**Exercices 09**

**Mise en situation**

À partir du fichier de départ : **Exercices 09.01 - Base de données\_Etudiant.py**, répondez aux questions suivantes. Ensuite, assurez-vous de respecter les bonnes pratiques :

* Ajoutez un Docstring;
* Avoir des commentaires;
* Le code est bien indenté;
* Vos variables sont bien nommées;
* Le code est bien espacé;
* La longueur des lignes est adéquate;
* Aucun mot réservé n'a été utilisé.

Aussi, n’oubliez pas d’ajouter en commentaire votre : Nom, date et description du programme.

**Question 01 – Sélection MySQL**

Vous devez créer une fonction qui recevra un nom d’usager et qui affichera les résultats d’une requête :

* **Base de données** : MySQL;
* **Manipulations** :
  + Préparez la connexion;
  + Lisez la totalité de la table jeux\_video;
  + Affichez la totalité de la table dans la console.

**Question 02 – Connexion et déconnexion**

Pour chaque base de données, vous devez isoler la connexion et la déconnexion dans des fonctions.

* **Base de données** : MySQL, PostgreSQL et SQLite;
* **Manipulations** :
  + Les fonctions de connexion reçoivent le nom de la base de données ou du fichier en paramètre;
  + La fonction de déconnexion reçoit l’objet de connexion en paramètre;
  + Reproduisez la question 1 à l’aide d’une fonction pour chaque base de données et assurez-vous d’utiliser vos nouvelles fonctions de connexion et déconnexion.

**Question 03 – Module**

Vous devez isoler les fonctions de connexion et déconnexion dans un module externe et en faire son utilisation.

* **Base de données** : MySQL, PostgreSQL et SQLite;
* **Manipulations** :
  + Les 4 fonctions doivent se retrouver dans un seul module;
  + Ajoutez le Docstring à votre nouveau module;
  + Inscrivez, dans votre fichier de réponse, comment accéder à la documentation du Docstring de votre nouveau module;
  + Vous devez importer ce module dans votre fichier de réponse de l’exercice.
  + Dans une seule fonction, à l’aide de votre nouveau module, connectez-vous aux bases de données, affichez la totalité des jeux et déconnectez-vous aux bases de données.

**Question 04 – Insertion MySQL**

À partir du fichier **hockey.csv**, insérez les données dans une table.

* **Base de données** : MySQL;
* **Manipulations** :
  + Préparez votre base de données;
  + Réutilisez le module externe;
  + Faites toutes les manipulations nécessaires pour insérer les données dans votre nouvelle table;
  + Affichez les données que vous insérez à la console.

**Question 05 – Insertion PostgreSQL**

À partir du fichier **hockey.json**, insérez les données dans une table.

* **Base de données** : PostgreSQL;
* **Manipulations** :
  + Préparez votre base de données;
  + Réutilisez le module externe;
  + Faites toutes les manipulations nécessaires pour insérer les données dans votre nouvelle table;
  + Affichez les données que vous insérez à la console.

**Question 06 – Insertion SQLite**

À partir du fichier **hockey.txt**, insérez les données dans une table.

* **Base de données** : SQLite;
* **Manipulations** :
  + Préparez votre base de données;
  + Réutilisez le module externe;
  + Faites toutes les manipulations nécessaires pour insérer les données dans votre nouvelle table;
  + Affichez les données que vous insérez à la console.

**Question 07 – Modification MySQL**

À partir du fichier **hockey\_update.csv**, modifiez les données dans une table.

* **Base de données** : MySQL;
* **Manipulations** :
  + Réutilisez le module externe;
  + Faites toutes les manipulations nécessaires pour modifier les données dans la bonne table;
  + Affichez les données que vous modifiez à la console.

**Question 08 – Modification PostgreSQL**

À partir du fichier **hockey\_update.json**, modifiez les données dans une table.

* **Base de données** : PostgreSQL;
* **Manipulations** :
  + Créez une fonction de type « one-shot » qui modifiera la structure de votre table PostgreSQL pour accepter un nouveau champ qui provient de votre fichier.
  + Dans une autre fonction :
    - Réutilisez le module externe;
    - Faites toutes les manipulations nécessaires pour modifier les données dans la bonne table;
    - Affichez les données que vous modifiez à la console.

**Question 09 – Modification SQLite**

À partir du fichier **hockey\_update.txt**, modifiez les données dans une table.

* **Base de données** : SQLite;
* **Manipulations** :
  + Réutilisez le module externe;
  + Faites toutes les manipulations nécessaires pour modifier les données dans la bonne table;
  + Affichez les données que vous modifiez à la console.

**Question 10 – Suppression MySQL**

Supprimez le joueur qui a 155 passes de la table.

* **Base de données** : MySQL;
* **Manipulations** :
  + Réutilisez le module externe;
  + Faites toutes les manipulations nécessaires pour modifier les données dans la bonne table;
  + Affichez les données que vous supprimez à la console.

**Question 11 – Suppression PostgreSQL**

Supprimez les joueurs qui sont des défenseurs de la table.

* **Base de données** : PostgreSQL;
* **Manipulations** :
  + Réutilisez le module externe;
  + Faites toutes les manipulations nécessaires pour modifier les données dans la bonne table;
  + Affichez les données que vous supprimez à la console.

**Question 12 – Suppression SQLite**

Supprimez le joueur qui s’appelle « Paul Coffey » de la table.

* **Base de données** : SQLite;
* **Manipulations** :
  + Réutilisez le module externe;
  + Faites toutes les manipulations nécessaires pour modifier les données dans la bonne table;
  + Affichez les données que vous supprimez à la console.

**Question 13 – Gestionnaire de hockey**

Vous devez refaire le gestionnaire de hockey que vous avez fait en Semaine 02.

* **Base de données** : SQLite;
* **Manipulations** :
  + Utilisation d'un nouveau fichier SQLITE pour la création de l'équipe :
    - Créez un nouveau fichier de base de données : **equipe.sqlite**;
    - Créez une fonction, de style « one-shot », pour créer la structure de la table pour recevoir les données.
  + La gestion des joueurs se fait maintenant avec la BD **equipe**.
  + Vous devez optimiser votre code en utilisation des fonctions:
    - Utilisez les fonctions existantes dans votre module externe pour l'ouverture et fermeture de la BD;
    - Une fonction pour chaque option du menu.
    - Votre connexion et la déclaration de votre curseur se font dans le programme principal.
    - Vous devez créer une fonction pour faire la vérification de l'existence d'un joueur dans la BD. Ajoutez cette fonction dans le module externe et utilisez-la pour les options : 3, 4 et 6.