

flyway 简单入门使用

作者: [sirwsl](#)

原文链接: <https://ld246.com/article/1615739731953>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)



故事

好久没写博客了，最近自己放松了不少，想想还是应该坚持下去，毕竟是自己选择的路，今天为什么写flyway，因为目前flyway作为数据库迁移工具，确实很好用，目前应该也有很多公司采用flyway对数据库的sql版本进行管理。

我所在公司也是一样，虽然直接建表很方便，但是当项目进行多租户部署的时候，就会出现数据库错导致程序异常情况，除此之外，上线的系统也会涉及数据库的相关修改，这个时候，采用flyway对数据库版本进行管理。在方便自己的同时也方便别人。

flyway是什么

刚刚的一个故事可能很多人还没明白，flyway是什么。

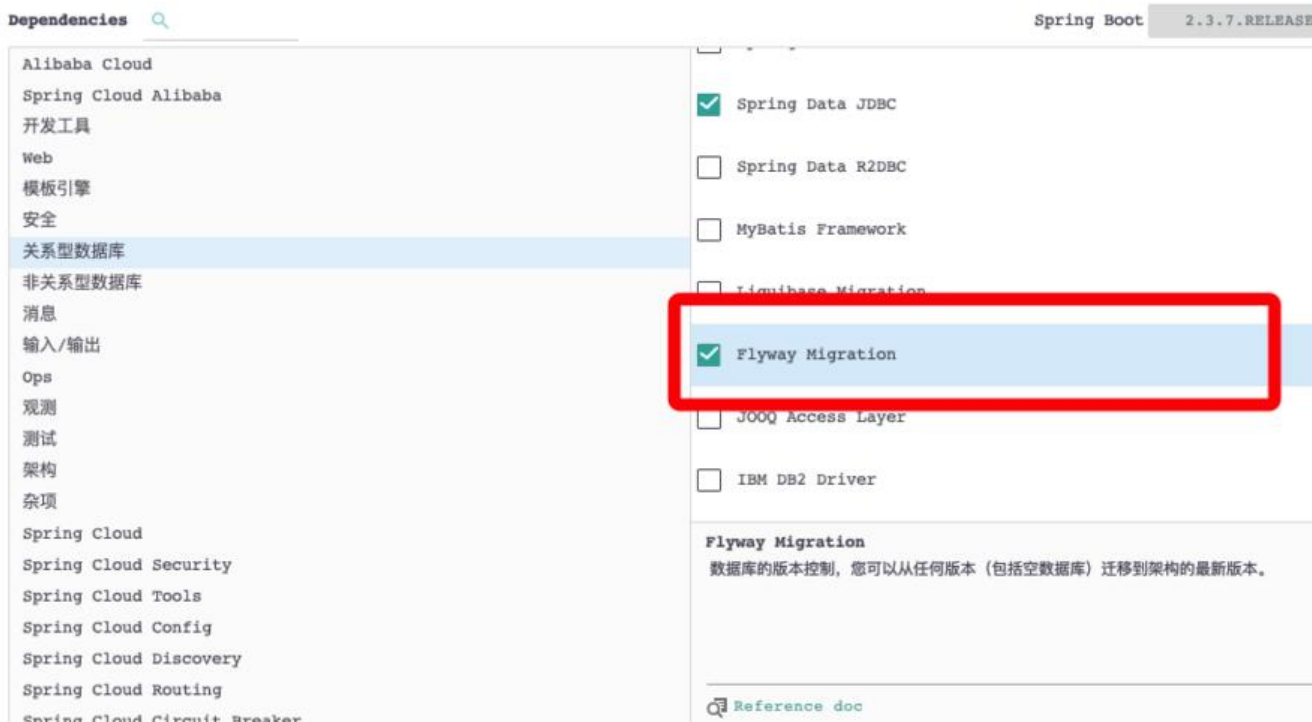
通俗点说：flyway就是在对数据库进行操作时候的版本管理工具

采用flyway能够很好的对数据库版本进行响应管理，多人协同操作，也不会造成错乱现象。

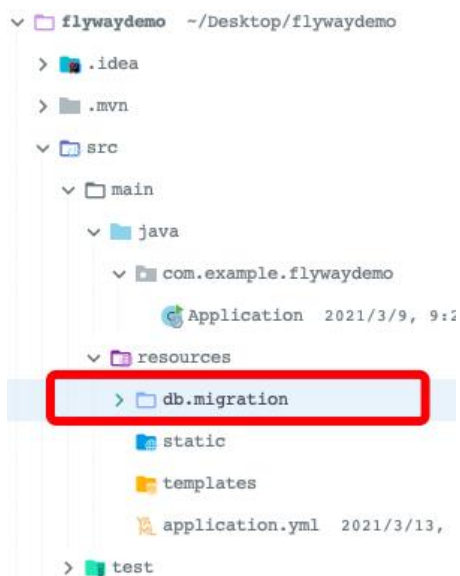
如何使用

flyway是java环境中数据库版本控制工具之一，针对目前现状来说，常在spring系列框架中使用，接下来以springboot为例进行讲解。

在创建Spring项目，项目初始化时候选择flyway依赖



项目创建完后你会看见resources资源目录下会有db/migration目录, 该目录用于存放sql文件。如下所示。



同时你也会看见相关依赖

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

如果是已经生成的项目, 后来需要加入flyway进行数据库版本管理, 直接添加依赖, 创建目录

