

# Le framework Spring

## Module 2 – Introduction à Spring Boot



# Objectifs

- Qu'est ce que Spring Boot ?
- Savoir créer un projet en utilisant Spring Boot

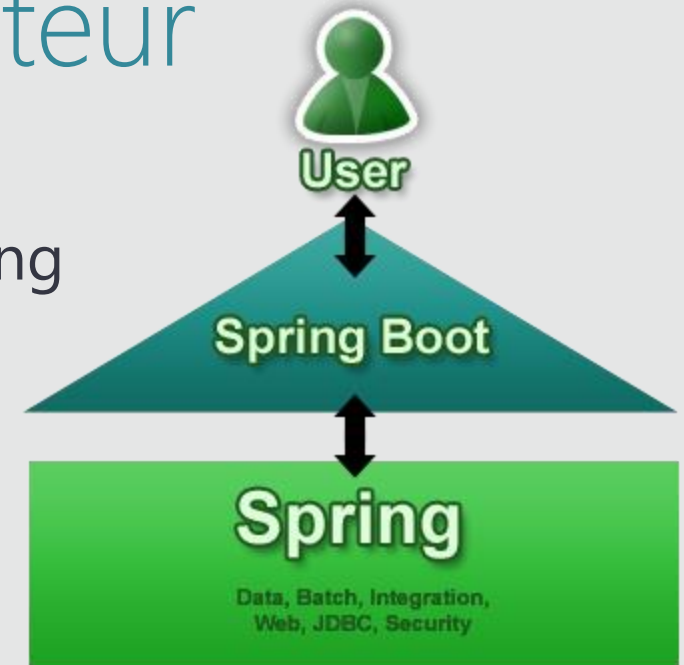
# Spring Boot

## Introduction

- Le problème :
  - Spring est un framework proposant de très nombreuses fonctionnalités
    - Web, Sécurité, batch, accès aux données, ...
  - Mais, sa configuration peut s'avérer complexe et fastidieuse
- La solution : **Spring Boot**, est un outil facilitant la configuration Spring grâce à :
  - L'auto-configuration
  - Les starters
  - L'approche client lourd

# Un seul Point d'entrée pour l'utilisateur

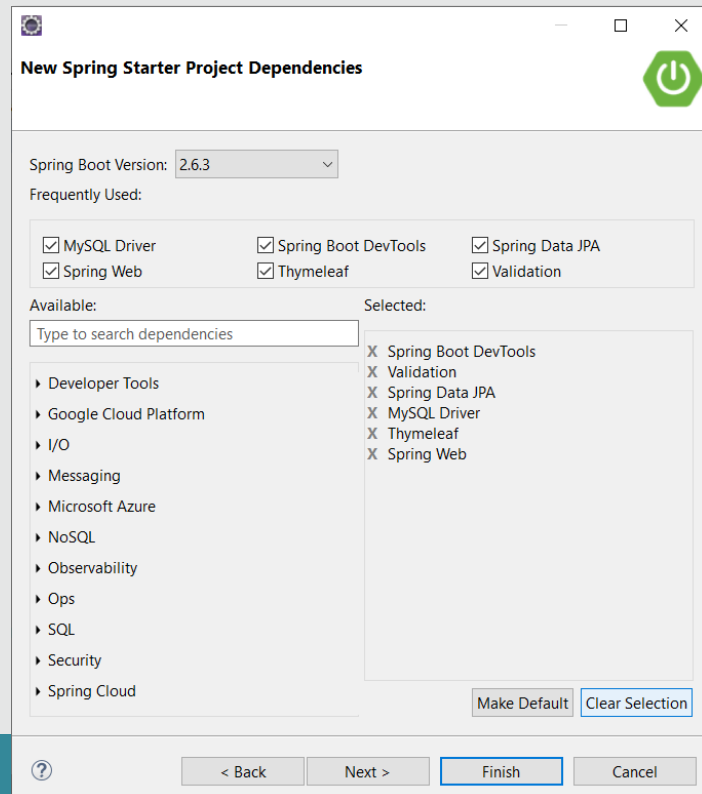
- Plus nécessaire d'intégrer tous les modules de Spring
- Les dépendances sont déjà intégrées
  - starters
- Utilisation de l'auto-configuration
  - @EnableAutoConfiguration
- Les serveurs Tomcat/Jetty sont embarqués
- Plus besoin de XML
- Application packagée en JAR orientée pour les micro-services



# Spring Boot

## Les Starters

- Permet de configurer les dépendances d'un projet
- Les dépendances sont organisées par catégories appelées starters
- Possibilité de travailler avec Maven ou Gradle



```
plugins {  
    id 'org.springframework.boot' version '2.6.4'  
    id 'io.spring.dependency-management' version '1.0.11.RELEASE'  
    id 'java'  
}  
  
group = 'fr.eni'  
version = '1'  
sourceCompatibility = '11'  
  
repositories {  
    mavenCentral()  
}  
  
dependencies {  
    implementation 'org.springframework.boot:spring-boot-starter-data-jpa'  
    implementation 'org.springframework.boot:spring-boot-starter-thymeleaf'  
    implementation 'org.springframework.boot:spring-boot-starter-validation'  
    implementation 'org.springframework.boot:spring-boot-starter-web'  
    developmentOnly 'org.springframework.boot:spring-boot-devtools'  
    runtimeOnly 'mysql:mysql-connector-java'  
    testImplementation 'org.springframework.boot:spring-boot-starter-test'  
}}
```

