

## TP5 : Multi-projets

### 5.1 Mise en place de la structure projet

Créer un projet de type *basic*

Reprendre les sources fournis et les dézipper dans le répertoire parent

Le projet comprend 2 modules :

- Un module librairie qui contient du code utilitaire
- Une module application application monolithique qui dépend du module précédent

Inclure les 2 modules dans le projet parent

Exécuter `./gradlew projects`

Appliquer les plugins :

- `java`
- `org.springframework.boot`
- `io.spring.dependency-management`

Exécuter `./gradlew task`

Par défaut, le plugin `org.springframework.boot` crée une application exécutable, ce qui n'est pas souhaitable pour le module *library*

Pour le désactiver et activer la création de jar nécessaire à la dépendance :

```
bootJar {
    enabled = false
}
jar {
    enabled = true
}
```

### 5.2 Gestion des dépendances

Les 2 projets utilisent mavenCentral

Les 2 projets utilisent le framework SpringBoot version 2.1.6.RELEASE

Les 2 projets ont comme dépendances :

- `org.springframework.boot :spring-boot-starter-test` dans la configuration *test*

Le projet *library* a comme dépendance :

- *org.springframework.boot :spring-boot-starter* dans la configuration *implementation*

Le projet application a comme dépendance :

- Le module *library*
- *org.springframework.boot :spring-boot-starter-web*
- *org.springframework.boot :spring-boot-starter-actuator*

Une fois après avoir indiqué ces dépendances, construire :

- Le module *library*
- Le module *application*
- L'intégralité du projet

Exécuter l'application en exécutant la tâche *bootRun* sur le module *application*

### 5.3 Projet C++

Récupérer les sources fournies qui est un projet multi-module contenant une application dépendante du modules *utilities* qui dépend du module *list*

Ecrire les fichiers *gradle* pour construire ce projet