截至目前依赖已经倒入, 接下来就是配置yml配置文件

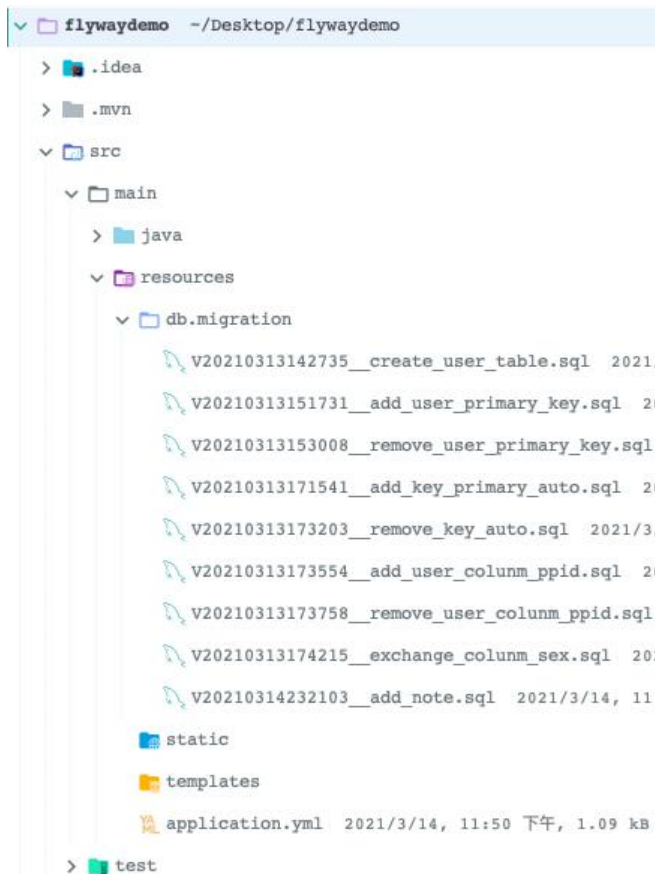
```
server:
  port: 8080
```

```
spring:
  datasource:
    url: jdbc:mysql://localhost:3306/flywaydemo
    username: root
    password: root
    driver-class-name: com.mysql.cj.jdbc.Driver
```

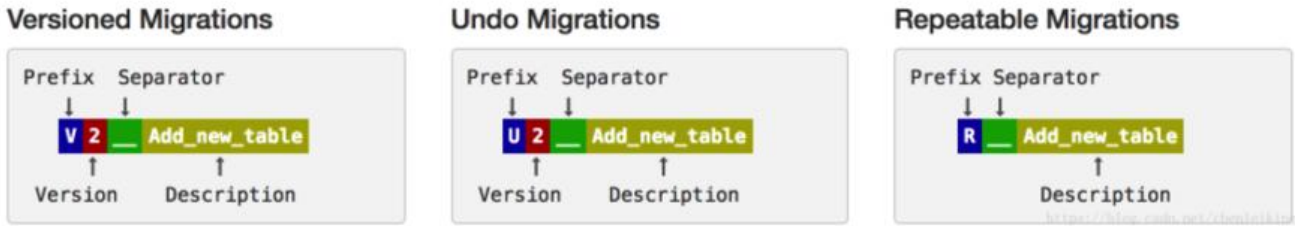
```
flyway:
  enabled: true # 开启flyway
  # flyway 的 clean 命令会删除指定 schema 下的所有 table, 生产务必禁掉。这个默认值是 false
  # 论上作为默认配置是不科学的。
  clean-disabled: true
  locations: classpath:db/migration #flyway文件目录
  table: flyway_schema_history #历史表名
  baseline-on-migrate: true
  validate-on-migrate: true #连续性校验
  baseline-version: 1 #指定默认版本
  encoding: UTF-8 #编码
  out-of-order: false # 是否允许不按顺序迁移 开发建议 true 生产建议 false
  sql-migration-prefix: V #前缀默认v
  sql-migration-separator: _ #分割符, 默认_
  sql-migration-suffixes: .sql #迁移文件, 默认.sql
```

