

# Tutoriel : Génération d'entités JPA avec Spring Boot + Hibernate Tools

Objectif : Connecter un projet Spring Boot à une base PostgreSQL, et générer automatiquement les entités JPA à partir des tables existantes.

## Prérequis

- Java 17 ou 21
- PostgreSQL lancé localement
- Une base avec des tables
- Maven installé

## Structure du projet

```
backend/  
??? pom.xml  
??? src/  
?   ??? main/  
?       ??? java/  
?       ?   ??? com/example/util/GenerateEntities.java  
?       ??? resources/  
?           ??? hibernate.cfg.xml  
?           ??? reveng.xml
```

## pom.xml (complet)

```
<project xmlns="http://maven.apache.org/POM/4.0.0"  
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
    xsi:schemaLocation="http://maven.apache.org/POM/4.0.0  
        http://maven.apache.org/xsd/maven-4.0.0.xsd">  
    <modelVersion>4.0.0</modelVersion>  
    <groupId>com.example</groupId>  
    <artifactId>gestion-notes</artifactId>  
    <version>0.0.1-SNAPSHOT</version>  
    <packaging>jar</packaging>  
  
    <name>gestion-notes</name>  
  
    <parent>  
        <groupId>org.springframework.boot</groupId>  
        <artifactId>spring-boot-starter-parent</artifactId>  
        <version>3.2.2</version>  
        <relativePath/>  
    </parent>  
  
    <properties>  
        <java.version>21</java.version>  
    </properties>  
  
    <dependencies>  
        <dependency>  
            <groupId>org.springframework.boot</groupId>  
            <artifactId>spring-boot-starter</artifactId>
```

# Tutoriel : Génération d'entités JPA avec Spring Boot + Hibernate Tools

```
</dependency>
<dependency>
  <groupId>org.hibernate</groupId>
  <artifactId>hibernate-tools</artifactId>
  <version>5.1.2.Final</version>
</dependency>
<dependency>
  <groupId>org.postgresql</groupId>
  <artifactId>postgresql</artifactId>
  <version>42.7.2</version>
</dependency>
<dependency>
  <groupId>org.jboss.logging</groupId>
  <artifactId>jboss-logging</artifactId>
  <version>3.4.3.Final</version>
</dependency>
<dependency>
  <groupId>javax.xml.bind</groupId>
  <artifactId>jaxb-api</artifactId>
  <version>2.3.1</version>
</dependency>
<dependency>
  <groupId>org.glassfish.jaxb</groupId>
  <artifactId>jaxb-runtime</artifactId>
  <version>2.3.1</version>
</dependency>
</dependencies>

<build>
  <plugins>
    <plugin>
      <groupId>org.apache.maven.plugins</groupId>
      <artifactId>maven-compiler-plugin</artifactId>
      <version>3.11.0</version>
      <configuration>
        <source>21</source>
        <target>21</target>
      </configuration>
    </plugin>
    <plugin>
      <groupId>org.codehaus.mojo</groupId>
      <artifactId>exec-maven-plugin</artifactId>
      <version>3.1.0</version>
    </plugin>
  </plugins>
</build>
</project>
```

## hibernate.cfg.xml

```
<hibernate-configuration>
```

# Tutoriel : Génération d'entités JPA avec Spring Boot + Hibernate Tools

```
<session-factory>
    <property name="hibernate.connection.driver_class">org.postgresql.Driver</property>
    <property name="hibernate.connection.url">jdbc:postgresql://localhost:5432/gestion_notes</property>
    <property name="hibernate.connection.username">postgres</property>
    <property name="hibernate.connection.password">postgres</property>
    <property name="hibernate.dialect">org.hibernate.dialect.PostgreSQLDialect</property>
</session-factory>
</hibernate-configuration>
```

## reveng.xml

```
<hibernate-reverse-engineering>
    <schema-selection match-catalog="gestion_notes" match-schema="public"/>
    <table-filter match-name=".*"/>
</hibernate-reverse-engineering>
```

## GenerateEntities.java

```
package com.example.util;

import org.hibernate.cfg.JDBCMetaDataConfiguration;
import org.hibernate.cfg.reveng.DefaultReverseEngineeringStrategy;
import org.hibernate.cfg.reveng.ReverseEngineeringStrategy;
import org.hibernate.tool.hbm2x.Exporter;
import org.hibernate.tool.hbm2x.POJOExporter;

import java.io.File;

public class GenerateEntities {
    public static void main(String[] args) {
        try {
            JDBCMetaDataConfiguration cfg = new JDBCMetaDataConfiguration();
            ReverseEngineeringStrategy strategy = new DefaultReverseEngineeringStrategy();
            cfg.setReverseEngineeringStrategy(strategy);
            cfg.configure("hibernate.cfg.xml");
            cfg.readFromJDBC();
            File outputDir = new File("src/main/java/com/example/entity");
            outputDir.mkdirs();
            Exporter exporter = new POJOExporter(cfg, outputDir);
            exporter.start();
            System.out.println("Entites generees avec succes !");
        } catch (Exception e) {
            System.err.println("Erreur :");
            e.printStackTrace();
        }
    }
}
```

## Commandes à exécuter

```
mvn dependency:copy-dependencies
mvn clean compile
```

## Tutoriel : Génération d'entités JPA avec Spring Boot + Hibernate Tools

```
java -cp "target/classes;target/dependency/*" com.example.util.GenerateEntities
```

### Résultat

Les entités .java sont générées dans : src/main/java/com/example/entity/