
	Evaluation	
	Evaluation formative Python	
	Lycée Jules Haag – salle 265	

Consignes

Faire les exercices suivants, dans l'ordre. Un exercice par cellule dans Jupyter Notebook que vous nommerez « **Evaluation_python** » et que vous enregistrerez dans votre session locale.

Internet autorisé

Exercices python piscine autorisés

Exercice 1 : min max

Le but de cet exercice est de créer une fonction **min_max** qui pour deux entiers **a** et **b** renvoie le couple (**min,max**) où **min** est le minimum de **a** et **b** et **max** le maximum. Attention, si l'un des nombres ou les deux nombres entrés sont négatifs ou nuls, la fonction renvoie l'un des message d'erreur suivant :

→ 'Attention : l'un des nombres entré est négatif ou nul'

→ 'Attention : les deux nombres entrés sont négatifs ou nuls'

Entrée : Deux nombres entiers **a** et **b**.



Sortie : La fonction **min_max** doit renvoyer en résultat le couple (**min,max**) où **min** est le minimum de **a** et **b** et **max** le maximum.

Exercice 2 : pluriel

Le but de cet exercice est de créer une fonction **pluriel** qui rajoute un s au **mot** donné en entrée s'il ne se termine pas déjà par s. On ne s'intéresse pas aux exceptions comme chou, oeil, cheval...

Entrée : Un **mot** normalement mettable au pluriel.

Sortie : La fonction **pluriel** doit renvoyer le **mot** rentré avec un 's' en plus à la fin s'il ne se termine pas déjà par s.

	Evaluation	
	Evaluation formative Python	
	Lycée Jules Haag – salle 265	

Exercice 3 : multiple de 7.

Le but de cet exercice est de créer une fonction **contient_x7** qui renvoie True si la **liste** donnée en entrée contient un multiple de 7 et False sinon.

Entrée : Une **liste** de nombres.

Sortie : True si la **liste** contient un multiple de 7. False sinon.

Exercice 4 : gestion de dictionnaire.



Le but de cet exercice est de créer une fonction **gest_dico** qui prend en paramètre une liste de dictionnaires présentant les caractéristiques d'une personne `[{nom : ARNADI, prenom : alexandre, age : 24, classe : IA1}]` et un seuil d'âge.

4-1 : La fonction analyse la liste de dictionnaires et supprime tous les dictionnaires donc la valeur 'age' est supérieure au seuil passé en paramètre. La fonction renvoie la liste de dictionnaires mise à jour.

Entrée : `liste = [{nom : ARNADI, prenom : alexandre, age : 24, classe : IA1}, {nom : PIEDO, prenom : Arthur, age : 26, classe : IA2}, {nom : MAGNO, prenom : julien, age : 21, classe : IA1}, {nom : BERNARD, prenom : amandine, age : 31, classe : IA1}, {nom : ARMANDI, prenom : jerome, age : 19, classe : IA2}, {nom : BRETON, prenom : théo, age : 20, classe : IA2}]`

Sortie :

```
>> liste_dico_updated = gest_dico(liste, 25)
>> print(liste_dico_updated)
>> [{nom : ARNADI, prenom : alexandre, age : 24, classe : IA1}, {nom : MAGNO, prenom : julien, age : 21, classe : IA1}, {nom : ARMANDI, prenom : jerome, age : 19, classe : IA2}, {nom : BRETON, prenom : théo, age : 20, classe : IA2}]
```

	Evaluation	
	Evaluation formative Python	
	Lycée Jules Haag – salle 265	

4-2 : La fonction sépare les dictionnaires selon la classe (**IA1** ou **IA2**). Elle renvoie donc deux listes de dictionnaires, l'une regroupant les dictionnaires des personnes étant en **IA1** et l'autre en **IA2**. Le dernier élément de chaque liste doit être un dictionnaire contenant la moyenne de tous les âges, l'âge min et l'âge max de chaque liste de dictionnaires.

Entrée : `liste = [{nom : ARNADI, prenom : alexandre, age : 24, classe : IA1}, {nom : PIEDO, prenom : Arthur, age : 26, classe : IA2}, {nom : MAGNO, prenom : julien, age : 21, classe : IA1}, {nom : BERNARD, prenom : amandine, age : 31, classe : IA1}, {nom : ARMANDI, prenom : jerome, age : 19, classe : IA2}, {nom : BRETON, prenom : théo, age : 20, classe : IA2}]`

Sortie :

```
>> IA1, IA2 = gest_dico(liste)
>> print(IA1)
>> [{nom : ARNADI, prenom : alexandre, age : 24, classe : IA1}, {nom : MAGNO, prenom : julien, age : 21, classe : IA1}, {nom : BERNARD, prenom : amandine, age : 31, classe : IA1}, {moyenne :25.3, min :21, max :31}]

>> print(IA2)
>> [{nom : PIEDO, prenom : Arthur, age : 26, classe : IA2}, {nom : ARMANDI, prenom : jerome, age : 19, classe : IA2}, {nom : BRETON, prenom : théo, age : 20, classe : IA2}, {moyenne :21.6, min :19, max :26}]
```