

Migraciones de Base de Datos con Flyway en Spring Boot

Aquí tienes una guía concisa y práctica sobre cómo usar Flyway con una aplicación Spring Boot para gestionar tu esquema de base de datos:

¿Qué es Flyway?

Flyway es una herramienta de migración de bases de datos que te ayuda a controlar la versión de tu esquema de base de datos. Aplica scripts SQL (migraciones) en un orden específico para evolucionar tu base de datos, lo que la hace ideal para proyectos Spring Boot donde la consistencia y la automatización son clave.

Paso 1: Añadir la dependencia de Flyway

Para empezar, añade Flyway a tu proyecto. Si estás usando **Maven**, actualiza tu `pom.xml`:

```
<dependency>
  <groupId>org.flywaydb</groupId>
  <artifactId>flyway-core</artifactId>
</dependency>
```

Para **Gradle**, añade esto a tu `build.gradle`:

```
implementation 'org.flywaydb:flyway-core'
```

Esto incorpora Flyway en tu aplicación Spring Boot.

Paso 2: Configurar la conexión a la base de datos

Flyway necesita saber qué base de datos gestionar. Define tu conexión a la base de datos en `application.properties` (o `application.yml`). Por ejemplo, con PostgreSQL:

```
spring.datasource.url=jdbc:postgresql://localhost:5432/mydb
spring.datasource.username=myuser
spring.datasource.password=mypassword
```

Spring Boot configura automáticamente Flyway para usar esta fuente de datos, por lo que no se necesita ninguna configuración adicional aquí.

Paso 3: Crear scripts de migración

Flyway utiliza scripts SQL versionados para actualizar tu base de datos. Coloca estos scripts en el directorio predeterminado: `src/main/resources/db/migration`. Cada script debe seguir este patrón de nomenclatura: `V<version>__<description>.sql`. El número de versión asegura que las migraciones se ejecuten en orden.

Por ejemplo, crea un archivo llamado `V1__create_users_table.sql`:

```
CREATE TABLE users (  
    id SERIAL PRIMARY KEY,  
    username VARCHAR(255) NOT NULL,  
    email VARCHAR(255) NOT NULL  
);
```

Guárdalo en `src/main/resources/db/migration`.

Paso 4: Ejecutar tu aplicación

Inicia tu aplicación Spring Boot (por ejemplo, mediante `mvn spring-boot:run` o tu IDE). Flyway: - Verificará la base de datos para un historial de migraciones. - Aplicará cualquier migración pendiente (como `V1__create_users_table.sql`) en secuencia.

Esto sucede automáticamente al iniciar gracias a la integración de Spring Boot, sin necesidad de intervención manual.

Paso 5: Personalizar Flyway (Opcional)

Puedes ajustar el comportamiento de Flyway con propiedades en `application.properties`. Aquí tienes algunas opciones comunes:

```
spring.flyway.locations=classpath:db/migration # Dónde encontrar los scripts  
spring.flyway.baseline-on-migrate=true         # Manejar bases de datos existentes  
spring.flyway.enabled=true                     # Activar/desactivar Flyway
```

Estas opciones te permiten adaptar Flyway a las necesidades de tu proyecto.

Características avanzadas

Flyway ofrece más que migraciones básicas: - **Migraciones basadas en Java**: Escribe migraciones en Java para lógica compleja. - **Migraciones repetibles**: Usa `R__<description>.sql` para scripts que se vuelvan a ejecutar cuando se cambien (por ejemplo, vistas). - **Deshacer migraciones**: Revertir cambios (requiere Flyway Pro).

Consulta la documentación de Flyway para una exploración más profunda.

Resumen

Para usar Flyway en Spring Boot: 1. Añade la dependencia de Flyway a tu proyecto. 2. Configura tu conexión a la base de datos en `application.properties`. 3. Escribe scripts SQL versionados en `src/main/resources/db/migration`. 4. Lanza tu aplicación—Flyway aplica las migraciones automáticamente.

Flyway simplifica la gestión del esquema de la base de datos con versionado y automatización, complementando perfectamente la experiencia de desarrollo simplificada de Spring Boot.