Hibernate & Spring Data JPA

Beginner to Guru

Introduction to Flyway

What is a Migration?

- Migrations are the process of moving programming code from one system to another
  - This is fairly large and complex topic of maintaining computer applications
- Database Migrations typically need to occur prior to, or in conjunction with application code
  - Can lead to run time errors if database does not match what is expected
- Database migrations are a very important part of the process of moving your application code to production
- Keep in mind, in larger organizations, you as the developer will NOT be doing the migration

Why Use a Migration Tool?

- Hibernate can manage my schema fine, why use a migration tool?
- Managing many environments and databases:
  - Dev (H2), CI/CD, QA, UAT, Production
- Development and CI/CD databases are easy, just rebuild each time
- QA, UAT, Production are permanent databases
  - What state are they in?
  - Has a script been applied?
  - How to create a new database to a release?

Why Use a Migration Tool?

- Database Migration tools can:
  - Create a new database
  - Hold history of migrations
  - Have a reproducible state of the database
  - Help manage changes being applied to numerous database instances
- Popular Open Source Database Migration Tools (Not a complete list):
  - Liquibase
  - Flyway

Liquibase and Flyway

- Common Features:
  - Command Line Tools
  - Integration with Maven and Gradle
  - Spring Boot Integration
  - Use script files which can be versioned and tracked
  - Use database table to track changes
  - Have commercial support

Liquibase / Flyway - Spring Boot Integration

- Spring Boot offers support for both Liquibase and Flyway
- Fundamentally Spring Boot will apply change sets
- Spring Boot's support is narrow in scope
- Both tools have additional capabilities available from the command line or build tool plugins
- The Spring Boot integration is convenient for migrations
  - Both tools may be used independently of Spring Boot

Liquibase vs Flyway

- Liquibase and Flyway are very similar in terms of functionality
- Share same concepts, slightly different terminology
- Liquibase supports change scrips in SQL, XML, YAML, and JSON
  - XML, YAML and JSON abstract SQL, which may be beneficial for different DB technologies
- Flyway supports SQL and Java only
- Liquibase is a larger and more robust product
- Flyway seems to have more popularity
- Both are mature and widely used

Liquibase vs Flyway - Which to Use?

- Liquibase is probably a better solution for large enterprises with complex environments
- Flyway is good for 90% of applications which don't need the additional capabilities
- Recommendation:
  - If one or the other is being used in the organization, use it
  - If in doubt, do your own research on each option
  - John's preference is Flyway - simple and easy to use

Flyway

Flyway Commands

- Migrate - Migrate to latest version
- Clean - Drops all database objects - NOT FOR PRODUCTION USE
- Info - Prints info about migrations
- Validate - Validates applied migrations against available
- Undo - Reverts most recently applied migration
- Baseline - Baselines an existing database
- Repair - Used to fix problems with schema history table

Running Flyway

- Command Line (CLI) - CLI available for Windows, MacOS, and Linux
  - Not covered in this course
- Maven / Gradle Plugins
  - Not covered in this course
- Spring Boot - Will run Flyway on startup to update configured database to latest changeset.

Our Next Steps

- Configure Spring Boot Support for Flyway
- Alter existing table with Flyway
- Add a new table with Flyway
- NOTE: Source code will start with end of Hibernate with MySQL section
  - same starting point as Liquibase section of course

Why Not Baseline?

- Baseline is for introducing Flyway to an existing database schema
  - Assumes database tables and objects will be there
- We wish to start from an empty database

SPRING FRAMEWORK

GURU