Atelier 1B– Librairie Gson / Klaxon

GSON et la magie de la réflexion Java ( Java Reflection )

Vous avez prouvé que vous êtes capables de récupérer des informations provenant d'un fichier JSON mais, dépendant de la complexité / profondeur de la structure du fichier, cela peut vite devenir compliqué à récupérer les données pour en créer des objets.

La librairie GSON (<https://github.com/google/gson> ) simplifie la lecture de la manière suivante :

On créer un modèle représentant les données qu'on espère trouver, sous forme de classes Java, ces classes peuvent contenir toutes ou certaines des variables représentées dans le fichier JSON.

Par les outils de la librairie GSON, on pourra extirper directement les objets dont le contenu des variables sont présents dans le fichier JSON à condition que le nom des variables d'instance correspondent aux noms des éléments dans le fichier JSON.

Donc en utilisent la technique de GSON on peux récupérer des objets si on a exactement le même nom que les objets on peux initialiser nos variable avec les éléments du fichier JASON (les instancier)….

**GSON a été conçu pour Java; on peut l’employer avec Kotlin mais des problèmes peuvent survenir avec la nullabilité. Nous emploierons la librairie Klaxon (**[**https://github.com/cbeust/klaxon**](https://github.com/cbeust/klaxon) **) , fonctionnant de façon similaire pour le langage Kotlin**

Exercice :

Faire une 3e version de la lecture du fichier JSON présent à :

<https://api.jsonbin.io/v3/b/67fe6a908a456b796689f63d?meta=false>

1. Utiliser une StringRequest comme dans la version 1
2. Ajouter la librairie pour utiliser Klaxon tel qu'indiqué dans le lien sur github ci-haut ( placer des parenthèses et des guillemets )
3. Ajouter une classe Produit et une classe ListeProduits qui contiendra une List de Produits car c'est ce que contient notre fichier JSON. Utilisez l’annotation @Json de la manière expliquée dans la documentation de Klaxon pour lier le nom de la liste dans le fichier avec celui ( trouvez-en un différent ) dans notre classe
4. Afin de ne pas faire des annotations à nouveau, assurez-vous que les noms des variables dans la classe Produit sont bien **nom** et **prix** car ce sont les noms utilisés dans le fichier JSON
5. Obtenez un objet ListeProduits en appelant la méthode parse de la classe Klaxon en lui passant la réponse du serveur et la classe déclarée au #3
6. Pour vous assurer que ça fonctionne, affichez d’abord le nombre d’éléments dans votre objet ListeProduits qui devrait être 3
7. Transformer votre liste afin de créer un SimpleAdapter et de gérer les clics sur les items de la liste complexe comme dans l’atelier 1.