springboot在创建项目时导入了flyway，可以不配置yml，采用默认构建

然后就可以开始在db\migration目录下开始写sql脚本。大概内容如下：



这里需要说一下sql文件的命名方式



命名基本遵循以上三种方式，采用前缀+版本+描述+后缀构成

为什么会是这样呢？因为启动项目时候，flyway会在数据库中生成默认的版本记录表，大概是这样：

flyway默认生成的log表为：flyway_schema_history 也可以自己在yml文件中进行配置自定义表名

名	类型	长度	小数点	不是 null	虚拟	键
installed_rank	int	11	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
version	varchar	50	0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
description	varchar	200	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
type	varchar	20	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
script	varchar	1000	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
checksum	int	11	0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
installed_by	varchar	100	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
installed_on	timestamp	0	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
execution_time	int	11	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
success	tinyint	1	0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

每当我们在db/migration目录下添加一个sql文件，启动项目后就会在flyway_schema_history表中加一条记录，而记录的内容就是根据sql文件的命名进行添加的。

installed_rank	version	description	type	script	checksum	installed_by	installed_on	execution_time	success
1	20210313142735	create user table	SQL	V20210313142735_create_user_table.sql	-214339428	root	2021-03-13 15:32:20	399	1
2	20210313151731	add user primary key	SQL	V20210313151731_add_user_primary_key.sql	-1954095318	root	2021-03-13 15:32:21	410	1
3	20210313153008	remove user primary key	SQL	V20210313153008_remove_user_primary_key.sql	-391403243	root	2021-03-13 17:14:47	373	1
4	20210313171541	add key primary auto	SQL	V20210313171541_add_key_primary_auto.sql	-106360582	root	2021-03-13 17:19:28	364	1
5	20210313173203	remove key auto	SQL	V20210313173203_remove_key_auto.sql	362759667	root	2021-03-13 17:34:33	382	1
6	20210313173554	add user column ppid	SQL	V20210313173554_add_user_column_ppid.sql	89399799	root	2021-03-13 17:40:38	411	1
7	20210313173758	remove user column ppid	SQL	V20210313173758_remove_user_column_ppid.sql	82352669	root	2021-03-13 17:40:39	406	1
8	20210313174215	exchange column sex	SQL	V20210313174215_exchange_column_sex.sql	1243426950	root	2021-03-14 16:12:39	375	1
9	20210314161603	exchange colName	SQL	V20210314161603_exchange_colName.sql	-778482307	root	2021-03-14 16:19:54	389	1
10	20210314162338	add default value	SQL	V20210314162338_add_default_value.sql	-708154419	root	2021-03-14 16:34:06	393	1
11	20210314164344	remvoe default value	SQL	V20210314164344_remvoe_default_value.sql	-122012156	root	2021-03-14 16:44:38	350	1
12	20210314165056	add col default	SQL	V20210314165056_add_col_default.sql	1155788316	root	2021-03-14 16:53:42	403	1
13	20210314165731	add values	SQL	V20210314165731_add_values.sql	-1101633686	root	2021-03-14 17:10:47	391	1
14	20210314232103	add note	SQL	V20210314232103_add_note.sql	1623482104	root	2021-03-14 23:23:05	518	1

