

Sujet Zéro – 2023 – sujet2 - Correction

Exercice 1 (4 points)

1. Commandes LINUX

a. Commandes Linux — Navigation dans l'arborescence

Le répertoire courant est **fiches**.

Commandes correctes :

• **Commande 5** : `cd ../collections/timbres`

👉 *Commentaire* : Depuis *fiches*, on remonte d'un niveau avec `..`, puis on redescend dans *collections/timbres*.

Les autres commandes sont incorrectes car :

- elles utilisent un mauvais chemin absolu,
- ou ne correspondent pas à l'arborescence donnée,
- ou ne partent pas du bon emplacement.

b. Commande Linux — Chemin absolu depuis la racine

Commande possible :

```
cd /home/documents/collections/timbres
```

👉 *Commentaire* : Un chemin absolu commence par `/` (racine du système de fichiers).

2. OSPF

a. Routage OSPF — Calcul du coût d'une liaison

$$C = \frac{10^8}{d}$$

Pour FastEthernet : $d=100 \text{ Mbit/s} = 100 \times 10^6 \text{ bit/s}$

$$C = \frac{10^8}{100 \times 10^6} = \frac{10^8}{10^8} = 1$$

Le coût d'une liaison FastEthernet est **1**.

👉 *Commentaire : Il faut bien convertir les Mbit/s en bit/s avant d'appliquer la formule.*

b. Routage OSPF — Recherche du chemin de coût minimal

- Liaison FTTH : 10 Gbit/s
 $d=10 \times 10^9$

$$C = \frac{10^8}{10 \times 10^9} = 0,01$$

Donc :

- coût FTTH = 0,01
- coût FastEthernet = 1

Le protocole OSPF choisit le chemin comportant **le plus de liaisons FTTH et le moins de FastEthernet**.

👉 *Commentaire : OSPF minimise la somme des coûts, pas le nombre de sauts.*

3. Fichier CSV — Descripteurs et valeurs

Extrait :

```
nom_timbre;annee_fabrication;nom_collectionneur
```

Descripteurs

- nom_timbre
- annee_fabrication
- nom_collectionneur

👉 *Commentaire : Les descripteurs correspondent aux en-têtes de colonnes.*

Valeurs associées

nom_timbre :

- Gustave Eiffel
- Marianne
- Alan Turing

annee_fabrication :

- 1950
- 1989
- 2012

nom_collectionneur :

- Dupont
- Durand
- Dupont

👉 *Commentaire : Chaque ligne correspond à un enregistrement.*

4. Base de données relationnelle

a. Définition d'une clé primaire

Une clé primaire est un attribut (ou un ensemble d'attributs) qui permet d'identifier **de manière unique** chaque tuple d'une relation.

👉 *Commentaire : Une clé primaire ne peut pas contenir de doublons.*

b. L'attribut nom peut-il être clé primaire ?

Non, "Gustave Eiffel" apparaît deux fois : il y a un doublon.

c. L'attribut annee_fabrication peut-il être clé primaire ?

Non. Plusieurs timbres peuvent avoir la même année de fabrication.

d. Proposition de solution

On peut utiliser une **clé primaire composée** :

- (nom, annee_fabrication)
- ou ajouter un nouvel attribut :
- id_timbre (identifiant unique)

👉 *Commentaire : L'ajout d'un identifiant artificiel est une pratique courante en base relationnelle.*

5. Requête SQL — UPDATE

```
UPDATE collectionneurs
SET ref_licence = 'Ythpswz'
WHERE nom = 'Dupond';
```

a. Résultat de la requête

Tous les collectionneurs ayant pour nom **Dupond** auront désormais la valeur 'Ythpswz' comme référence.

👉 *Commentaire : La condition WHERE s'applique à tous les enregistrements correspondant au nom Dupond.*

b. Pourquoi ref_licence ne peut plus être clé primaire ?

Parce que plusieurs lignes auront désormais la même valeur 'Ythpswz'.

👉 *Commentaire : Une clé primaire doit être unique ; ici il y a duplication.*

6. Requête SQL — SELECT avec condition

On veut les personnes nées en 1963 ou après.

```
SELECT nom, prenom, nbre_timbres  
FROM collectionneurs  
WHERE annee_naissance >= 1963;
```

👉 *Commentaire : La clause WHERE filtre les tuples selon la condition sur l'année.*
