

RND. Tables. Exercices

Exercice 1 : Vrai / Faux

| | VRAI | FAUX |
|--|------|------|
| Un fichier CSV est un fichier de données au format texte. | | |
| La virgule est toujours le séparateur des données dans un fichier CSV. | | |
| Si d est un dictionnaire, l'instruction $d[[10,15]] = 5$ provoque une erreur. | | |
| Une liste de dictionnaires est un format pratique pour gérer les données d'un fichier CSV. | | |
| Chaque donnée, même celle correspondant à un âge par exemple, sont de format texte. | | |
| La première ligne d'un fichier CSV contient les champs de la table de données. | | |

Exercice 2 : QCM

Pour chaque question, une seule réponse est correcte parmi les quatre proposées.

Question 1 : Qu'est-ce que le format de fichier CSV ?

1. un format de fichier mis au point par Microsoft pour Excel.
2. un format de fichier pour décrire une base de données.
3. un format de fichier où les données sont séparées par un caractère tel qu'une virgule.
4. un format de fichier décrivant une page Web.

Question 2 : On dispose d'un fichier nommé « *info.csv* », le programme à droite crée un tableau de données à partir du fichier.

```
nom, prenom, naissance, deces
lovelace, ada, 1815, 1852
von neumann, john, 1903, 1957
turing, alan, 1912, 1954
mccarthy, john, 1927, 2011
floyd, robert, 1936, 2001
```

```
file = open("info.csv", "r")
firstline = file.readline()      # chargement de la ligne d'entête
tableau = [line.split(',') for line in file] # chargement des données
```

Les index des lignes de ce tableau vont ...

1. de 0 à 3.
2. de 1 à 4.
3. de 0 à 4.
4. de 0 à 5.

Question 3 : On a extrait les deux premières lignes de différents fichiers. Déterminer celui qui est un authentique fichier CSV :

1.

```
Nom,Pays,Temps
Camille Muffat,France,241.45
```
2.

```
Nom Pays Temps
Camille Muffat France 241.45
```
3.

```
[{ "Nom": "Camille Muffat", "Pays": "France", "Temps": 241.45},
```
4.

```
[{ Nom: "Camille Muffat", Pays: "France", Temps: 241.45},
```

Question 4 : Que réalise l'instruction suivante ?

```
mon_fichier = open("exemple.txt", "r")
```

1. Elle permet d'ouvrir le fichier "exemple.txt" en mode lecture si le fichier est dans le même dossier que le fichier du programme Python comportant cette instruction.
2. Elle permet d'ouvrir le fichier "exemple.txt" en mode lecture même si le fichier n'est pas dans le même dossier que le fichier du programme Python comportant cette instruction.
3. Elle permet d'ouvrir le fichier "exemple.txt" en mode écriture si le fichier est dans le même dossier que le fichier du programme Python comportant cette instruction.
4. Elle permet d'ouvrir le fichier "exemple.txt" en mode écriture même si le fichier n'est pas dans le même dossier que le fichier du programme Python comportant cette instruction.

Question 5 : On considère une table de données <films> qui est une liste de dictionnaires "Titre", "Réalisateur", "Date". Un enregistrement de la table contient un titre de film, le nom de son réalisateur et l'année de sortie du film.

```
films = [{"Titre": "ET", "Réalisateur": "Spielberg", "Date": 1982}, {"Titre": "Ready Player One",
"Réalisateur": "Spielberg", "Date": 2018}, {"Titre": "The Artist", "Réalisateur": "Hazanavicius",
"Date": 2011},...]
```

On souhaite obtenir la liste des films réalisés par "Spielberg" ou par "Hazanavicius" depuis l'année 2000. Quelle doit être la condition utilisée pour sélectionner les enregistrements correspondants, lors du parcours de la table ?

