

Explication correction TP bonus conception SQL

Vous trouverez ci-dessous l'explication de la correction de chaque requête de consultation du TP bonus.

1 Afficher la liste des membres du **personnel** avec leur id, nom, prénom, nom de poste

Correction :

```
SELECT id_personnel, nom_personnel, prenom_personnel, nom_poste FROM  
personnel INNER JOIN poste ON personnel.id_poste = poste.id_poste;
```

Explication :

Pour effectuer cette requête et récupérer le **nom du poste** du membre du **personnel**, en plus des informations contenues dans la table **personnel** nous devons réaliser une jointure (**INNER JOIN**) afin de pouvoir retourner le **nom du poste contenu** dans la table **poste**.

2 Afficher la liste des **livreurs** avec leur id, nom, prénom,

Correction :

```
SELECT id_personnel, nom_personnel, prenom_personnel FROM personnel INNER  
JOIN poste ON personnel.id_poste = poste.id_poste WHERE nom_poste =  
'Livreur';
```

Explication :

Pour effectuer cette requête et récupérer uniquement les membres du **personnel** qui ont comme **nom de poste** : livreur, en plus des informations contenues dans la table **personnel**. Nous devons réaliser une jointure (**INNER JOIN**) avec la table **poste** pour pouvoir chercher uniquement les enregistrements qui ont comme **nom de poste** : livreur.

3 Afficher la liste des **produits** de type **alimentaire** avec leur id, nom, description, image, tarif, stock.

Correction :

```
SELECT id_produit, nom_produit, description_produit, image_produit,  
prix_produit, stock_produit FROM produit INNER JOIN categorie ON  
produit.id_categorie = categorie.id_categorie WHERE nom_categorie =  
"Alimentation";
```

Explication :

Pour effectuer cette requête et récupérer uniquement les produits qui ont comme **nom de catégorie** : Alimentation, en plus des informations contenues dans la table **produit**. Nous devons réaliser une jointure (**INNER JOIN**) avec la table **categorie** pour pouvoir chercher uniquement les enregistrements qui ont comme **nom de catégorie** : Alimentation.

Auteur :

Mathieu MITHRIDATE

Date création :

xx / xx / 20xx

Relu, validé & visé par :

☒ Jérôme CHRETIENNE
☒ Sophie POULAKOS
☒ Mathieu PARIS

Date révision :

xx / xx / 20xx



Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.

Explication correction TP bonus conception SQL

4 Afficher la liste des **produits** du **ticket 2** avec leur nom, description, prix, quantité ajouté.

Correction :

```
SELECT nom_produit, description_produit, prix_produit, quantite FROM  
ajouter INNER JOIN produit ON ajouter.id_produit = produit.id_produit WHERE  
ajouter.id_ticket = 2;
```

Explication :

Pour effectuer cette requête et récupérer uniquement les **produits** qui ont été ajouté au **ticket 2** nous devons réaliser une jointure avec la table **produit** car c'est elle qui contient les informations des **produits**. Enfin nous ajoutons une condition (**WHERE**) pour n'afficher que les **produits** du **ticket** qui a comme valeur d' **id_ticket** : 2.

5 Calculer le **prix TTC** de chaque **produit** (prix *quantité) pour le **ticket 7**.

Correction :

```
SELECT nom_produit, (prix_produit*quantite) AS 'prix total' FROM ajouter  
INNER JOIN produit ON ajouter.id_produit = produit.id_produit WHERE  
ajouter.id_ticket = 7;
```

Explication :

Pour effectuer cette requête et récupérer uniquement les **produits** ainsi que le **prix TTC** de chacun nous devons réaliser une jointure avec la table **produit**. Pour retourner le **prix TTC** nous allons passer en paramètre des champs à afficher le calcul suivant (**prix_produit * quantite**) et utiliser un **alias** avec la commande **AS** pour renommer la colonne en **prix total**. Enfin nous allons ajouter la condition (**WHERE**) du ticket qui a comme **id_ticket** : 7.

