** BD Migration by Flyway**



**CONTENTS**

1. Flyway ............................................................................................................................3
   1. How flyway works ..................................................................................................3
2. Migration .......................................................................................................................4
   1. Need of DB Migration ............................................................................................4
3. Flyway commands .........................................................................................................4
   1. Migrate ..................................................................................................................5
   2. Info ........................................................................................................................5
   3. Repair ....................................................................................................................6
   4. Validate .................................................................................................................6
   5. Undo ......................................................................................................................6
   6. Baseline .................................................................................................................7
   7. Clean .....................................................................................................................7
4. Versioned Migration .....................................................................................................8
   1. How to perform versioned migration ...................................................................9
5. Repeatable Migration ..................................................................................................9
6. 5.1 How to perform repeatable migration ................................................................10

5.2 Modifying migration ............................................................................................10

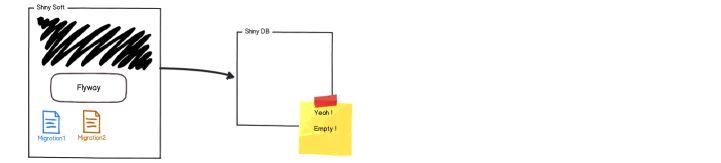
1. Undo Migration ...........................................................................................................11
   1. How to perform undo migration .........................................................................11
2. Dealing with flyway configuration file .........................................................................11
3. Migration Fails..............................................................................................................12
4. Skipping a Particular Migration ...................................................................................12

**Flyway:**

Flyway is open source database migration tool. It is based around 7 basic commands: Migrate, Clean, Info, Validate, Undo, Baseline and Repair. Flyway can be written in Sequential Query Language or Java for advanced data transformation.

**How Flyway Woks:**

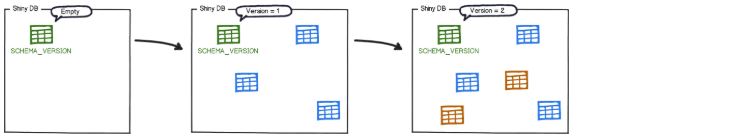
Suppose I have a scenario is when you point flyway to an empty database, it will try to relocate its schema history table. As the database is empty flyway would not find it and will create it instead.



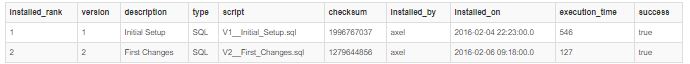
Suppose now you have a database with single empty table called flyway\_schema \_history by default.



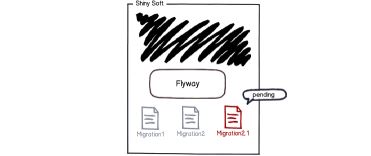
This table will be used to track the state of the database. Immediately after words flyway will begin scanning the filesystem or class path of the application for migrations. They can be written in either SQL or JAVA. The migrations are then sorted based on their version number and applied in order.



After each migration get applied the schema history table is updated accordingly.

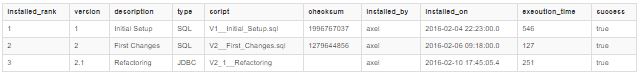


Flyway will once again scan the filesystem and class path of the application for migrations. The migrations are checked against the schema history table. If their version number is lower or equals to the one of the version marked as current, they are ignored. The remaining migrations are the pending migrations available but not applied.





Then the schema table is updated as like this.



**Migration:**

With flyway all changes to the database are called migrations. Migrations can be either versioned or repeatable. Versioned migrations come in 2 forms: regular and undo.

**Why database migrations need:**

Suppose we have a project Shiny and its primary deliverable is a piece of software called shiny soft that connect to a database called shiny DB.



* Migrations are essentially version control of your database, they keep a record of how your database was created and altered over time, and migrations and seeders make it easier to re-deploy the application or distribute to other people.
* If you do not use migrations, there is no record of database structure bundled into your app. Otherwise you Have to keep a copy of schema dump somewhere.

