

Amérique du Nord – 2024 – sujet1 - Correction

Exercice 3 (8 points)

Partie A

1. Séparateur du fichier

Réponse :

Le séparateur est le **point-virgule ;**.

👉 Commentaire :

On le voit dans l'en-tête : `discipline; chapitre; question; réponse`

2. Justification

Le point-virgule est choisi car certaines réponses contiennent des **virgules** (ex : *Allemagne, Italie, Japon*). Si la virgule était utilisée comme séparateur, le fichier serait mal interprété.

3. Fonction `charger(nom_fichier)`

```
import csv

def charger(nom_fichier):
    with open(nom_fichier, 'r', encoding='utf-8') as fichier:
        donnees = list(csv.DictReader(fichier, delimiter=';'))
    return donnees
```

👉 Commentaire :

- *DictReader crée un dictionnaire pour chaque ligne.*
- *Les clés sont les noms des champs (discipline, chapitre, question, réponse).*
- *On utilise `delimiter=';'` car le séparateur est le point-virgule.*
- *On convertit en liste pour pouvoir accéder aux éléments par indice (`donnees[i]`).*

4. Méthode du module `time`

Réponse : `time.sleep()`

👉 *Commentaire :*

`time.sleep(n)` suspend l'exécution pendant `n` secondes.

Elle est souvent utilisée pour laisser le temps à l'utilisateur de lire un message.

5. Type de données `[i]`

Réponse : `dict`

👉 *Commentaire :*

- `donnees` est une **liste**.
 - Chaque élément de cette liste est un **dictionnaire**.
 - On peut donc accéder à une réponse avec : `donnees[i]['réponse']`
-

6. Compléter les lignes 41 à 44

Exemple attendu :

```
if reponse_utilisateur == donnees[i]['réponse']:
    print("Bonne réponse")
else:
    print("Mauvaise réponse")
```

👉 *Commentaire :*

- On compare deux chaînes de caractères.
 - Attention : la comparaison est sensible aux majuscules/minuscules.
 - Une amélioration possible serait d'utiliser `.lower()` pour éviter les erreurs de casse.
-

Partie B – Base de données relationnelle

7. Ajouter la boîte 5

```
INSERT INTO boite (id, lib, frequence)
VALUES (5, 'tous les quinze jours', 15);
```

👉 *Commentaire :*

INSERT INTO permet d'ajouter un nouvel enregistrement. On doit respecter l'ordre des attributs déclarés.

8. Corriger la date de Pearl Harbor

```
UPDATE flashcard
SET reponse = '7 décembre 1941'
WHERE question = 'Pearl Harbor - date';
```

👉 *Commentaire :*

- *UPDATE modifie une valeur existante.*
 - *La clause WHERE est indispensable pour éviter de modifier toutes les lignes.*
-

9. Liste des disciplines

```
SELECT lib
FROM discipline;
```

👉 *Commentaire :* *SELECT permet d'extraire des données. Ici, on demande uniquement la colonne lib.*

10. Chapitres de la discipline "histoire"

```
SELECT chapitre.lib
FROM chapitre
JOIN discipline ON chapitre.id_disc = discipline.id
WHERE discipline.lib = 'histoire';
```

👉 *Commentaire :*

- *JOIN permet de relier deux tables.*
- *On relie via la clé étrangère `id_disc`.*
- *La condition `WHERE` filtre uniquement la discipline "histoire".*

11. dentifiants des flashcards de la discipline "histoire"

```
SELECT flashcard.id
FROM flashcard
JOIN chapitre ON flashcard.id_ch = chapitre.id
JOIN discipline ON chapitre.id_disc = discipline.id
WHERE discipline.lib = 'histoire';
```

👉 *Commentaire : Ici on enchaîne deux jointures.*

- *flashcard → chapitre*
- *chapitre → discipline*

C'est un schéma classique à trois tables.

12. Supprimer les flashcards de la boîte 3

```
DELETE FROM flashcard
WHERE id_boite = 3;
```

👉 *Commentaire :*

DELETE supprime des lignes.

Sans `WHERE`, toute la table serait supprimée (erreur grave en examen).