6 Calculer combien de fois le produit Lait a été ajouté sur un ticket

Correction :

```
SELECT count(ajouter.id_produit) AS 'nbr de fois lait' FROM ajouter INNER  
JOIN produit ON ajouter.id_produit = produit.id_produit WHERE  
produit.nom_produit = 'lait';
```

Explication :

Pour effectuer cette requête nous allons introduire les **fonctions d'agrégation SQL**, Elles permettent de réaliser des calculs sur l'ensemble des **résultats retournés**. **Count** va renvoyer la somme du nombre de fois ou **id_produit** est retourné par la requête. Nous allons effectuer une jointure avec la table **produit** pour pouvoir rechercher par le **nom_produit** : lait.

Auteur :

Mathieu MITHRIDATE

Date création :

xx / xx / 20xx

Relu, validé & visé par :

☒ Jérôme CHRETIENNE
☒ Sophie POULAKOS
☒ Mathieu PARIS

Date révision :

xx / xx / 20xx



Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.

Explication correction TP bonus conception SQL

7 Afficher la liste des produits avec le nombre de fois où ils ont été ajouté sur un ticket

Correction :

```
SELECT nom_produit, count(ajouter.id_produit) AS 'nbr de fois ajouté' FROM  
ajouter INNER JOIN produit ON ajouter.id_produit = produit.id_produit GROUP  
BY ajouter.id_produit;
```

Explication :

Pour effectuer cette requête nous allons introduire en plus des **fonctions d'agrégation SQL**, la commande **GROUP BY** qui va modifier la façon dont la fonction **count** va fonctionner. Elle va regrouper les résultats par l'**id_produit**. Nous allons également faire une jointure avec la table **produit** pour pouvoir récupérer les informations des différents **produits**.

8 Calculer le montant TTC du ticket 11 somme (des produits*quantité)

Correction :

```
SELECT sum(prix_produit*quantite) AS 'montant ttc' FROM ajouter INNER JOIN  
produit ON ajouter.id_produit = produit.id_produit WHERE ajouter.id_ticket  
= 11;
```

Explication :

Pour effectuer cette requête nous allons utiliser une nouvelle **fonction d'agrégation SQL : sum** qui va effectuer la **somme** (addition) du calcul en paramètre. Nous allons également effectuer une jointure (**inner join**) avec la table **produit** pour récupérer le **prix_produit**. Nous ajoutons une condition (**WHERE**) pour rechercher uniquement par l'**id_ticket**.

9 Afficher la liste des vendeurs avec le nombre de fois où ils ont été ajouté sur un ticket.

Correction :

```
SELECT nom_personnel, prenom_personnel, count(ticket.id_personnel) AS 'Nbr  
de fois sur un ticket' FROM ticket INNER JOIN personnel ON  
ticket.id_personnel = personnel.id_personnel GROUP BY ticket.id_personnel;
```

Explication :

Pour effectuer cette requête nous allons devoir combiner les notions précédentes. On va utiliser la **fonction d'agrégation count** pour compter le nombre de fois où **id_personnel** est retourné par la requête. On va réaliser une jointure (**INNER JOIN**) avec la table **personnel** et grouper (**GROUP BY**) par **id_personnel** de la table **ticket**.

Auteur :

Mathieu MITHRIDATE

Date création :

xx / xx / 20xx

Relu, validé & visé par :

☒ Jérôme CHRETIENNE
☒ Sophie POULAKOS
☒ Mathieu PARIS

Date révision :

xx / xx / 20xx



Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.

Explication correction TP bonus conception SQL

10 Calculer le chiffre d'affaire des **produits** vendus au mois de **février 2023**.

Correction :

```
SELECT sum(quantite*prix_produit) AS 'chiffre affaire' FROM ajouter INNER JOIN produit ON ajouter.id_produit = produit.id_produit INNER JOIN join ticket ON ajouter.id_ticket = ticket.id_ticket WHERE month(date_ticket) = 2;
```

Explication :

Pour effectuer cette requête nous allons devoir utiliser toutes les notions précédentes (**fonction d'agrégation**) ainsi que 2 jointures (**INNER JOIN**) pour pouvoir récupérer les informations depuis les tables **ticket** et **produit**. Nous allons appeler une **fonction SQL** dans la condition (**WHERE**) pour chercher le mois de février : **month**. Elle prend en paramètre une **date** et retourne le **numéro du mois**, dans notre cas **2** pour février.

