max - \min} \Rightarrow \frac{f - \min}{\max - \min}$$

$f_{\min} = \text{np.min}(f, \text{axis}=1, \text{keepdims}=\text{True})$

$f_{\max} = \text{np.max}(f, \text{axis}=1, \text{keepdims}=\text{True})$

$m = (f - f_{\min}) / (f_{\max} - f_{\min})$

def normmax_ligne(f):

fmin = np.min(f, axis=1, keepdims=True)

fmax = np.max(f, axis=1, keepdims=True)

m = (f - fmin) / (fmax - fmin)

return m

def norm1_ligne(f):

k = f / np.sum(f, axis=1, keepdims=True)

return k

def norm1_colonne(f):

k = f / np.sum(f, axis=0, keepdims=True)

return k

On considère le tableau f de l'exercice précédent, mais vous devez donner des réponses les plus générales possibles. Par exemple, on évitera de supposer que f a 5 lignes et 8 colonnes.

Q13) Comment extraire les valeurs d'indices 4 à 7 (inclus) de la seconde ligne du tableau ?

Q14) Comment remplacer toutes les valeurs d'indices 4 à 7 (inclus) de la seconde ligne du tableau par 111 ?

Q15) Comment remplacer toutes les valeurs d'indices 4 à 7 (inclus) de la seconde ligne du tableau par celles de la troisième ligne ?

Q16) Comment extraire les 4 dernières valeurs de la seconde ligne du tableau ?

Q17) Comment extraire une colonne sur deux ?

Q18) Comment extraire une valeur sur deux de la colonne d'indice 3 ?

$$Q13) f[1, 4:8]$$

$$Q14) f[1, 4:8] = 111$$

$$Q15) f[1, 4:8] = f[2, 4:8]$$

$$Q16) f[1, -4:] \quad f[1, -4:]$$

$$Q17) f[:, ::2]$$

↑↑
pas

$$Q18) f[:, :2, 3]$$

3	1	0	2	4	9	1	1
2	3	2	4	8	5	9	9
1	4	7	7	5	3	2	1
1	2	1	7	6	8	7	1
3	1	4	5	8	2	2	2

3 _F	1 _(T)	0 _(T)	2 _(T)	4 _F	9 _F	1 _(T)	1 _(T)
2 _(T)	3 _F	2 _(T)	4 _F	8 _F	5 _F	0 _(T)	9 _F
1 _(T)	4 _F	7 _F	0 _(T)	5 _F	0 _(T)	2 _(T)	1 _(T)
1 _(T)	2 _(T)	1 _(T)	7 _F	6 _F	8 _F	7 _F	1 _(T)
3 _F	1 _(T)	4 _F	5 _F	8 _F	2 _(T)	2 _(T)	2 _(T)

60

Q19) Qu'affiche la ligne `print(np.sum(np.array([10, 20, 30])))` ?

Q20) Qu'affiche la ligne `print(np.sum(np.array([True, False, True])))` ?

T = True
F = False

- b1 = f == 0
- b2 = f <= 2
- b3 = f > -10

des T pour (5,8)

		1	0	1			
3 _F	1 _F	0 _(T)	2 _F	4 _F	9 _F	1 _F	1 _F
2 _F	3 _F	2 _F	4 _F	8 _F	5 _F	0 _(T)	9 _F
1 _F	4 _F	7 _F	0 _(T)	5 _F	0 _(T)	2 _F	1 _F
1 _F	2 _F	1 _F	7 _F	6 _F	8 _F	7 _F	1 _F
3 _F	1 _F	4 _F	5 _F	8 _F	2 _F	2 _F	2 _F

2

Q21) Que contient alors `b1` ?

Q22) Que contient alors `b2` ?

Q23) Que contient alors `b3` ?

Q24) Qu'affiche alors la ligne `print(np.sum(b1))` ?

Q25) Qu'affiche alors la ligne `print(np.sum(b2))` ?

Q26) Qu'affiche alors la ligne `print(np.sum(b3))` ?