

ASRBD 2 - Evaluation SQL - Base "music"

Ecrivez et testez les requêtes suivantes numérotées de 1 à 10.

Pour chaque requête, vous fournirez :

- le code SQL de la requête
- une copie écran du résultat obtenu dans un fichier texte **qui portera votre nom**.

Envoyez ce fichier par mail à l'adresse suivante : **bruno.bonnefoy@ext.groupe-igslyon.com**

Vous n'êtes pas obligé de traiter les requêtes dans l'ordre mais **chaque requête doit être numérotée**.

Requêtes

01 – Liste d'albums – Affichez titre album, nom de l'artiste, année de sortie, albums sortis entre 1970 et 1990 (années incluses). Tri du plus récent au plus ancien puis par nom d'album croissant. Proposez deux solutions.

```
SELECT albums.alb_titre, artistes.art_nom, albums.alb_annee
FROM albums, artistes
WHERE albums.art_id = artistes.art_id
AND albums.alb_annee BETWEEN 1970 AND 1990
ORDER BY albums.alb_annee DESC, artistes.art_nom ASC
```

OU

```
SELECT albums.alb_titre, artistes.art_nom, albums.alb_annee
FROM albums, artistes
WHERE albums.art_id = artistes.art_id
AND albums.alb_annee >= 1970 AND albums.alb_annee <= 1990
ORDER BY albums.alb_annee DESC, artistes.art_nom ASC
```

02 – Affichez le prix moyen des albums dont la date de sortie est strictement supérieure à 1990.

```
SELECT ROUND(AVG(albums.alb_prix), 2)
FROM albums
WHERE albums.alb_annee > 1990;
```

03 – Créez une vue des ventes appelée « Ven_view1 » avec les éléments suivants : date de la vente, nom de l'album, prix de vente

```
CREATE VIEW ven_view1 AS
(SELECT ventes.ven_date, albums.alb_titre, ventes.ven_prix
FROM ventes, albums
WHERE ventes.alb_id = albums.alb_id)
```

04 – Effectuez une baisse de 10% du prix de tous les albums dont le prix est supérieur ou égal à 18 euros

```
UPDATE albums
SET albums.alb_prix = albums.alb_prix * 0.9
WHERE albums.alb_prix >= 18
```

05 – Affichez : titre album, année de sortie, prix de l'album et nom artiste, de l'album (ou des albums) le(s) plus récent(s). Tri par nom album croissant.

```
SELECT albums.alb_titre, albums.alb_annee, albums.alb_prix, artistes.art_nom
FROM albums, artistes
WHERE albums.art_id = artistes.art_id
AND albums.alb_annee = (SELECT MAX(albums.alb_annee) FROM albums)
ORDER BY albums.alb_titre ASC
```

06 – Liste des genres qui ne sont affectés à aucun artiste dans la base de données. Affichez : libellé du genre. Tri par libellé du genre croissant.

```
SELECT genres.gen_libelle
FROM genres
WHERE genres.gen_id NOT IN (SELECT artistes.gen_id FROM artistes)
ORDER BY genres.gen_libelle ASC;
```

07 – Nombre d'artistes par genre et par pays
Affichez : nombre d'artistes, libellé du genre musical et libellé du pays.
Tri par nombre d'artistes décroissant, puis par libellé du genre croissant.

```
SELECT COUNT(artistes.art_id), genres.gen_libelle, pays.pay_libelle
FROM artistes, genres, pays
WHERE artistes.gen_id = genres.gen_id
AND artistes.pay_id = pays.pay_id
GROUP BY artistes.gen_id, artistes.pay_id
ORDER BY COUNT(artistes.art_id) DESC, genres.gen_libelle ASC
```

08 – Liste d'artistes – Affichez nom artiste, libellé du pays auquel appartient l'artiste, artistes dont le nom commence par un 'M'. Tri par nom d'artiste croissant.

```
SELECT artistes.art_nom, pays.pay_libelle
FROM artistes, pays
WHERE artistes.pay_id = pays.pay_id
AND artistes.art_nom LIKE 'M%'
ORDER BY artistes.art_nom ASC
```

09 – Qui est (sont) le(s) membre(s) actuel(s) des Rolling Stones ?

Affichez : nom du membre et instrument joué.

Tri par ordre alphabétique du nom de membre.

```
SELECT membres.mem_nom, instruments.ins_libelle
FROM membres, artistes, historique, instruments
WHERE historique.art_id = artistes.art_id
AND historique.mem_id = membres.mem_id
AND membres.ins_id = instruments.ins_id
AND artistes.art_nom = "Rolling Stones"
AND YEAR(NOW()) BETWEEN historique.his_debut AND historique.his_fin
ORDER BY membres.mem_nom ASC
```

10 – Créez une procédure appelée « Proc_Albums » qui liste les albums dont l'année de sortie est égale à l'année passée en paramètre de la procédure.

Affichez : titre de l'album, nom de l'artiste, libellé du genre musical.

Tri par ordre alphabétique du nom de l'artiste.

Pour tester votre procédure, appelez-la avec l'année 1982 et joignez le résultat à votre compte-rendu.

```
DROP PROCEDURE IF EXISTS Proc_Albums;
DELIMITER //
CREATE PROCEDURE Proc_Albums (IN année INTEGER(4))
BEGIN
SELECT albums.alb_titre, artistes.art_nom, genres.gen_libelle
FROM albums, artistes, genres
WHERE albums.art_id = artistes.art_id
AND artistes.gen_id = genres.gen_id
AND albums.alb_annee = année
ORDER BY artistes.art_nom ASC;
END //
DELIMITER ;

CALL Proc_Albums(1982);
```