

Exercice 2 : Etudiants et notes

Objectifs

Liste, list comprehension

Dictionnaire

OrderDict

Enoncé

On veut traiter les notes d'un ensemble d'étudiants.

Pour les étudiants Pierre, Claire et Francis, générer une liste de 5 notes pour chacun

Les 5 notes seront des nombres aléatoires entre 7 et 17 (méthode `randint(a, b)` du module `random`)

- Etape 1

Afficher la liste des étudiants et leurs notes

Calculer la moyenne de chacun

Exemple de résultat d'exécution du code

```
17:26:39.643 INFO Students list is ['Pierre', 'Claire', 'Francis']
17:26:39.643 INFO Students marks are {'Claire': [8, 9, 17, 9, 7], 'Francis': [10, 14, 17, 12, 12], 'Pierre': [11, 7, 10, 10, 11]}
17:26:39.643 INFO Students averages are {'Claire': 10.0, 'Francis': 13.0, 'Pierre': 9.8}
```

- Etape 2

Afficher la liste des étudiants classés selon leur moyenne

Afficher les couples (étudiant, moyenne) classés selon leur moyenne (l'occasion d'utiliser

`OrderedDict`)

Exemple de résultat d'exécution du code

```
17:35:52.635 INFO Students averages are {'Claire': 9.6, 'Francis': 9.2, 'Pierre': 13.0}
17:35:52.635 INFO Here is the list of students order by their average mark to exam of June: ['Pierre', 'Claire', 'Francis']
17:35:52.636 INFO Here is the list of (students, mark) order by their average mark to exam of June: [('Pierre', 13.0), ('Claire', 9.6), ('Francis', 9.2)]
```

- Etape 3

Définir une liste de 5 coefficients (0.5, 1, 2, 1, 2) et les associer aux notes pour le calcul de la moyenne ($avg = \frac{\sum(N_i * W_i)}{\sum(W_i)}$ avec N_i une note, W_i le poids associé à la note)

Afficher, pour chaque étudiant son nom, la liste des tuples (note, poids), sa moyenne

Exemple de résultat d'exécution du code

```
18:33:11.802 INFO Marks weights are [0.5, 1, 2, 1, 2]
18:33:11.802 INFO Students list is ['Pierre', 'Claire', 'Francis']
18:33:11.802 INFO Students marks are {'Claire': [17, 8, 7, 9, 17], 'Francis': [13, 7, 9, 7, 16], 'Pierre': [12, 16, 12, 17, 7]}
18:33:11.802 INFO Students name, marks & weight and avg are
18:33:11.802 INFO name Claire, marks & weight [(17, 0.5), (8, 1), (7, 2), (9, 1), (17, 2)], avg 11.31
18:33:11.802 INFO name Francis, marks & weight [(13, 0.5), (7, 1), (9, 2), (7, 1), (16, 2)], avg 10.85
18:33:11.802 INFO name Pierre, marks & weight [(12, 0.5), (16, 1), (12, 2), (17, 1), (7, 2)], avg 11.85
```

.../...

Indications

Les étudiants seront mis dans une liste

Les notes seront générées dans une liste (via le mécanisme list-comprehension)

L'association nom d'étudiant - liste de notes se fera avec un dictionnaire

A propos du tri

Si on a une liste et un dictionnaire sont en concordance (les éléments de la liste sont les clés du dictionnaire)

le tri de la liste selon les valeurs du dico est faisable selon cette indication

<https://docs.python.org/3.6/howto/sorting.html#sortinghowto> (voir au bas de page)

Concernant le logging, si on veut modifier le format de la date, une solution possible est

```
MY_FORMAT = "%(asctime)s.%(msecs)03d %(levelname)-6s %(message)s"
logging.basicConfig(format=MY_FORMAT, datefmt='%H:%M:%S', level=logging.DEBUG)
```