

Exercice 1 Partie 1

- 1) Les descripteurs de la table sont 'nom\_pays', 'nom\_capitale', 'latitude', 'longitude', 'code\_pays', 'continent'.
- 2) Il faut écrire l'instruction `table_capitales = lecture_fichier('capitales.csv')`
- 3) `table_capitales` est du type liste de listes
- 4) la variable `recherche_1` va contenir une liste de listes des capitales dont la latitude est négative.
- 5) le critère de sélection doit être `lambda x: x[5] == 'South America'`

Ainsi on doit saisir:

`recherche_2 = selectionner(table_capitales, lambda x: x[5] == 'South America')`

- a) le critère de sélection doit être `lambda x: x[5] == 'South America' and float(x[2]) > 0`  
Ainsi: `recherche_3 = selectionner(table_capitales, lambda x: x[5] == 'South America' and float(x[2]) > 0)`

Partie 2

- 1) `les_descripteurs = table_sacs[0]`
- 2) `s1` contient la liste de listes des sacs triées par longueurs de sacs croissantes. La ligne des descripteurs est absente.
- 3) `s2` contient la liste de listes des sacs triées par prix H.T de sacs décroissants. la ligne des descripteurs est absente.
- 4) `ma_selection = s2[0:5]`  
 ↑  
 En effet `s2` ne contient pas la liste des descripteurs puisqu'elle provient de `table_sacs[1:]`  
 ↑  
 la ligne de rang 5 ne sera pas prise, ce qui fait bien 5 lignes dont les rangs sont 0, 1, 2, 3, 4.

5) a) `s3 = sorted(table_sacs[1:], key=lambda ligne: ligne[1])`

pour chaque ligne de la liste de listes `table_sacs[1:]` la fonction `sorted` se basera sur l'élément de rang 1 qui est le nom du sac.

b) `[ligne for ligne in s3 if ligne[5] == 'noir']`

condition de filtrage

## Exercice 2

- 1) Oui les deux tables ont les mêmes descripteurs.
- 2) la façon de procéder de Betty est incorrecte pour deux raisons:
  - les identifiants pour les enregistrements de la table 2 sont les mêmes que pour les enregistrements de la table 1.
  - les températures n'ont pas les mêmes unités.
- 3) Première boucle for.  
Elle réalise la conversion des températures de t2 de degrés F en degrés C.

```
for ligne in t2[1:]:  
    ligne[1] = (ligne[1] - 32) * (5/9)
```

Remarque:

on peut améliorer en arrondissant à 1 décimale.  
La boucle for à écrire est alors:

```
for ligne in t2[1:]:  
    ligne[1] = round((ligne[1] - 32) * (5/9), 1)
```

Deuxième boucle for

Elle renumérote les id\_exp de t2 en leur ajoutant le nombre de lignes de t1 (en ne comptant pas les descripteurs)

```
for ligne in t2[1:]:  
    ligne[0] = ligne[0] + len(t1) - 1
```

↑  
est le nombre de lignes dans la table t1 en ne comptant pas la ligne des descripteurs.