1.

```
(film["Réalisateur"] == "Spielberg" or film["Réalisateur"] == "Hazanavicius") and film["Date"] >= 2000
```
2.

```
film["Réalisateur"] == "Spielberg" and film["Réalisateur"] == "Hazanavicius" and film["Date"] >= 2000
```
3.

```
film["Réalisateur"] == "Spielberg" or film["Réalisateur"] == "Hazanavicius" or film["Date"] >= 2000
```
4.

```
(film["Réalisateur"] == "Spielberg" and film["Réalisateur"] == "Hazanavicius") or film["Date"] >= 2000
```

Exercice 3 : QCM

Pour chaque question, une seule réponse est correcte parmi les quatre proposées.

Question 1 : Laquelle de ces listes de chaînes de caractères est triée en ordre croissant ?

1. ["112", "19", "27", "45", "8"]
2. ["8", "19", "27", "45", "112"]
3. ["8", "112", "19", "27", "45"]
4. ["19", "112", "27", "45", "8"]

Question 2 : On définit une table d'élèves et une liste finale de la façon suivante :

```
table_eleves = [{"prenom": "Ada", "nom": "Lovelace", "age": 17}, {"prenom": "Charles", "nom": "Babbage", "age": 18}, ....., {"prenom": "John", "nom": "Von Neumann", "age": 16}]
liste_finale = [eleve for eleve in table_eleves if eleve["age"] >= 18]
```

Que contient la liste finale ?

1. La liste des prénoms des élèves majeurs de la table.
2. La liste des âges des élèves majeurs de la table.
3. La liste des élèves majeurs de la table, chaque élément de la liste étant représenté par un dictionnaire.
4. La liste des élèves majeurs de la table, chaque élément de la liste étant représenté par une liste.

Question 3 : On a récupéré le contenu d'un fichier CSV contenant le nom, le prénom et l'âge de personnes dans une table implémentée par la liste de dictionnaires suivante :

```
table = [{"nom": "dupont", "prenom": "jean", "age": 16}, {"nom": "durant", "prenom": "pierre", "age": 15}, {"nom": "doe", "prenom": "jane", "age": 16}]
```

Quelle expression représente-t-elle la liste des noms des personnes dont l'âge a pour valeur 16 ?

1. [personne[nom] for personne in table if personne[age] == 16]
2. [personne["nom"] for personne in table if personne["age"] == 16]
3. [personne["nom"] for personne in table if personne["age"] = 16]
4. [nom if age == 16 for nom, age in table]

Question 4 : On considère la liste de p-uplets suivante :

```
table = [('Grace', 'Hopper', 'F', 1906), ('Tim', 'Berners-Lee', 'H', 1955), ('Ada', 'Lovelace', 'F', 1815), ('Alan', 'Turing', 'H', 1912)]
```

où chaque p-uplets représente un informaticien ou une informaticienne célèbre ; le premier élément est son prénom, le deuxième élément son nom, le troisième élément son sexe ('H' pour un homme, 'F' pour une femme) et le quatrième élément son année de naissance (un nombre entier entre 1000 et 2000).

On définit une fonction :

```
def fonctionMystere(table):
    mystere = []
    for ligne in table:
        if ligne[2] == 'F':
            mystere.append(ligne[1])
    return mystere
```

Que vaut *fonctionMystere(table)* ?

1. ['Grace', 'Ada']
2. [('Grace', 'Hopper', 'F', 1906), ('Ada', 'Lovelace', 'F', 1815)]
3. ['Hopper', 'Lovelace']
4. []

Question 5 : Laquelle de ces affirmations est vraie ?

1. On ne peut accéder au contenu d'un fichier CSV que par l'intermédiaire d'un programme Python.
2. CSV est un format de chiffrement des données.
3. Le format CSV a été conçu pour assurer la confidentialité d'une partie du code d'un programme.
4. Les fichiers CSV sont composés de données séparées par des caractères comme des virgules.

Exercices suivants :

A l'aide de Jupyter, **charger** les fichiers exemple.csv, top14.csv puis ceux du site de météo France. **Suivre** les consignes.