MockMVC

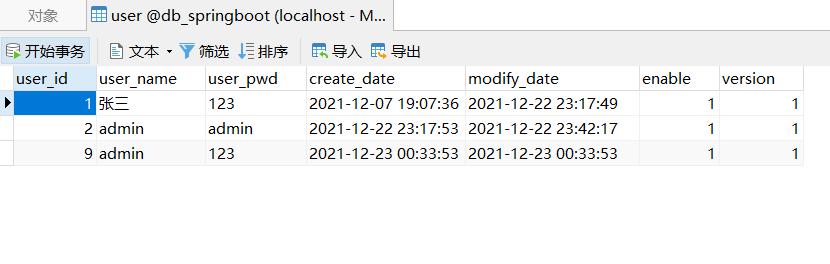
通常我们在测试Web应用程序时，都会编写一些测试来断言应用程序的行为，常见的测试方式是通过启动应用程序并侦听连接（就像在生产中所做的那样），然后发送 HTTP 请求并断言响应。

另一种有用的方法是根本不启动服务器，而只测试其下的层。这里我们可以使用 SpringMockMvc并通过使用@AutoConfigureMockMvc在测试用例上注释为我们注入。它能够让Spring 处理传入的 HTTP 请求，并将其传递给我们的控制器。这样，几乎整个堆栈都被使用了，而我们的代码将以与处理真实 HTTP 请求完全相同的方式调用，达到了无需启动服务器的效果。

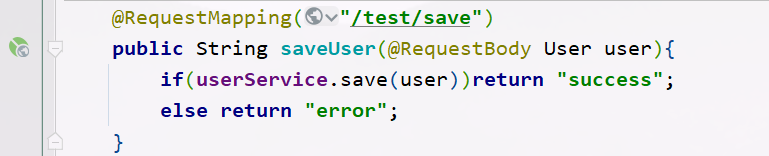
在测试中，我们还可以使用@WebMvcTest注解将测试范围缩小到仅 Web 层。从而无需启动完整的 Spring 应用程序上下文。如果在具有多个控制器的应用程序中，我们可以使用@WebMvcTest(UserController.class)来详细地指定我们测试的web层。

