



Residência
em Software

Módulo Programação JAVA (Avançado)

MÊS 01



INSTITUIÇÃO EXECUTORA



UESC

COORDENADORA



APOIO

MINISTÉRIO DA
CIÊNCIA, TECNOLOGIA
E INOVAÇÃO



Migração de banco de dados com **Flyway** e **Spring Boot**

O que é o Flyway?

O Flyway é uma ferramenta que ajuda a **rastrear e gerenciar migrações de banco de dados**. Ele permite que você migre facilmente seu banco de dados de uma versão para outra, reduzindo o **risco de perda ou corrupção de dados**.



Migração de banco de dados com **Flyway** e **Spring Boot**

Por que usar uma ferramenta para versionar banco de dados?

Para responder a perguntas, como:

- Em que estado está o banco de dados nesse ambiente?
- Um script específico já foi aplicado ou não?
- Esse hotfix na produção foi implantado em outros ambientes posteriormente?
- Como posso configurar uma nova instância de banco de dados para um estado específico ou mais recente?

Executar verificações como: checar os scripts SQL para descobrir se alguém adicionou uma coluna, modificou um procedimento armazenado ou coisas semelhantes em todos os nossos ambientes (dev, teste, staging, produção).

Migração de banco de dados com **Flyway** e **Spring Boot**

Por que usar uma ferramenta para versionar banco de dados?

- Criar um banco de dados do zero.
- Ter uma única fonte para a versão do estado do banco de dados.
- Ter um estado reproduzível do banco de dados em ambientes locais e remotos.
- Automatizar a implantação de alterações no banco de dados, minimizando os erros humanos.

Migração de banco de dados com **Flyway** e **Spring Boot**

Alternativas para Versionamento de Banco de Dados em ambiente JAVA



Flyway

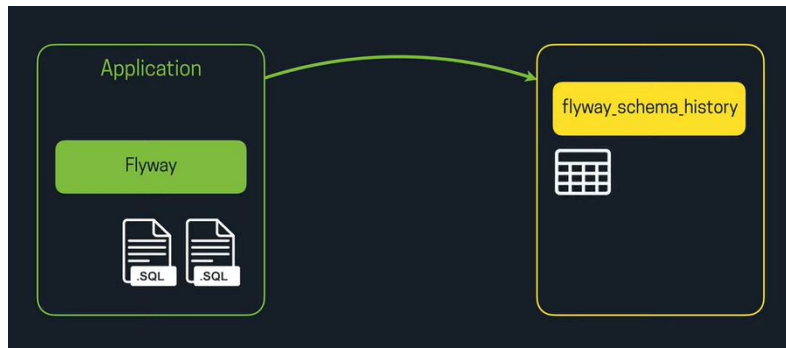


Liquibase

Migração de banco de dados com **Flyway** e **Spring Boot**

Como funciona o Flyway?

O Flyway tenta **localizar migrações fornecidas pelo usuário** no sistema de arquivos e no classpath Java. Por padrão, ele **carrega todos os arquivos** na pasta dentro do classpath que estão conforme a convenção de nomenclatura configurada.



Migração de banco de dados com **Flyway** e **Spring Boot**

Ex. Aplicações do Flyway

As migrações com versão são normalmente usadas para:

- Alterações DDL (instruções **CREATE/ALTER/DROP** para **TABLES, VIEWS, TRIGGERS, SEQUENCES, INDEX...**)
- Alterações simples de dados de referência (CRUD em tabelas de dados de referência)
- Alterações simples de dados em massa (CRUD em tabelas de dados regulares)

Migração de banco de dados com **Flyway** e **Spring Boot**

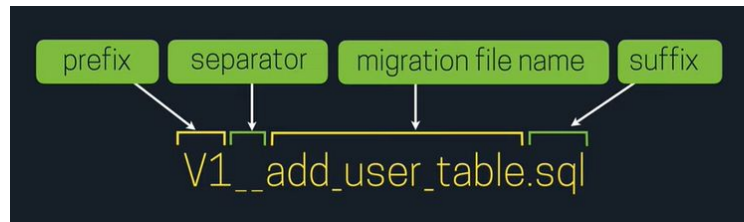
Versioned Migrations

Essas migrações possuem uma **versão única** sendo aplicadas exatamente uma vez.

São usadas para mudanças estruturais, como **criação de tabelas, alteração de colunas ou adição de índices**.

Cada script de migração versionado corresponde a uma versão específica do seu aplicativo.

Exemplo: Criando uma tabela de usuários.



