**Redgate Flyway**

**Why Flyway & What it Solves:**

Flyway is based on the concept of "migrations," which are scripts that modify the database schema. These migrations are versioned and executed in sequence (Incremental order).

* **Migration Scripts**: Each migration script is versioned, usually with a name like V1\_\_Initial\_schema.sql, V2\_\_Add\_column.sql, etc.
* **Flyway Locations**: The migrations are stored in a specific directory. The default directory is db/migration, but it can be customized.
* **Flyway Metadata Table**: Flyway keeps track of the migrations applied to a database in a table called flyway\_schema\_history. This table stores the version, description, and checksum of each applied migration.

**Core Flyway commands**

**a. flyway migrate**

**Purpose**: This command applies all the pending migrations to the database.

**Syntax :** flyway migrate

**Description :** Flyway will look at the database and apply any migration scripts that have not yet been executed, based on their version numbers.

**b. flyway clean**

**Purpose**: This command drops all objects (tables, views, etc.) in the schema. **Use with caution** as it can delete all data.

**Syntax** : flyway clean

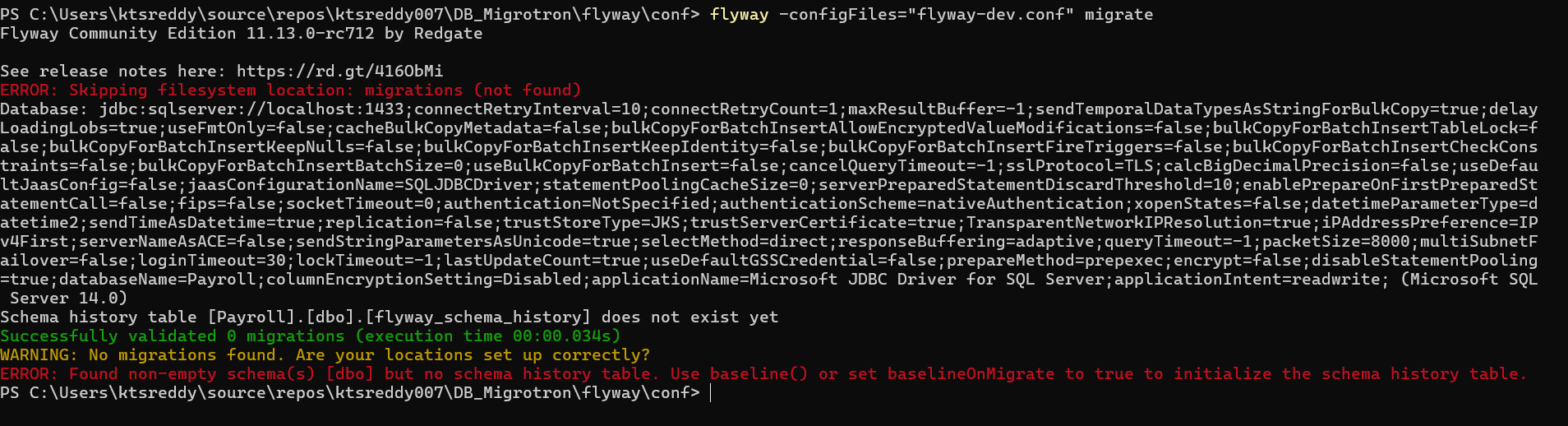
**Description**: It's usually used in development or testing environments where you want to reset the database. **It's irreversible**, so don't use it in production without proper backups.

