

# TP1 : Prise en main

## ***Auto configuration et Configuration de Debug***

Créer un nouveau projet SpringBoot avec un simple classe Main et sans ajout de modules

Dans la classe Main, utiliser la valeur de retour de `SpringApplication.run(Application.class, args);` pour afficher les beans Spring configurés.

Démarrer l'application en ligne de commande après avoir fait un build (*mvn package* ou *gradlew bootJar*)

Ajouter le module starter-web et réexécuter le programme

Ajouter les classes du TP précédent, la dépendance sur `spring-boot-configuration-processor` et l'activation de ***devtools***

Tester le redémarrage automatique lors du changement du code Java

## ***Configuration Externe***

Renommer le fichier *application.properties* en *application.yml*

Définir les propriétés suivantes :

- Valeur aléatoire dans le fichier *yml* affectée à une variable du contrôleur  
Afficher la valeur dans un point d'accès du contrôleur
- Dans *HelloProperties* (Ajouter des validations):
  - *hello.greeting* : Non vide
  - *hello.styleCase* : Upper, Lower ou Camel
  - *hello.position* : 0 ou 1

## ***Profil***

Définir un port différent pour le profil *prod*

Activer le profil :

- Via votre IDE
- Via la ligne de commande après avoir généré le *fat jar*

## ***Mode DEBUG et Configuration des traces***

Activer l'option ***-debug*** au démarrage

Modifier la configuration afin de générer un fichier de trace

Modifier le niveau de trace du logger *org.springframework.boot* à DEBUG sans l'option ***-debug***