Migração de banco de dados com **Flyway** e **Spring Boot**

Repeatable Migrations

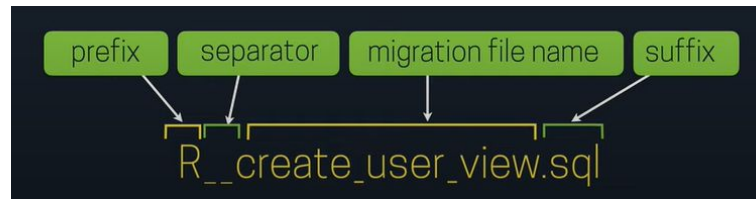
Migrações repetíveis não possuem versão. Em vez disso, eles são (re)aplicados sempre que sua soma de verificação é alterada.

Normalmente usado para:

- (Re)criar views, procedures, functions ou pacotes.
- Reinserções de dados de referência em massa.

Eles são aplicados em ordem alfabética.

Exemplo: Gerenciando uma view da tabela usuário.



Migração de banco de dados com **Flyway** e **Spring Boot**

Propriedades Spring-Flyway

```
spring.flyway.sql-migration-prefix=V  
spring.flyway.repeatable-sql-migration-prefix=R  
spring.flyway.sql-migration-separator=__  
spring.flyway.sql-migration-suffixes=.sql
```

#O Flyway pode ser configurado para usar DataSource diferente em vez de usar DataSource
primário do aplicativo Spring Boot usando as propriedades abaixo:

```
spring.flyway.url= jdbc:postgresql://localhost:5432/app_test_flyway  
spring.flyway.user= postgres  
spring.flyway.password= SenhaDB
```

source: resources/application.properties

Migração de banco de dados com **Flyway** e **Spring Boot**

Migração baseada em SQL

```
CREATE TABLE test_user(  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    username VARCHAR(255) NOT NULL UNIQUE,  
    first_name VARCHAR(255) NOT NULL,  
    last_name VARCHAR(255) NOT NULL,  
);
```

Migração de banco de dados com Flyway e Spring Boot

Migração baseada em Java

```
1  package db.migration;
2
3  import org.flywaydb.core.api.migration.BaseJavaMigration;
4  import org.flywaydb.core.api.migration.Context;
5  import org.springframework.jdbc.core.JdbcTemplate;
6  import org.springframework.jdbc.datasource.SingleConnectionDataSource;
7
8  /**
9   * Example of a Java-based migration using Spring {@link JdbcTemplate}.
10  */
11  public class V2_InsertRandomUsers extends BaseJavaMigration {
12
13      public void migrate(Context context) {
14
15          final JdbcTemplate jdbcTemplate = new JdbcTemplate(new SingleConnectionDataSource(context.getConnection(), true));
16
17          // Create 10 random users
18          for (int i = 1; i <= 10; i++) {
19              jdbcTemplate.execute(String.format("insert into test_user(username, first_name, last_name) "
20                  + "values('%d@reflectoring.io', 'Elvis_%d', 'Presley_%d')", i, i, i));
21          }
22      }
23  }
```

Migração de banco de dados com **Flyway** e **Spring Boot**

Migração como parte de um processo de CI/CD

```
pipeline {  
  agent any  
  
  stages {  
    checkout scm  
  
    stage('Apply Database Migrations') {  
      steps {  
        script {  
          if (isUnix()) {  
            sh '/gradlew flywayMigrate --info'  
          } else {  
            bat 'gradlew.bat flywayMigrate --info'  
          }  
        }  
      }  
    }  
  }  
}
```

Um pipeline Jenkins usando o plugin Flyway e Gradle para executar as migrações de banco de dados

Migração de banco de dados com **Flyway** e **Spring Boot**

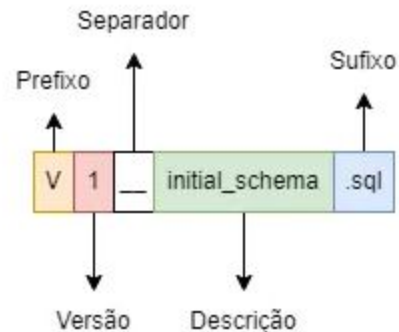
Padrão de nomenclatura dos scripts

Prefixo: deve iniciar com V em maiúsculo

Versão: número da versão, incremental e separada por . ou _ ex: 1.1, 1_1, 2, 2.2

Separador: __ (2 underscores)

