Sujet 3 : SQL, Python, HTML, JSON,...

Dans cette série d'exercices vous allez apprendre à exécuter des requêtes SQL directement depuis un langage de programmation (ex : Python) puis à traiter (aka algorithmique) le résultat obtenu (ex : génération de code HTML, export de données au format JSON, etc.).

Outre les manipulations "techniques", le principal objectif de ces exercices est de vous familiariser avec l'écosystème numérique dans lequel vous allez commencer à évoluer. Un tel écosystème est constitué de différentes technologies, d'une multitude de langages de programmation, de formats de données divers et variés, etc. En tant que futur diplômés d'un BUT RT ils vous incombera, dans votre future vie professionnelle, de choisir le bon mix technologique en fonction du projet à réaliser.

⋆ ⋆ ⋆

1 Requêtes SQL depuis Python

En vous aidant du support de cours du module R207 vous écrirez un programme Python qui exécute une requête SQL (un `select`) sur votre base de données SQLite et affiche les données du résultat "proprement" sur sa sortie standard.

Remarques : À vous de tester différentes solutions, différentes requêtes, etc. pour voir comment vous sont retournées les données du résultat du `select`. L'idée n'est pas de "bâcler" l'exercice pour obtenir un programme qui fonctionne à peu près sans que l'on sache réellement comment, mais plutôt de **bien comprendre les mécanismes mis en œuvre** !

2 Génération de code HTML depuis Python

Dans ce second exercice il vous est demandé de générer, depuis votre programme Python, les données du résultat de votre `select` au format HTML (ex : sous forme d'une table HTML).

Inutile dans cet exercice d'aller écrire dans un fichier depuis Python. Vous pouvez simplement vous contenter "d'afficher" sur la sortie standard de votre programme (via des `print`) le code HTML. Dans le terminal vous redirigerez la sortie standard (opérateur `>` du shell) vers un fichier `.html` que vous pourrez ensuite ouvrir dans votre navigateur web préféré.

```
python3 monProgramme.py > output.html
```

Remarques : Point d'explications ici sur le format HTML ; cela a déjà été abordé dans les modules précédents. Si ce n'est pas le cas, à vous d'utiliser Internet "**efficacement**" pour comprendre comment est construite une page HTML et comment est défini un tableau en HTML. Cette autonomie vis-à-vis de l'apprentissage des technologies fait également partie de votre formation. Tout au long de votre future carrière professionnelle, de nouvelles technologies émergeront et vous devrez vous y former sans pour autant retourner systématiquement sur les bancs de l'école...

3 Export de données en JSON depuis Python

L'idée est la même que dans l'exercice précédent, mais en exportant cette fois les données au format JSON. Là encore, vous trouverez les premiers éléments d'information sur JSON dans le support de cours du module R207 avant d'aller vous perdre dans les méandres d'Internet...

Il y a pour cela deux approches qu'il vous est demandé d'expérimenter :

- ▷ Procéder de la même façon que pour générer le code HTML à l'exercice précédent, c'est-à-dire en "forgeant" les données JSON entièrement "à la main", y compris les différents délimiteurs du format JSON.
- ▷ Utiliser les API JSON pour Python. Pour cela vous structurerez d'abord les données en Python (tableaux, listes, tuples,...) que vous convertirez ensuite en JSON. Il vous suffira ensuite de *stringnifier* les données JSON sur la sortie standard de votre programme Python.

4 Plus loin avec SQL en Python...

À vous d'aller consulter la documentation de l'API SQLite pour Python afin de réaliser un `insert into` dans votre base de données depuis un programme Python.

5 Import de données JSON vers Python

En utilisant les API JSON pour Python vous écrirez un programme qui charge les données JSON se trouvant dans un fichier pour ensuite les traiter (aka algo en Python) pour, par pur exemple, les injecter dans votre base de données via des `insert into` en SQL.