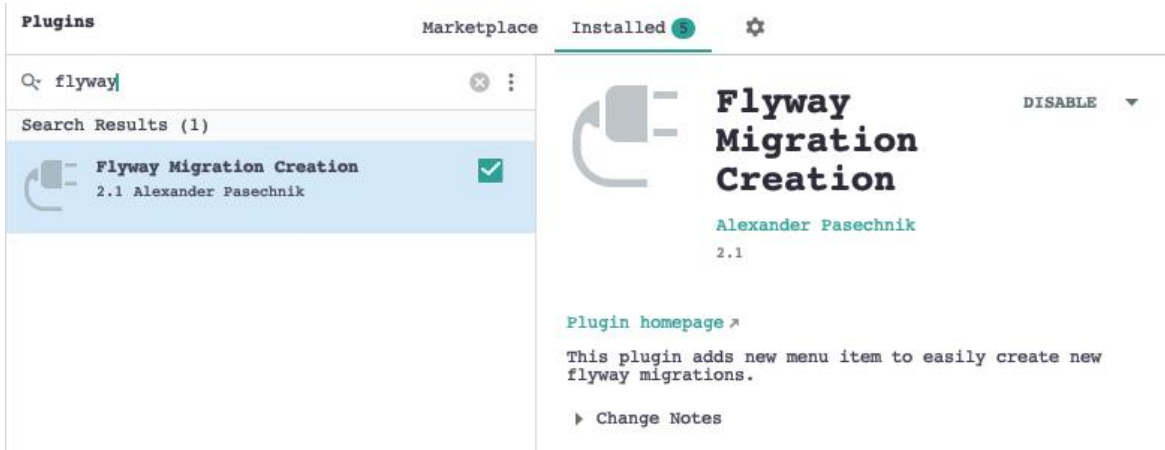
大概样子就是这样。其中version与description就是sql文件划分出来的

