

# [Activiti 6.x] Springboot 1.5x MySQL

---

Activiti 6.x      Springboot 1.5x +  
Activiti6.0 + MySQL

STS   java8

**New Spring Starter Project**

Service URL:

Name:

☒ Use default location

Location:

Type:  Packaging:

Java Version:  Language:

Group:

Artifact:

Version:

Description:

Package:

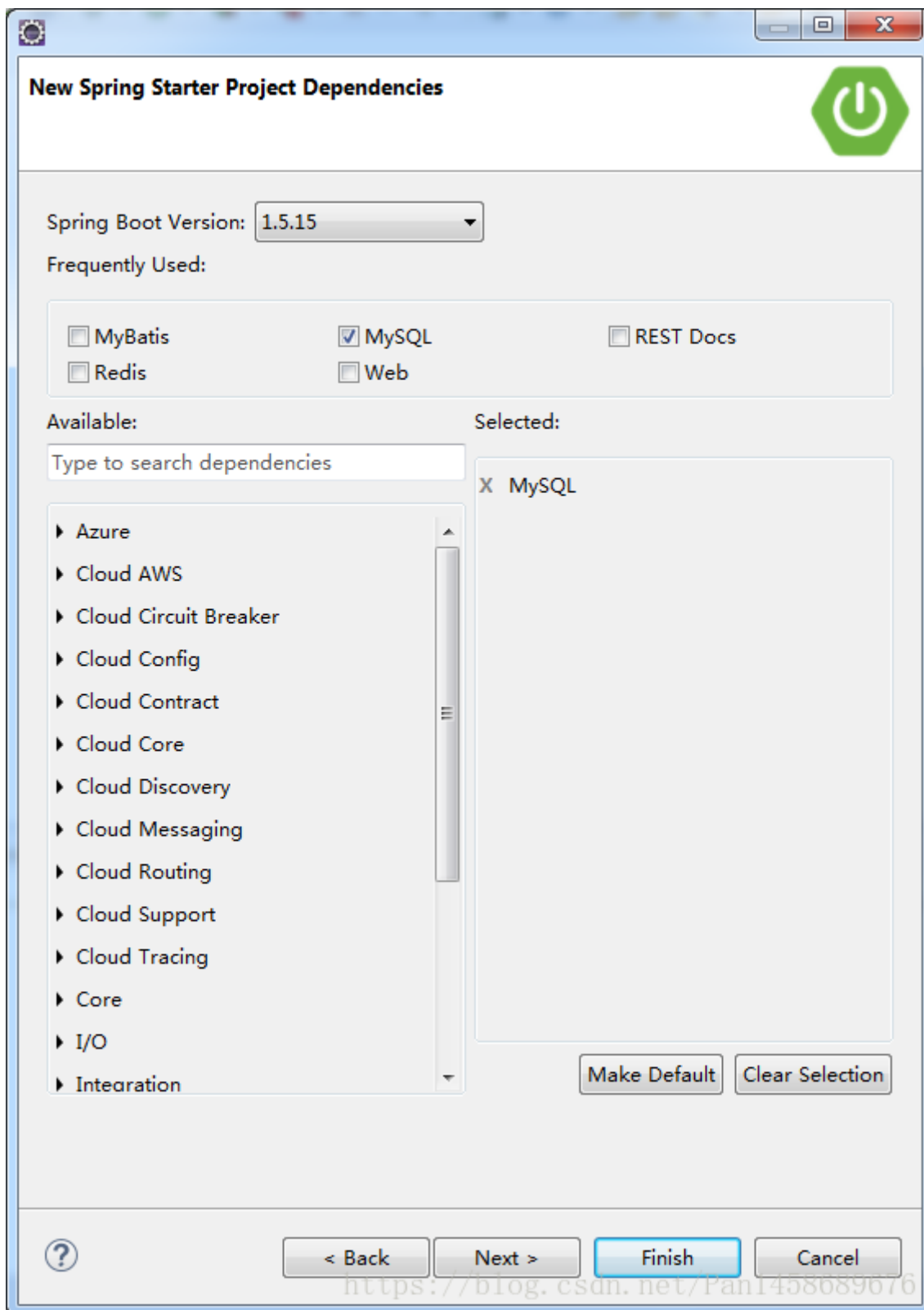
Working sets

☐ Add project to working sets

Working sets:

<https://blog.csdn.net/Pan1458689676>

springboot 1.x      activiti starter      springboot1.x      1.x      web      mysql



1.

```
<dependency>
  <groupId>org. activiti</groupId>
  <artifactId>activiti-spring-boot-starter-basic</artifactId>
```

```
<version>6.0.0</version>
</dependency>
```