Descrição: breve descrição do que esta sendo alterado



Padrão de nomenclatura do Flyway

Migração de banco de dados com **Flyway** e **Spring Boot**

Padrão de nomenclatura dos scripts

V1.0__create_user.sql

V1_1__create_user.sql

V001__create_user.sql

V2__InsertRandomUsers.java

R__Create_View_TopPosts.sql

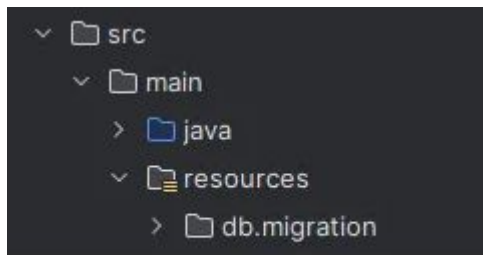
U1__Undo_Init.sql

V202401252036__update_users.sql — Também podemos colocar a data inversa com hora e minuto, fazem isso para evitar que diferentes desenvolvedores subam um script com a mesma versão.



Migração de banco de dados com **Flyway** e **Spring Boot**

Padrão de nomenclatura dos scripts



Criando o diretório dos scripts

Ex. Caminho: `src/main/resources/db/migration/V001__initial_creation.sql`

Migração de banco de dados com **Flyway** e **Spring Boot**

Integrando Flyway com Spring Boot

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

Adicionando dependência do Flyway

Migração de banco de dados com **Flyway** e **Spring Boot**

Adicionar o Flyway a projetos existentes

- Faça backup dos dados existentes
- Crie script SQL do banco de dados existente
- Set a versão de script preparado como linha de base

Migração de banco de dados com **Flyway** e **Spring Boot**

Adicionar o Flyway a projetos existentes

No aplicativo Spring Boot, as seguintes propriedades definem a linha de base:

```
spring.flyway.baseline-on-migrate=true # enables the baselining  
spring.flyway.baseline-version=1 # sets baseline version
```

Para introduzir o Flyway ao banco de dados existente precisa usar o conceito chamado Linha de Base. Primeiro, precisamos fazer a linha de base das alterações existentes no banco de dados para uma versão específica.

A linha de base fará com que o Migrate ignore todas as migrações até a versão de linha de base, inclusive. As migrações mais recentes serão aplicadas normalmente.

Migração de banco de dados com **Flyway** e **Spring Boot**

Detectando alterações em migrações existentes

Depois de aplicar as migrações, o arquivo não deve ser alterado, pois o Flyway mantém o controle das migrações aplicadas com base nas versões e na soma de verificação.

Se introduzimos alguma alteração em um arquivo de migração existente o aplicativo deverá lançar uma exceção:

```
org.flywaydb.core.api.exception.FlywayValidateException: Validate failed:  
Migrations have failed validation Migration checksum mismatch for migration  
version 2
```

Migração de banco de dados com **Flyway** e **Spring Boot**

Flyway - Principais comandos

Maven goals	Description
<code>flyway:baseline</code>	Baselines an existing database, excluding all migrations up to and including baselineVersion.
<code>flyway:clean</code>	Drops all database objects (tables, views, procedures, triggers, ...) in the configured schemas. The schemas are cleaned in the order specified by the schemas property..
<code>flyway:info</code>	Retrieves the complete information about the migrations including applied, pending and current migrations with details and status
<code>flyway:migrate</code>	Triggers the migration of the configured database to the latest version.
<code>flyway:repair</code>	Repairs the Flyway schema history table. This will remove any failed migrations on databases without DDL transactions
<code>flyway:undo</code>	Undoes the most recently applied versioned migration. Flyway teams only.
<code>flyway:validate</code>	Validate applied migrations against resolved ones on the classpath. This detect accidental changes that may prevent the schema(s) from being recreated exactly.

Migração de banco de dados com **Flyway** e **Spring Boot**

Adicionando Flyway ao Projeto

Etapas 1: pom.xml

- <https://gist.github.com/rog3r/d6458a8a6a381d25a1930da6db397f88>

Etapas 2: db/migration

- <https://gist.github.com/rog3r/86a1e1952e242a47242612e9d240eebc>
- <https://gist.github.com/rog3r/955760a4fed02011f2aec37e85c4ac3b>
- **Flyway Repo:**
- <https://github.com/flyway/flyway/>
- **Flyway Migrations:**
- <https://github.com/ReachBinduKumar/43-flyway-tutorial/tree/main/migrations>



Residência
em Software



Contato

rogerio.jesus@cepedi.org.br

<https://moodle.residenciatic18.cepedi.org.br/>