重点

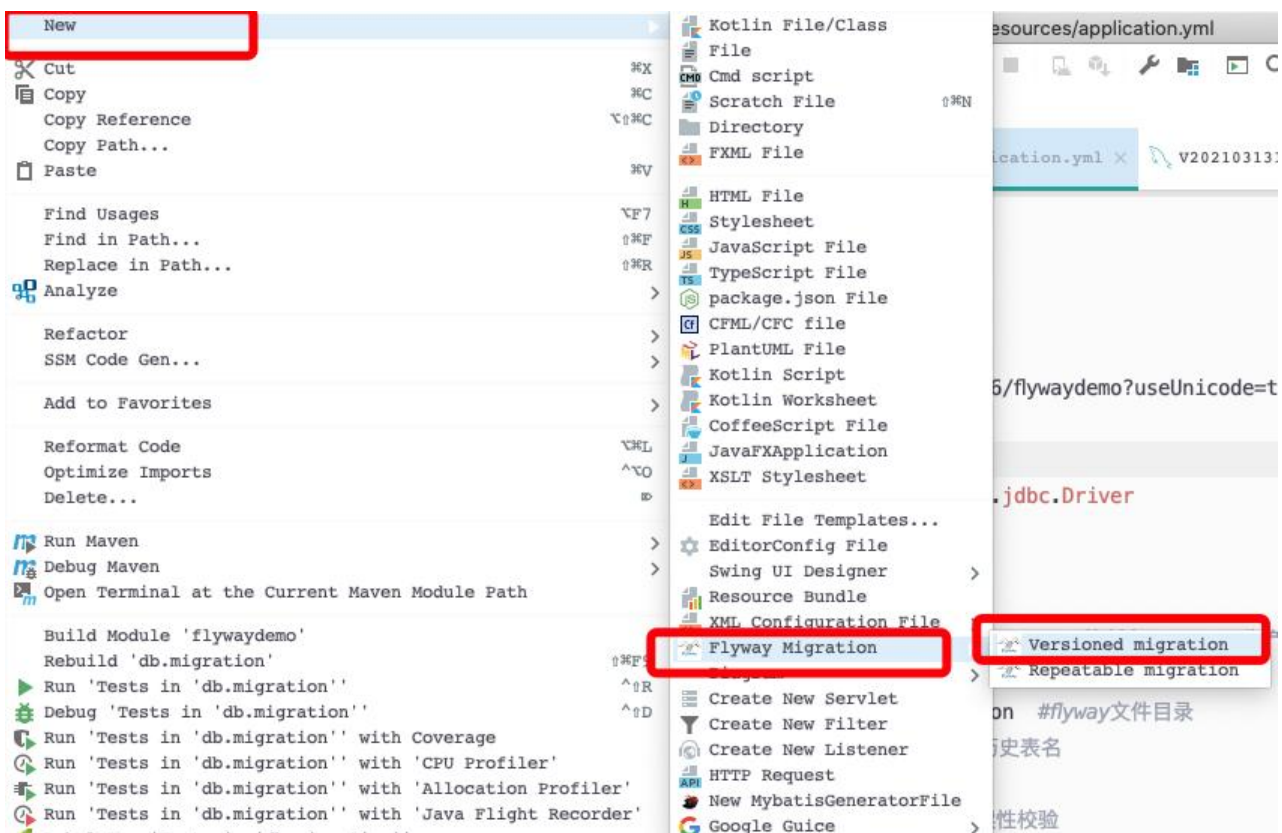
1、sql的版本需要严格递增

例如上一个版本是V1.0 那么下一个版本只能是V1.01 或者 V1.1.0 或者V 2.0.1之类的，因为flyway启动时候会去flyway_schema_history检测是否有已存在的记录，如果有就不在执行，如果没有就执行然后添加记录，这时候版本的命名对我们的管理就会很方便。

这里推荐idea插件 Flyway Migration Creation

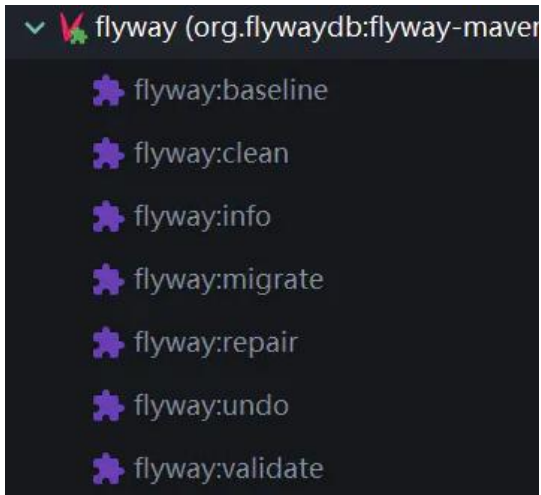


安装插件后，每次在创建.sql脚本时候，选中db/migration目录，然后右键new选择Flyway Migration然后选择第一个Versioned migration，然后只需要输入description就可以，他会自动根据当前时间创建version



2. 在对.sql脚本进行添加时候，每次都需要新增，千万别去改以前的sql文件，否则会导致创建失败
3. 项目执行60%的错误都解决办法都是去history记录表中删除你所添加的这个版本号的记录，然后新执行，剩下20%是你sql写错了，还剩下20%问google吧

flyway常用功能



具体的以后再介绍吧，这里就入门以下就好了

flyway常用sql

1. 创建表

```
CREATE TABLE `user` (  
  `id` int(11) DEFAULT NULL,  
  `name` varchar(255) DEFAULT NULL,  
  `sex` varchar(255) DEFAULT NULL,  
  `detail` varchar(255) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

创建表

2. 添加主键

```
ALTER TABLE `user`  
  ADD CONSTRAINT PK_user PRIMARY KEY (id)
```

#添加主键

#ALTER TABLE `表名` ADD CONSTRAINT PK_表名 PRIMARY KEY (字段名)

3. 移除主键

```
Alter table `user` drop primary key
```

#Alter table 表名 drop primary key

4. 添加主键自增

```
alter table `user` modify id int auto_increment primary key
```

主键自增

#alter table tableName modify id int auto_increment primary key

5. 移除主键自增

```
alter table `user` modify id int(11),drop primary key;
```

```
#重置为普通列  
#alter table class modify id int(11),drop primary key;
```

6.添加列

```
ALTER TABLE `user` ADD ppid bigint
```

```
# 添加列 ALTER TABLE `user` ADD ppid bigint
```

7.删除列

```
ALTER TABLE `user` DROP COLUMN ppid
```

```
#移除表中的列
```

8.改变列属性

```
alter table `user` modify column `sex` varchar(20) DEFAULT NULL
```

9.添加注释

```
ALTER TABLE `user` COMMENT '用户表';
```

```
ALTER TABLE `user` MODIFY COLUMN sex tinyint(1) COMMENT '性别';
```

```
#表注释
```

```
#ALTER TABLE userinfo COMMENT '用户信息资料表';
```

```
# 字段注释
```

```
#ALTER TABLE userinfo MODIFY COLUMN uname VARCHAR(40) COMMENT '姓名';
```

更多的相关东西参见官网doc

<https://flywaydb.org/documentation/concepts/migrations>

也可以去看看这个大佬的博客

<https://blog.csdn.net/chenleiking/article/details/80691750>

sql相关的语句可以去w3school里看看

https://www.w3school.com.cn/sql/sql_alter.asp

flyway的学习demo

[flywaydemo.zip](#)