## 2. @EnableAutofiguration

```
1 package com.ptm.prdemo;
2
3 import org.springframework.boot.SpringApplication;
4 import org.springframework.boot.autoconfigure.EnableAutoConfiguration;
5 import org.springframework.boot.autoconfigure.SpringBootApplication;
6
7 @EnableAutoConfiguration
8 @SpringBootApplication
9 public class ActivitiApplication {
10
11     public static void main(String[] args) {
12         SpringApplication.run(ActivitiApplication.class, args);
13     }
14 }
15
```

<https://blog.csdn.net/Pan1458689676>

## 3. activiti

```
*****
APPLICATION FAILED TO START
*****
```

Description:

Cannot determine embedded database driver class for database type NONE

Action:

If you want an embedded database please put a supported one on the classpath. If you have database:

<https://blog.csdn.net/Pan1458689676>

## 4. druid mysql H2 QAQ

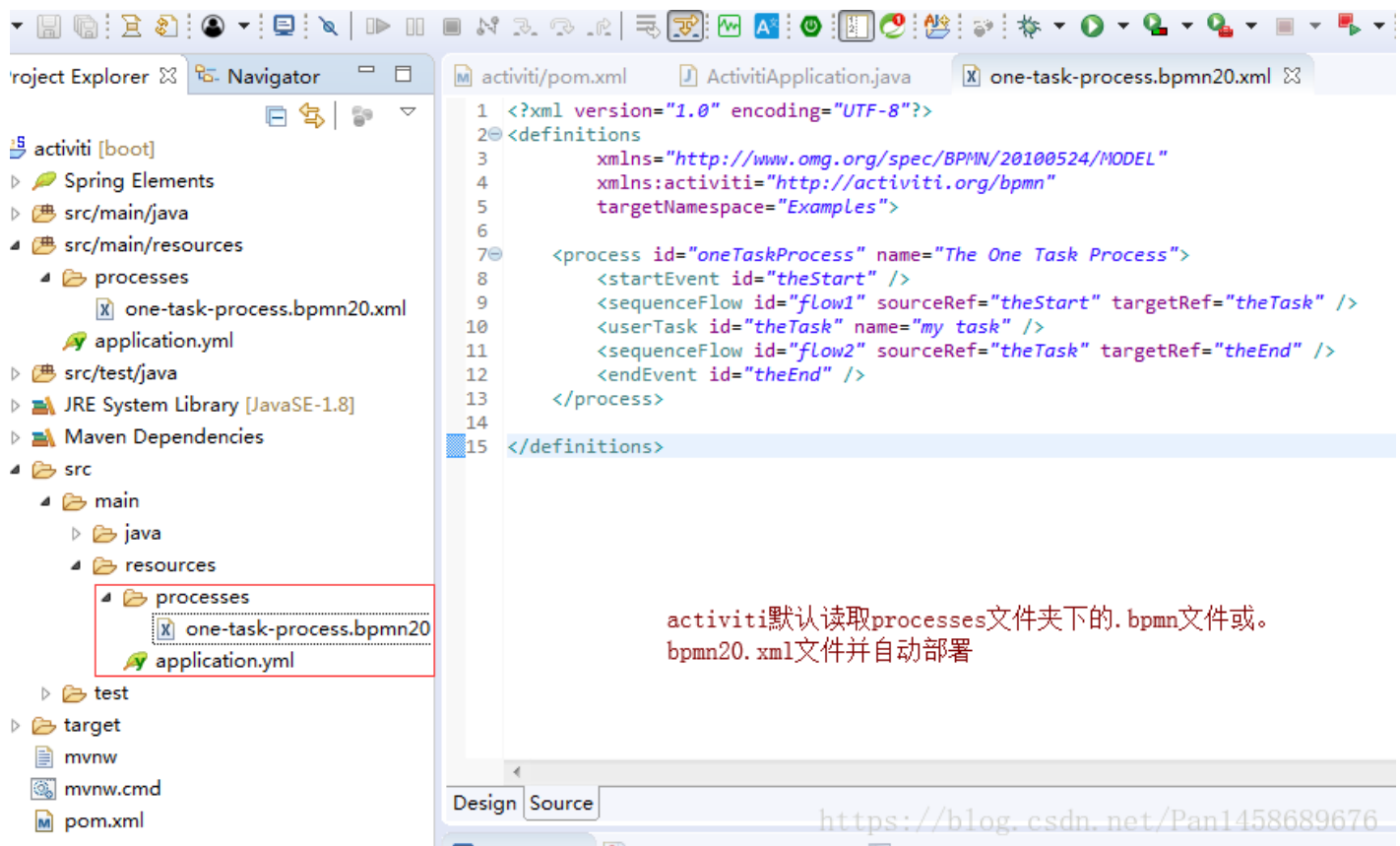
```

1 server :
2   port : 8080
3
4 spring:
5   datasource:
6     url: jdbc:mysql://. . . /Phime?useSSL=false
7     username: . . .
8     driver-class-name: com.mysql.jdbc.Driver
9     password: . . .
10    tomcat:
11      max-active: 1000
12      max-idle: 1000
13
14 logging:
15   path: /var/logs # 保存日志文件目录路径
16   level:
17     com.ptm.prdemo: debug # 不同目录下的日志可配置不同级别
18   #file: 日志文件, 绝对路径或相对路径
19   #config: 日志配置文件, Spring Boot默认使用classpath路径下的日志配置文件, 如: logback.xml
20   #Spring Boot中的logging.path和logging.file这2个属性, 只需要配置其中之一即可, 如果同时配置, 则使用logging.file属性