**Flyway Basic Commands:**

* Migrate
* Info
* Repair
* Validate
* Undo
* Baseline
* Clean

**Migrate:**



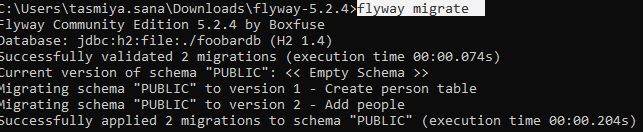
Migrate is the centerpiece of the flyway work. It will scan the file system or your class path for available migrations. It will compare them to the migrations that have been applied to the database. If any difference is found, it will migrate the database to close the gap.

* Go to flyway folder -> Go to SQL folder -> Then create two SQL folder and write SQL script. And give the name of that folder as like below.



Then go to command prompt and execute the below command for migration.

<flyway><space><migrate>



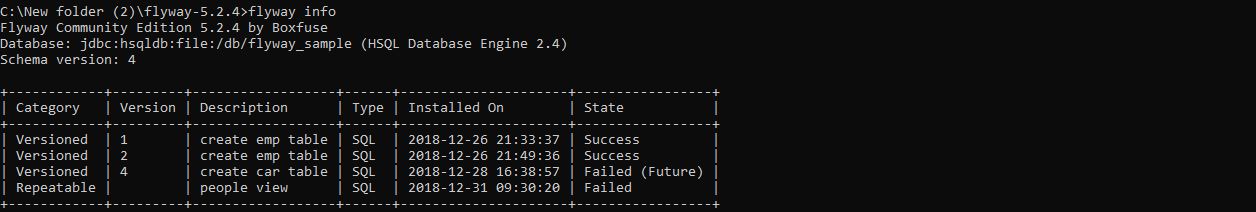
**Info:**

This command prints the details and status information about all the migration.



Go to command prompt and execute the below command.

<flyway><space>< info>

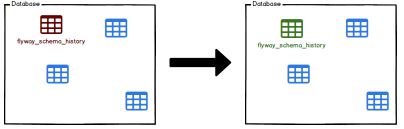


**Repair:**

Repairs the schema history table.

If database supports DDL transactions, Flyway does the work for you.

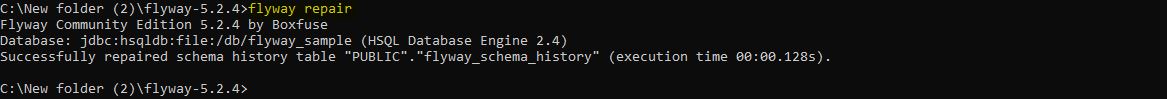
If your database doesn’t, these are the steps to follow:



Repair command has two main uses:

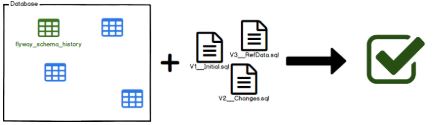
* Remove failed migration entries
* Realign the checksums, description and types of the applied migrations with the ones of the available migrations.

Go to command prompt and execute this command. <flyway><space><repair>



**Validate:**

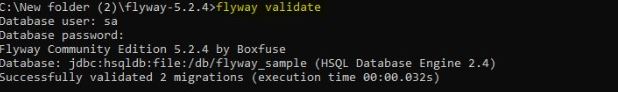
Validates the applied migrations against the available ones.



Validate helps you verify that the migration applied to the database match the ones available locally. This is very useful to detect accidental changes that may prevent you from reliably recreating the schema.

* Go to command prompt and execute the below command.

<Flyway><space><validate>



**Undo:**

Undo migration is the opposite of Regular versioned migrations. An undo migration is responsible for undoing the effects of the versioned migration with the same version. Undo migration are optional and not required to run regular versioned migrations.

* Only enterprise and pro version of flyway DB supports *flyway undo* command. Community edition does not support flyway undo.



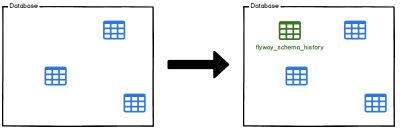
* Go to command prompt and execute the above command

<flyway><space><undo>



**Baseline:**

Baseline an existing database, excluding all migrations up to an including baseline Version.