**Setting up Flyway Migration on Existing DB Schema :**

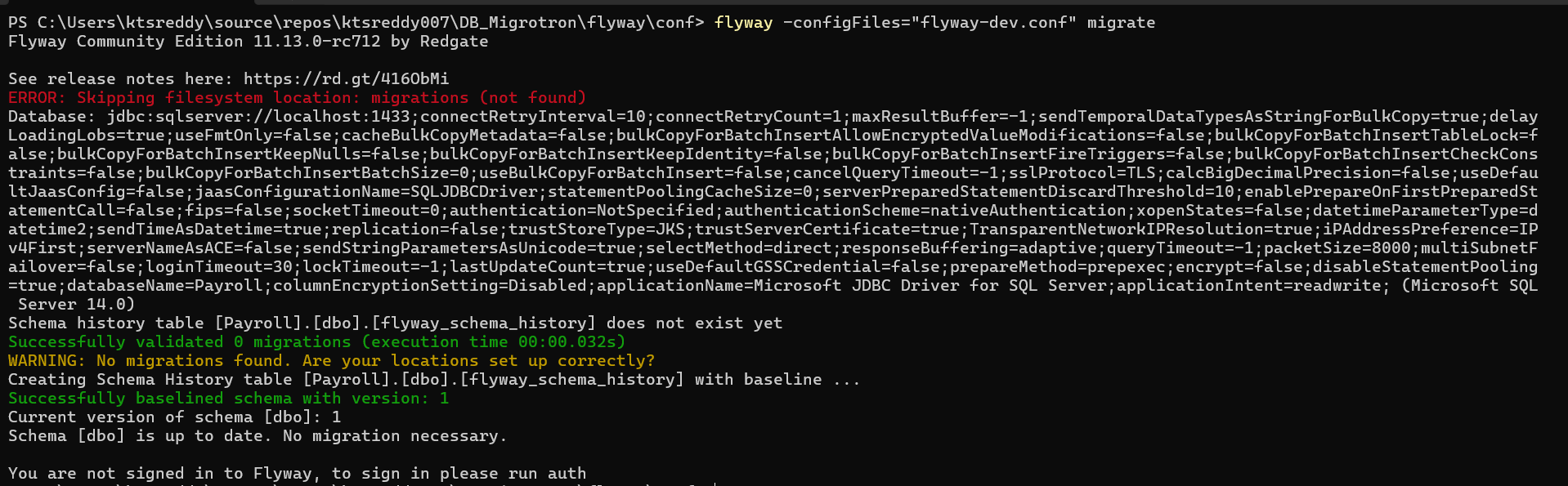
To make flyway recognize that the DB has existing schema and need to create a fresh SchemaHistory table we need to keep below parameter to true in conf file.

**flyway.baselineOnMigrate=true**

Below is the screenshot when we keep the parameter to **false**



After Setting up to **true** we get below output where its creating schema history table .



**Flyway does not load .env files by default (Like secrets and others which are fetched from env file)**

We would get below error



To avoid above follow below process

**Option 1: Manually set env variables in the shell (PowerShell in our case)**

Before running Flyway, set the required environment variables manually:

$env:DB\_NAME=" Give ur original value here "

$env:DB\_USER="Give ur original value here"

$env:DB\_PASS=" Give ur original value here "

Than run - flyway -configFiles="flyway-dev.conf" info

Below Error is a Checksum mismatch error .



The error you're encountering is related to a **checksum mismatch** in Flyway's schema history table (typically named flyway\_schema\_history). This means that the checksum of the migration script in the database **does not match** the checksum of the migration file you are trying to run.

**Error Explanation:**

* **Migration version 2** has been applied to your database.
* The checksum of the migration script that was applied (470051026) does **not match** the checksum of the migration script that Flyway is trying to apply now (-554499142).

This usually happens when:

* The migration file has been modified after it was initially applied (i.e., you edited the script).
* The migration script has changed in a way that alters its checksum, but the schema history table still holds the old checksum.

**Solution Options:**

**1. Revert Changes to the Migration File:**

If you unintentionally changed the migration file after it was applied to the database, and you want to preserve the old script, **revert the file to its original state** so the checksum matches the previously applied one.

You can:

* Roll back the changes you made to the migration script (e.g., undo code changes in the migration SQL file).
* Ensure the script is exactly the same as it was when it was first executed.

**2. Run Flyway Repair (Recommended if you want to update the history):**

If you’ve intentionally changed the migration file, you need to let Flyway know that the script has been updated.

The command flyway repair will:

* **Update** the checksum in the schema history table (flyway\_schema\_history) to match the new checksum.
* **Preserve the migration version** without reapplying it.

**Steps to fix it:**

1. Ensure your migration script is in the desired final state (the updated version).
2. Run the repair command to correct the schema history:

flyway repair

This will update the stored checksum in the flyway\_schema\_history table so that Flyway doesn't try to reapply the migration or complain about the checksum mismatch.

**3. Delete the Migration from the Schema History (Advanced):**

If you're certain the migration script should be reapplied (perhaps after a bug fix or significant change), you can manually delete the migration entry from the flyway\_schema\_history table. **Be cautious with this approach**, as deleting the wrong entry can cause data inconsistencies.

**Steps to manually remove the entry**:

1. Query the flyway\_schema\_history table to locate the row for version 2.
2. Delete that specific entry.

If we want to migrate all files till the targeted version we use below command

flyway migrate -target=5

**Use Flyway's -outOfOrder Option (If Versions are Out of Order)**

If you're trying to apply a version out of order, meaning that you want to apply a migration that is not the most recent one, Flyway also has an -outOfOrder flag to handle such cases.

flyway migrate -target=3 -outOfOrder=true

This will allow Flyway to apply version 3 even if later versions (e.g., version 4) are in a pending state.