```

<https://blog.csdn.net/Pan1458689676>

## 5.processes



The screenshot shows an IDE with the following components:

- Project Explorer:** Shows the project structure. The `src/main/resources/processes` folder is highlighted with a red box, containing `one-task-process.bpmn20.xml` and `application.yml`.
- Navigator:** Shows the same project structure.
- Editor:** Displays the content of `one-task-process.bpmn20.xml`. The XML content is as follows:
 

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <definitions
3   xmlns="http://www.omg.org/spec/BPMN/20100524/MODEL"
4   xmlns:activiti="http://activiti.org/bpmn"
5   targetNamespace="Examples">
6
7   <process id="oneTaskProcess" name="The One Task Process">
8     <startEvent id="theStart" />
9     <sequenceFlow id="flow1" sourceRef="theStart" targetRef="theTask" />
10    <userTask id="theTask" name="my task" />
11    <sequenceFlow id="flow2" sourceRef="theTask" targetRef="theEnd" />
12    <endEvent id="theEnd" />
13  </process>
14
15 </definitions>
      
```

activiti默认读取processes文件夹下的.bpmn文件或.bpmn20.xml文件并自动部署

<https://blog.csdn.net/Pan1458689676>

“ one-task-process.bpmn20.xml

```
<?xml version="1.0" encoding="UTF-8"?>
```

```

<definitions
  xmlns="http://www.omg.org/spec/BPMN/20100524/MODEL"
  xmlns:activiti="http://activiti.org/bpmn"
  targetNamespace="Examples">

  <process id="oneTaskProcess" name="The One Task Process">
    <startEvent id="theStart" />
    <sequenceFlow id="flow1" sourceRef="theStart" targetRef="theTask" />
    <userTask id="theTask" name="my task" />
    <sequenceFlow id="flow2" sourceRef="theTask" targetRef="theEnd" />
    <endEvent id="theEnd" />
  </process>

