

Fiche de TP1 - SpringBoot: JPA/Hibernate, Data REST

On souhaite réaliser une application simplifiée pour la gestion de la scolarité en utilisant SpringBoot.

Une formation a un numéro auto-increment, un nom et implique un ensemble d'étudiants. Un étudiant est défini par un numéro (auto-incrémenté), un nom, une date de naissance et il est inscrit dans une formation.

1. Créer un projet **SpringBoot** en utilisant **spring initializr** et choisir les dépendances suivantes :
 - i. Spring Web, Spring REST Repositories, Spring Data JPA, Mysql Driver, Lombok, DevTools
2. Créer les entités correspondantes en prenant en considération les contraintes suivantes :
 - i. Tous les noms sont obligatoires et d'une longueur de 20 caractères.
 - ii. La date de naissance est définie avec le format YYYY-MM-DD
3. Réaliser la couche DAO et peupler votre base de données.
 - i. **Query Derivation** :
 - Ajoutez une méthode qui recherche les étudiants par leurs noms
 - Ajoutez une méthode qui recherche les étudiants par leurs noms et leurs dates naissances (après une date donnée)
 - ii. **Annotated Query** :
 - Ajoutez une méthode qui retourne le nom d'un étudiant en indiquant son Id
 - Ajoutez une méthode qui modifie la date de naissance des étudiants en indiquant leurs noms
 - Ajoutez une méthode qui retourne le nombre d'étudiants par formation
4. Testez les différents Paths avec les différentes méthode http (GET, POST, PUT, PATCH, DELETE) générées par **@RepositoryRestResource**.
5. Créez une projection qui retourne pour chaque étudiant son nom et le nom de la formation dans laquelle est inscrit
6. Créez une API REST personnalisée en utilisant l'annotation **@RestController**. Cet API expose les Paths suivants :
 - i. GET <http://localhost:8081/api/etudiant/all>: retourne tous les étudiants
 - ii. GET <http://localhost:8081/api/etudiant/3> : retourne un étudiant par son **Id** (*PatchVariable*)
 - iii. GET <http://localhost:8081/api/etudiant?id=1> : retourne un étudiant par son **Id** (*RequestParam*)
 - iv. POST <http://localhost:8081/api/etudiant/?idf=2> : insert un étudiant dans la formation numéro **Idf**
 - v. POST <http://localhost:8081/api/etudiantp>: insert un étudiantd (le tout est définit dans le *RequestBody*)
 - vi. Put <http://localhost:8081/api/etudiant/1> : modifie un étudiant en indiquant son Id (le tout est définit dans le *RequestBody*)
 - vii. Patch <http://localhost:8081/api/etudiant/1>: modifie un étudiant en indiquant son Id (le tout est définit dans le *RequestBody*)