下面为 **“添加用户“** 功能做一个测试：

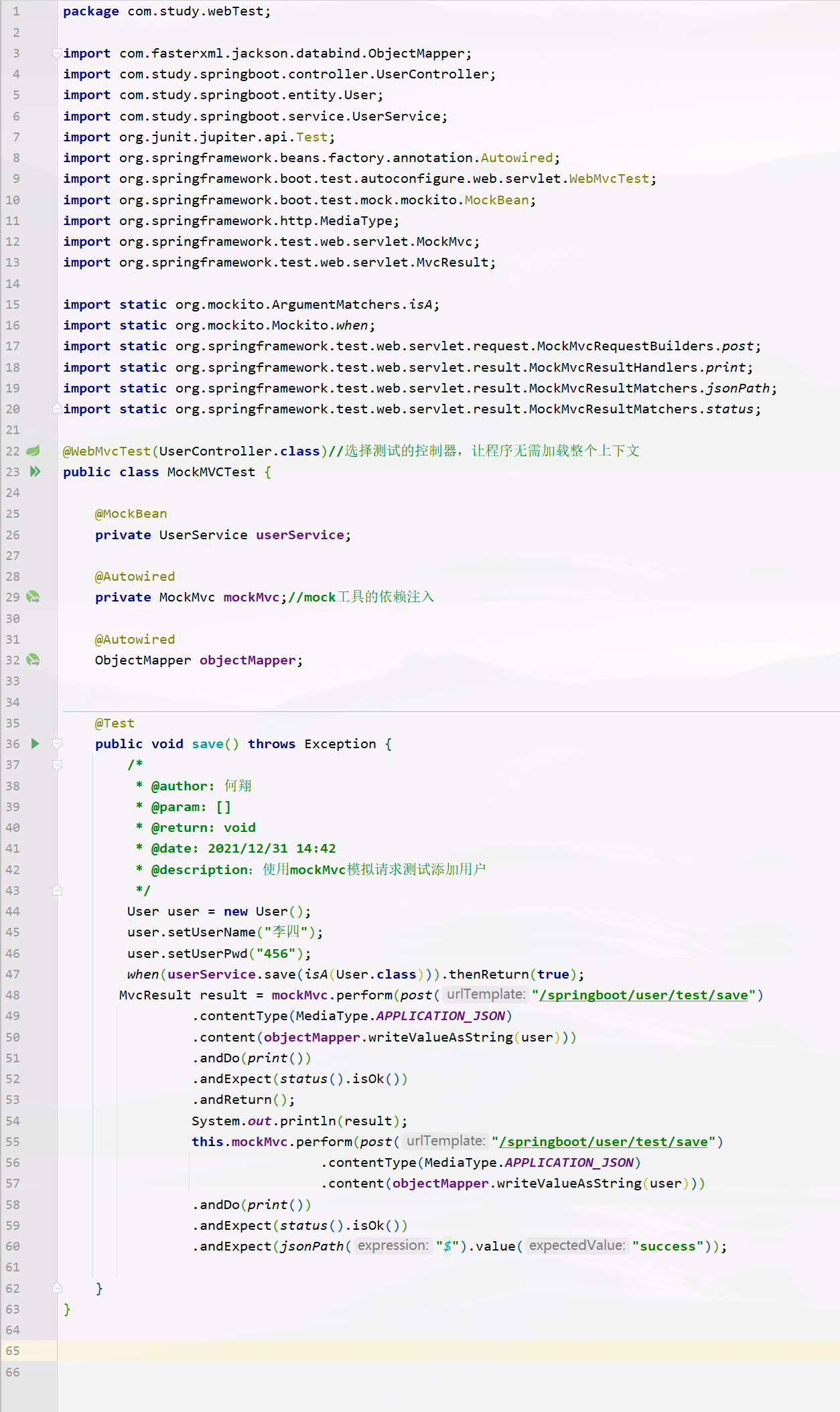
添加用户前的数据：



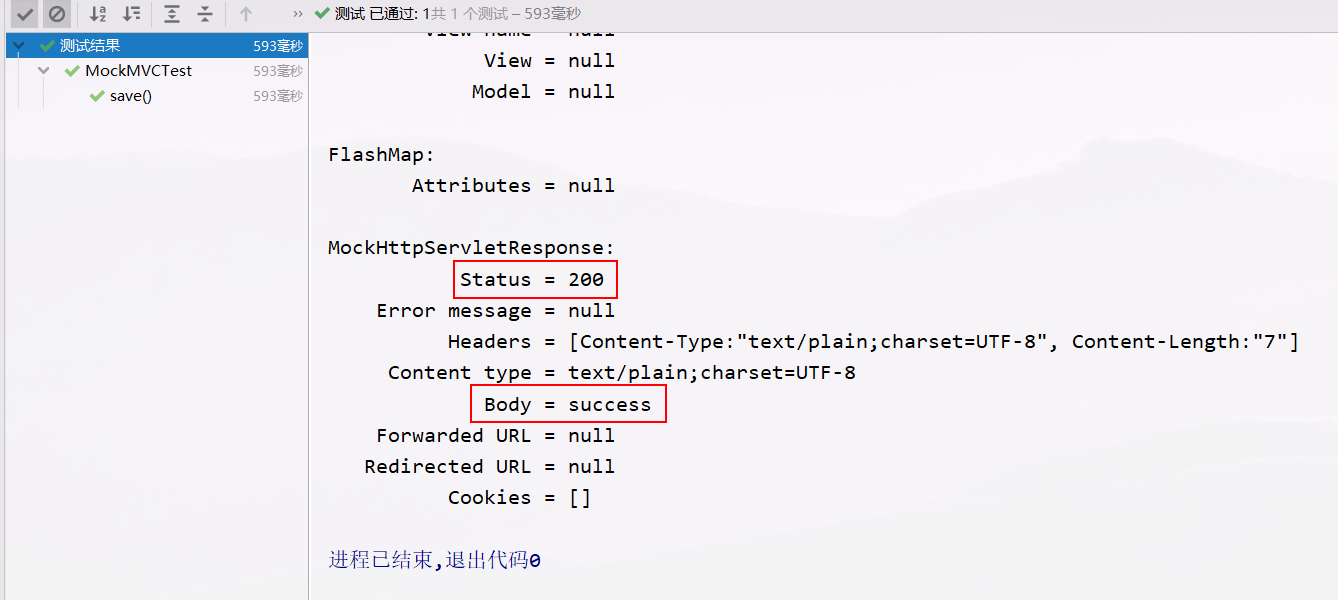
添加用户的测试代码：（UserController.java）



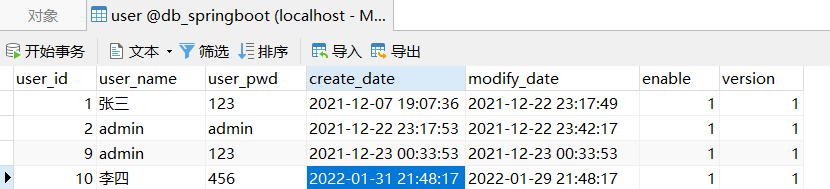
添加用户的测试代码：



测试成功：



数据查看：



总结：

以往controller层的单元测试，通常只能使用浏览器测试，如果通过启动服务器，建立http client进行测试，这样会使得测试变得很麻烦，比如，启动速度慢，测试验证不方便，依赖网络环境等，这样会导致测试无法进行，为了可以对Controller进行测试，可以通过引入MockMVC进行解决。

  MockMvc实现了对Http请求的模拟，能够直接使用网络的形式，转换到Controller的调用，这样可以使得测试速度快、不依赖网络环境，而且提供了一套验证的工具，这样可以使得请求的验证统一而且很方便。