Existing database setup

These are the steps to follow to successfully integrate Flyway in a project with an existing database.

Take a DDL and reference data extract from production

First start by taking a snapshot of your most important database: production. This will be the starting point for migrations.

Generate a sql script that includes the entire DDL (including indexes, triggers, procedures, ...) of the production database. To do this you will need to add insert statements for all the reference data present in the database.

This script will form your base migration. Save it on the classpath in the directory you configured with baseDir. Give it a relevant version number and description such as V1__Base_version.sql.

Clean all databases containing data you don't mind losing

Now comes the point where we have to make sure that the migrations meant for production will work everywhere.

For all databases with unimportant data you don't mind losing, perform

```
> flyway clean
```

to completely remove their contents.

Align the remaining databases with production

Check all remaining databases. You must make sure that their structure (DDL) and reference data matches production exactly. This step is important, as all scripts destined for production will be applied here first. For the scripts to succeed, the objects they migrate must be identical to what is present in production.

Give these databases a baseline version

Now comes the time to baseline the databases that contain data (including production) with a baseline version. Use the same version and description you used for the production extract script created above.

You can accomplish it like this:

```
> flyway baseline
```

You must perform this step for each database that hasn't been cleaned.

Done !

Congratulations ! You are now ready.

When you execute

```
> flyway migrate
```

the empty databases will be migrated to the state of production and the others will be left as is.
As soon as you add a new migration, it will be applied identically to all databases.

    Environment Variables   ➔ (/documentation/envvars)