

# EVALUATION MYSQL 2022

Réaliser par : Giovanni

## I- ANALYSE ET CORRECTION DE LA BASE DE DONNEES

### 1- La base de données

### 2- Correction des erreurs

Lors du chargement de la base de données dans Workbench , on a eu le code erreur 1825 : Failles to open the referenced table

- Pour le résoudre, on a modifier l'ordre d'exécution du script c'est-à-dire placé la table œuvre avant le table livre

### 3- Nombre de tuple après exécution du script

```
SELECT
    (SELECT COUNT(*) FROM `biblio`.`oeuvres`) +
    (SELECT COUNT(*) FROM `biblio`.`adherents`) +
    (SELECT COUNT(*) FROM `biblio`.`livres`) +
    (SELECT COUNT(*) FROM `biblio`.`emprunter`) AS `Nombres de
    tuples`;
```

### 4- Nombre d'attribut

#### a) Pour la table œuvre

```
DESCRIBE `biblio`.`oeuvres`;
```

#### b) Pour la table adhérent

```
DESCRIBE `biblio`.`adherents`;
```

#### c) Pour la table livres

```
DESCRIBE `biblio`.`livres`;
```

## 5- Affichage de la clé primaire

```
SELECT COLUMN_NAME AS PRIMARY_KEY
FROM INFORMATION_SCHEMA.COLUMNS
WHERE TABLE_SCHEMA = 'biblio'
AND TABLE_NAME = '<livres>'
AND COLUMN_KEY = 'PRI';
```

On change juste le « table\_name » pour les autres table

## II- INTERACTION AVEC LA BASE DE DONNEES

**NB : La clé primaire de la table Œuvre a été changée en id\_o, donc NO = id\_o**

### 9- Quels sont les livres actuellement empruntés ?

```
SELECT * FROM biblio.emprunter, biblio.livres where dateRet IS NULL
AND emprunter.NL=livres.NL;
```

### 10- Quels sont les livres empruntés par Jeannette Lecoeur ? Vérifier dans la réponse qu'il n'y a pas d'homonymes.

```
SELECT editeur FROM biblio.livres WHERE livres.NL IN
(SELECT NL FROM emprunter WHERE NA =
(SELECT NA FROM biblio.adherent WHERE nom='Jeannette' and
prenom='Lecoeur')) );
```

On mettant nom = Jeannette et prénom = Lecoeur , la requête envoie un ligne 0 donc pas de résultat .

Mais on inversant l'attribut on obtient Lecoeur Jeannette

**11- Quels sont tous les livres empruntés en septembre 2009.**

```
SELECT editeur FROM biblio.emprunter, biblio.livres WHERE dateEmp like  
'2009-06-%' AND emprunter.NL=livres.NL;
```

**12- Tous les adhérents qui ont emprunté un livre de Fedor Dostoievski.**

```
SELECT nom from biblio.adherents where NA IN  
  ( SELECT NA from biblio.emprunter where NL IN  
    (SELECT NL FROM biblio.livres where id_o IN  
      ( SELECT id_o FROM biblio.oeuvres where auteur  
        ='Fedor DOSTOIEVSKI'))));
```

**13- Un nouvel adhérent vient de s'inscrire : Olivier DUPOND, 76, quai de la Loire, 75019 Paris, téléphone : 0102030405**

```
INSERT INTO `biblio`.`adherents` (`NA`, `nom`, `prenom`, `adr`, `tel`)  
VALUES ('31', 'Olivier', 'Dupond', '76 quai de la Loire,75019 Paris',  
'0102030405');
```

**14- Martine CROZIER vient d'emprunter « Au cœur des ténèbres » que vous venez d'ajouter et « Le rouge et le noir » chez Hachette, livre n°23. Faire les mises à jour de la BD.**

```
INSERT INTO `biblio`.`emprunter` (`NL`, `dateEmp`, `dureeMax`, `NA`)  
VALUES ('23', '2022-06-28', '14', '7');  
INSERT INTO `biblio`.`emprunter` (`NL`, `dateEmp`, `dureeMax`, `NA`)  
VALUES ('31', '2022-06-28', '14', '7');
```

**15- M. Cyril FREDERIC ramène les livres qu'il a empruntés. Faire la mise à jour de la BD**

```
UPDATE `biblio`.`emprunter` SET `dateRet` = '2022-06-28' WHERE (`NL` =  
'9') and (`dateEmp` = '2022-06-17');
```