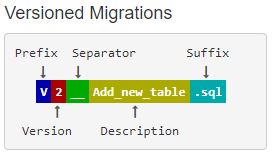
Base line is for introducing flyway to existing databases by baselining them at a specific version. The will cause to ignore all migrations up to and including the base line version

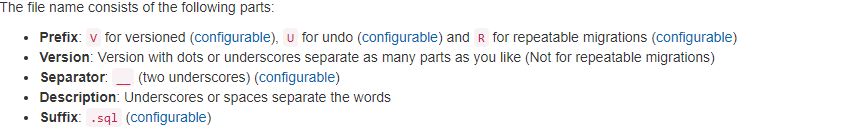
**Types of Migration:**

* Versioned Migration
* Repeatable Migration
* Undo Migration

**Versioned Migration:**

The most common types of migration are called as Versioned Migration. Each Versioned Migration has a versioned, a description and a checksum. The version must be unique.





**Uses:**

Versioned Migration is used for

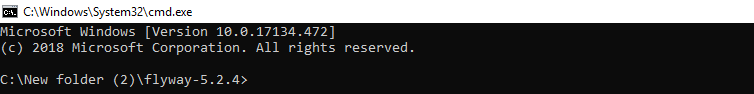
* Creating, Altering, dropping table, indexes, foreign key, enums.
* Reference data updates,
* User data correction.

**Creating Version Migration:**

First you have to flyway folder, then go to SQL folder, inside the SQL folder write SQL query for database.

Then go to the flyway folder and open command prompt like below.

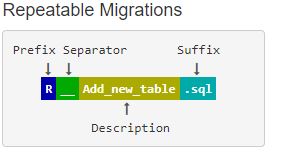


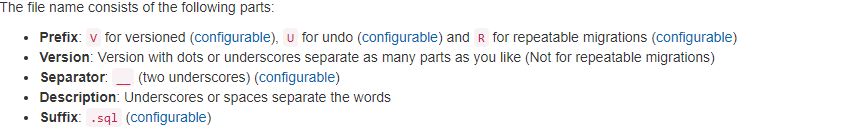


For database migration you Have to execute one command, that is **flyway migrate**.

**Repeatable Migration:**

Repeatable Migrations are very useful for managing database objects whose definition can then simplify be maintained in single file in version control.





**Uses:**

Repeatable Migration is used for

* Bulk Reference data reinserts
* (Re-)creating views/procedures/functions/packages

**Create a Repeatable Migration**

Here we are going to create a repeatable migration to manage a view of the person table. With flyway’s default naming convention, the file name will be like versioned migration, except for the **‘V’** prefix which is now replaced with a **‘R’**.

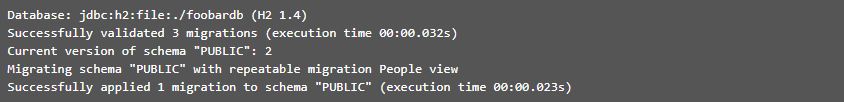
Go to flyway folder -> Then go to SQL folder -> Create one table like R\_\_people\_view.sql and then write script like



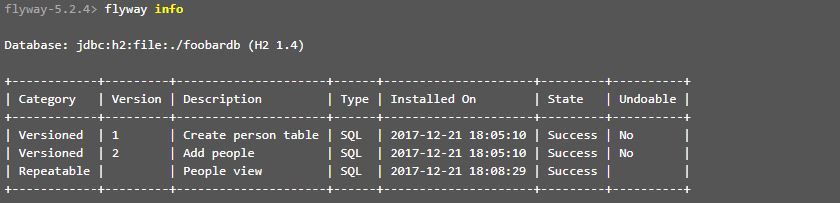
Then go to command prompt and type flyway migrate.



This will give the following results.



Then you can check that this is indeed the new status. Flyway info command give the information about our migration.