Spring Boot

# Création automatique d'une application exécutable

```
package fr.eni.demosWeb;
```

```
import org.springframework.boot.SpringApplication;
```

```
import org.springframework.boot.autoconfigure.SpringBootApplication;
```

```
@SpringBootApplication
```

```
public class PremierProjetApplication {
```

```
    public static void main(String[] args) {  
        SpringApplication.run(  
            PremierProjetApplication.class, args);  
    }  
}
```

```
@SpringBootApplication
```

```
@EnableAutoConfiguration
```

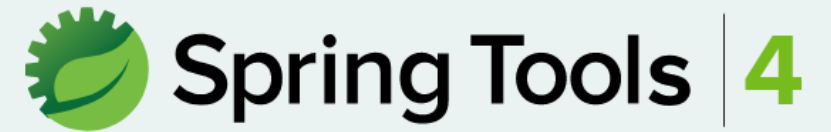
```
@ComponentScan
```

```
public @interface SpringBootApplication {
```

Exécutable directement sur  
le tomcat intégré!

# Outils préconisés par Spring

- Spring Tools : Plugin pour les IDEs (Eclipse, Visual Studio, Atom IDE)
- Spring initializr : <https://start.spring.io/>



The screenshot shows the Spring Initializr web application. At the top left is a hamburger menu icon and the 'spring initializr' logo. The main content area is divided into three sections: 'Project', 'Language', and 'Dependencies'. In the 'Project' section, 'Maven Project' is selected. In the 'Language' section, 'Java' is selected. In the 'Dependencies' section, there is a button 'ADD DEPENDENCIES... CTRL + B' and the text 'No dependency selected'. Below these sections is the 'Spring Boot' version selection area, where '2.6.3' is selected. At the bottom is the 'Project Metadata' section with input fields for 'Group' (com.example), 'Artifact' (demo), 'Name' (demo), and 'Description' (Demo project for Spring Boot). At the very bottom are three buttons: 'GENERATE CTRL + G', 'EXPLORE CTRL + SPACE', and 'SHARE...'.

Spring Boot

# Installation de Gradle

## Démonstration





Spring Boot

# Installation de Spring Tools

## Démonstration



Spring Boot

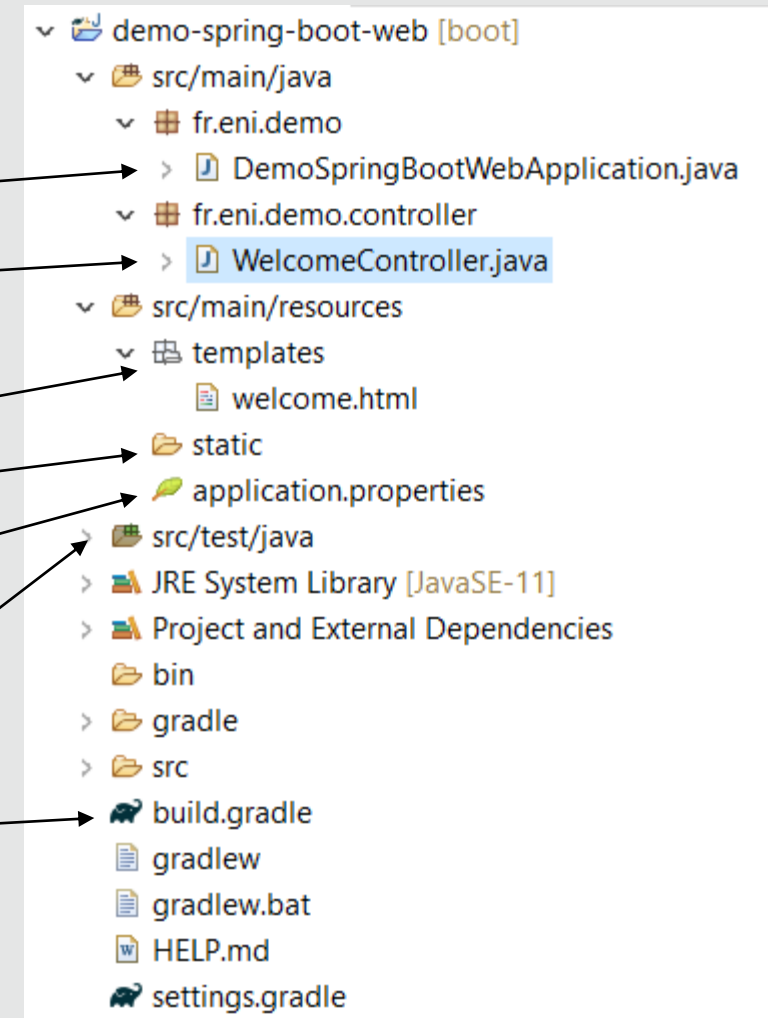
# Première application Web avec Spring Boot

## Démonstration



# Architecture du projet

- La classe Main pour exécuter Spring Boot
- La classe du contrôleur de la démonstration
- Dans le répertoire « templates », ajouter les templates de page (Thymeleaf)
- Dans « static » mettre les fichiers de CSS, JS
- Fichier de propriétés pour configurer le projet
- Il y a un package de tests unitaires avec une classe par défaut
- Fichier de configuration Gradle du projet



# Spring Boot

## Conclusion

- Spring Boot permet
  - Une mise en place simplifiée d'une application Spring
  - De diminuer la configuration
- Spring Boot peut remplacer un socle d'Entreprise
  - Plus nécessaire d'avoir un serveur applicatif
  - Fabrication d'un JAR et non d'un WAR