**16- M. Cyril FREDERIC essaye d'emprunter le livre n°23. Ecrire la requête. Que constatez-vous ?**

```
INSERT INTO `biblio`.`emprunter` (`NL`, `dateEmp`, `dureeMax`, `NA`)  
VALUES ('23', '2022-06-26', '14', '28');
```

On obtient une code erreur suivante :

ERROR 1062 (23000): Duplicate entry '23-2022-06-28' for key 'emprunter.PRIMARY'

**17- M. Cyril FREDERIC essaye d’emprunter le livre n°29. Écrire la requête. Que constatez-vous ?**

```
INSERT INTO `biblio`.`emprunter` (`NL`, `dateEmp`, `dureeMax`, `NA`)  
VALUES ('29', '2022-06-26', '21', '28');
```

La requête passe sans problème

**18- Quels sont le ou les auteurs du titre « Voyage au bout de la nuit »**

```
SELECT auteur from biblio.oeuvres where titre ='Voyage au bout de la nuit';
```

**19- Quels sont les ou les éditeurs du titre « Narcisse et Goldmund »**

On avait 2 possibilités mais on a choisi d’utiliser une requête imbriquée avec une sous requête.

**Nb : id\_o = NO** pour la clé primaire

```
SELECT editeur from biblio.livres where NL IN  
(select id_o from biblio.oeuvres where titre ='Narcisse et Goldmund');
```

**20- Quels sont les adhérents actuellement en retard ?**

**IS NULL** indique que la colonne dateRet est vide c’est à dire qu’il n’y a pas encore un retrait

```
SELECT nom FROM biblio.adherents, biblio.emprunter WHERE  
adherents.NA = emprunter.NA and dateRet IS NULL;
```

**21- Quels sont les livres actuellement en retard ?**

```
SELECT titre FROM biblio.oeuvres WHERE id_o IN  
(SELECT id_o FROM biblio.livres WHERE NL in  
(SELECT NL FROM biblio.emprunter WHERE dateRet is null  
));
```

**22- Quels sont les adhérents en retard avec le nombre de livre en retard et la moyenne du nombre de jour de retard .**

```
SELECT adherents.NA , prenom, nom ,count(emprunter.NL) as 'nombre de
livres
en retard', AVG((to_days(current_date()-
to_days(date_add(dateEmp,INTERVAL + dureeMax DAY ))))AS 'Moyenne
du nombre de jour de retard'
FROM biblio.adherents,biblio.livres, biblio.emprunter WHERE dateRet IS
NULL AND emprunter.NA = adherents.NA AND datediff(NOW(),
DATE_ADD(dateEmp, INTERVAL dureeMax day)) > dureeMax
GROUP BY NA;
```

**23- Nombre de livres empruntées par auteur**

```
SELECT COUNT(NL) AS Nbr_Livre,auteur FROM
biblio.Livre_Oeuvres_Adherent_Emprunt GROUP BY auteur;
```

**24- Nombre de livres empruntés par éditeur.**

```
SELECT editeur, count(dateEmp) AS 'nombre_editeur' FROM
biblio.livres, biblio.emprunter WHERE livres.NL = emprunter.NL group by
editeur ORDER BY count(dateEmp) desc;
```

**25- Durée moyenne des emprunts rendus. On commencera par afficher les durées des emprunts rendus.**

```
SELECT dateEmp,dateRet,dateRet-dateEmp AS Duree,NL
FROM biblio.emprunter
WHERE dateRet IS NOT NULL ORDER BY Duree;
```

On constate une date négative sur 2 tuples car la date de retour est antérieure à la date d'emprunt.

**26- Durée moyenne des retards pour l'ensemble des emprunts.**

```

SELECT ROUND(AVG(CASE
                    WHEN dateRet IS NULL THEN
                        CURRENT_DATE() - dateEmp
                    ELSE dateRet - dateEmp
                END)) AS Duree_retard FROM biblio.emprunter
WHERE ((CASE
        WHEN dateRet IS NULL THEN
            CURRENT_DATE() - dateEmp
        ELSE dateRet - dateEmp
    END) > dureeMax
    OR (dateRet - dateEmp) <= dureeMax) and (dateRet -
    dateEmp)>=0;

```

**Durée moyenne des retards parmi les seuls retardataires.**

```

SELECT ROUND(AVG(CASE
                    WHEN dateRet IS NULL THEN
                        CURRENT_DATE() - dateEmp
                    ELSE dateRet - dateEmp
                END)) AS Duree_retard FROM emprunter
WHERE (CASE
        WHEN dateRet IS NULL THEN CURRENT_DATE() -
        dateEmp
        ELSE dateRet - dateEmp
    END) > dureeMax and (dateRet - dateEmp)>=0;

```