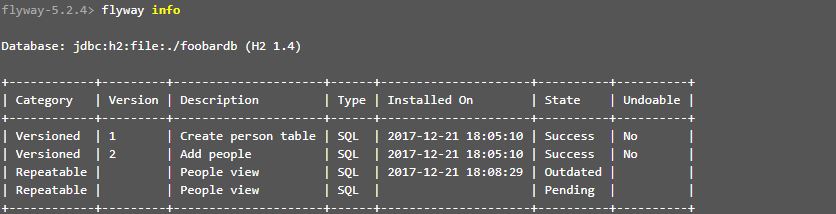


**Modifying the Migration:**

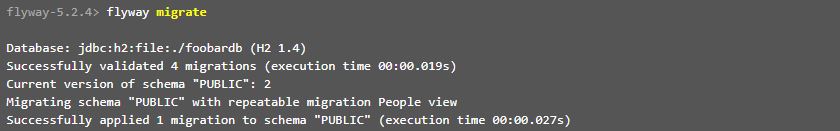
Now we will see what happens when we modify our migration file in place. So, modify your SQL query as follows.



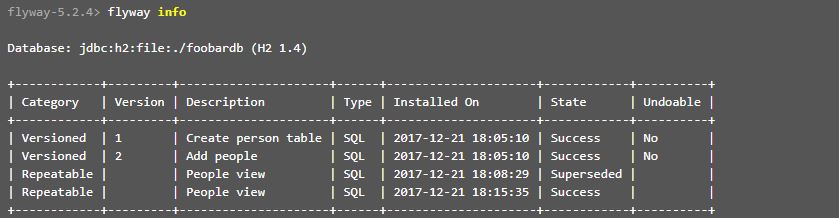
And then check the status again.



So now it clearly shows that the repeatable migration that was previously applied has become outdated and is marked as pending again. So, let’s do exactly that:



And the status is now



**Undo Migrations:**

Undo migration is the opposite of Regular versioned migrations. An undo migration is responsible for undoing the effects of the versioned migration with the same version. Undo migration are optional and not required to run regular versioned migrations.

* Only enterprise and pro version of flyway DB supports *flyway undo* command. Community edition does not support flyway undo.

**Dealing with flyway configuration file:**

These are some main configuration parameters which we need to modify according to our work-

1. flyway.url=

It is JDBC url to use to connect to the database.

Ex- flyway.url=jdbc:postgresql://localhost:5432/testdb

1. flyway. User=

User to use to connect to the database. Flyway will prompt you to enter it if not specified.

Ex- flyway. User=postgres

1. flyway. Password=

Password to use to connect to the database. Flyway will prompt you to enter it if not specified.

Ex- flyway. Password=user@123

1. flyway. Locations=

Comma-separated list of locations to scan recursively for migrations. Locations starting with filesystem: contains the locations of the SQL filesystem where all the SQL migration script takes place. By default, the value will be - filesystem:<<INSTALL-DIR>>/sql)

Ex- flyway. locations=filesystem: C:\flyway-5.2.4\DataBase\App1\dev\sql\DB\_CHANGES

**Does Flyway perform a roll back if a migration fails?**

Flyway runs each migration in a separate transaction. In case of failure this transaction is rolled back. Unfortunately, today only DB2, PostgreSQL, Derby, Enterprise DB and to a certain extent SQL Server support DDL statements inside a transaction. Other databases such as Oracle will implicitly sneak in a commit before and after each DDL statement, drastically reducing the effectiveness of this roll back. One alternative if you want to work around this, is to include only a single DDL statement per migration. This solution however has the drawback of being quite cumbersome.

**Skipping a Particular migration:**

If the migration has already been done and you don’t want the Particular change. Then you can skip the migration by following steps –

1. delete that row from “flyway\_schema\_history” table where version=<version>

For ex – delete from table where version=001;

1. Add the same row with insert command with same values except version.

For ex – if version = 001 then change it in 1

1. Delete the changes done by migration V001\_\_database.sql
2. Run flyway migrate then it will not give any error and migrate to the further version.