</definitions>

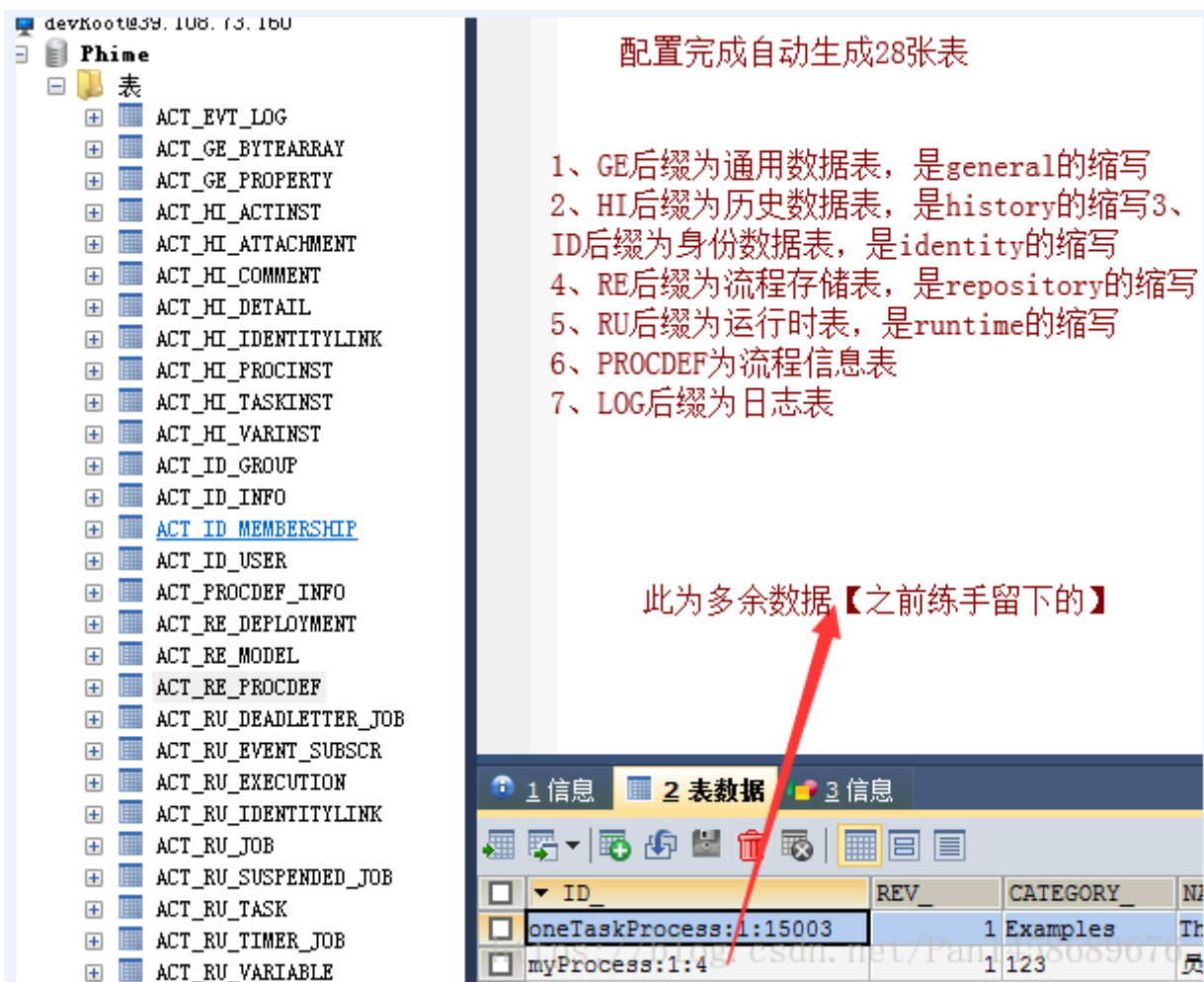
```

## 6.

配置完成自动生成28张表

- 1、GE后缀为通用数据表，是general的缩写
- 2、HI后缀为历史数据表，是history的缩写
- 3、ID后缀为身份数据表，是identity的缩写
- 4、RE后缀为流程存储表，是repository的缩写
- 5、RU后缀为运行时表，是runtime的缩写
- 6、PROCDEF为流程信息表
- 7、LOG后缀为日志表

此为多余数据【之前练手留下的】



The screenshot shows the Phime database interface. On the left, a list of 28 tables is displayed, including ACT\_EVT\_LOG, ACT\_GE\_BYTEARRAY, ACT\_GE\_PROPERTY, ACT\_HI\_ACTINST, ACT\_HI\_ATTACHMENT, ACT\_HI\_COMMENT, ACT\_HI\_DETAIL, ACT\_HI\_IDENTITYLINK, ACT\_HI\_PROCINST, ACT\_HI\_TASKINST, ACT\_HI\_VARINST, ACT\_ID\_GROUP, ACT\_ID\_INFO, ACT\_ID\_MEMBERSHIP, ACT\_ID\_USER, ACT\_PROCDEF\_INFO, ACT\_RE\_DEPLOYMENT, ACT\_RE\_MODEL, ACT\_RE\_PROCDEF, ACT\_RU\_DEADLETTER\_JOB, ACT\_RU\_EVENT\_SUBSCR, ACT\_RU\_EXECUTION, ACT\_RU\_IDENTITYLINK, ACT\_RU\_JOB, ACT\_RU\_SUSPENDED\_JOB, ACT\_RU\_TASK, ACT\_RU\_TIMER\_JOB, and ACT\_RU\_VARIABLE. On the right, the 'oneTaskProcess' table is selected, and its data is displayed in a table view. The table has columns for ID\_, REV\_, CATEGORY\_, and NAME. The data shows two rows: one for 'oneTaskProcess:1:15003' and another for 'myProcess:1:4'.

ID_	REV_	CATEGORY_	NAME
oneTaskProcess:1:15003	1	Examples	The One Task Process
myProcess:1:4	1	123	my task

Revision #1

Created 25 March 2020 16:20:51 by

Updated 2 April 2020 16:37:05 by