11 Calculer le chiffre d'affaire pour le vendeur **Bourhis Julie**.

Correction :

```
SELECT sum(quantite*prix_produit) AS 'Chiffre affaire', ajouter.id_ticket FROM ajouter
```

```
INNER JOIN produit ON ajouter.id_produit = produit.id_produit
```

```
INNER JOIN ticket ON ajouter.id_ticket = ticket.id_ticket
```

```
INNER JOIN personnel on ticket.id_personnel = personnel.id_personnel WHERE nom_personnel = 'Bourhis' AND prenom_personnel = 'Julie';
```

Explication :

Pour effectuer cette requête nous allons devoir utiliser toutes les notions précédentes (**fonction d'agrégation**) ainsi que 3 jointures (**INNER JOIN**) pour pouvoir récupérer les informations depuis les tables **produit**, **ticket** et **personnel**. Nous allons ajouter 2 conditions (**WHERE**) séparé par le mot clé **AND** pour rechercher uniquement les enregistrements dont le **nom_personnel** et **prenom_personnel** correspondent.

Auteur :

Mathieu MITHRIDATE

Date création :

xx / xx / 20xx

Relu, validé & visé par :

☒ Jérôme CHRETIENNE
☒ Sophie POULAKOS
☒ Mathieu PARIS

Date révision :

xx / xx / 20xx



Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.

Explication correction TP bonus conception SQL

12 Afficher la liste des **produits** avec le **chiffre d'affaire total** (produits vendus * quantité /ticket).

Correction :

```
SELECT nom_produit, description_produit, prix_produit, image_produit,  
sum(quantite*prix_produit) AS 'Chiffre affaire' FROM ajouter INNER JOIN  
produit ON ajouter.id_produit = produit.id_produit GROUP BY  
ajouter.id_produit;
```

Explication :

Pour effectuer cette requête nous allons devoir utiliser toutes les notions précédentes (**fonction d'agrégation**) ainsi que 1 jointure (**INNER JOIN**) pour pouvoir récupérer les informations depuis la tables **produit**. Les données sont regroupées par **id_produit**.

13 Afficher la liste des **vendeurs** avec leur nom, prénom, **chiffre d'affaire total** (produits *quantité / ticket).

Correction :

```
SELECT nom_personnel, prenom_personnel, sum(quantite*prix_produit) AS  
'Chiffre affaire vendeur'FROM ajouter INNER JOIN produit ON  
ajouter.id_produit = produit.id_produit INNER JOIN ticket ON  
ajouter.id_ticket = ticket.id_ticket INNER JOIN personnel ON  
ticket.id_personnel = personnel.id_personnel INNER JOIN poste ON  
personnel.id_poste = poste.id_poste WHERE poste.nom_poste = 'Vendeur' GROUP  
BY personnel.id_personnel;
```

Explication :

Pour effectuer cette requête nous allons devoir utiliser toutes les notions précédentes (**fonction d'agrégation**) ainsi que 4 jointures (**INNER JOIN**) pour pouvoir récupérer les informations depuis les tables **produit**, **ticket**, **personnel** et **poste**. Nous allons ajouter une condition (**WHERE**) par le **nom_poste** : Vendeur. Les données sont groupées par **id_personnel**.

Auteur :

Mathieu MITHRIDATE

Date création :

xx / xx / 20xx

Relu, validé & visé par :

☒ Jérôme CHRETIENNE
☒ Sophie POULAKOS
☒ Mathieu PARIS

Date révision :

xx / xx / 20xx



Toute reproduction, représentation, diffusion ou rediffusion, totale ou partielle, de ce document ou de son contenu par quelque procédé que ce soit est interdite sans l'autorisation expresse, écrite et préalable de l'ADRAR.