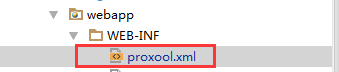
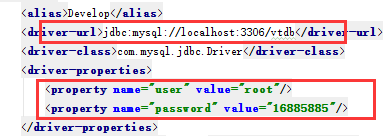
例子1：

**添加学院信息**

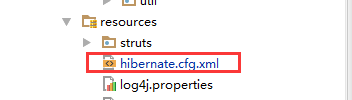
1. **确认连接池 和 hibernate.cfg.xml里的 数据库 都已配置好**

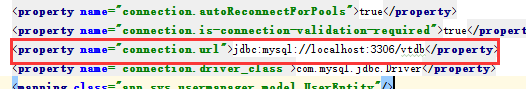
Proxool连接池：



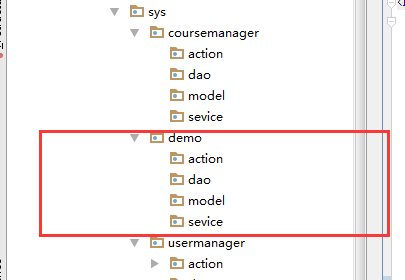


Hibernate：





**第二步、创建demo模块，同时创建相应的包Package**



Action包 ：是存放Action类的地方，常以 XXXAction来命名。 里面有属性和方法， 用来承受前台暴风雨般的访问！同时给予前台热情的回应。后面细讲。

Model包：丢在里面的都是一些实体类，以XXXEntity 来称呼，每个Entity类都表示一张数据库表。后面细讲。

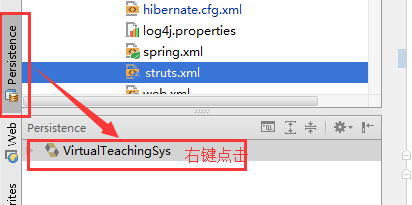
Dao 包: 里面都是一些 XXXDao 类， Dao 类专门对Entity类进行增删改查的操作，其存储了一些基本的增删改查的方法。一个Dao只能操纵其对应的Entity，比如，ADao.save(AEntity);是对的，而ADao.save(BEntity);电脑就会爆炸。后面细讲。

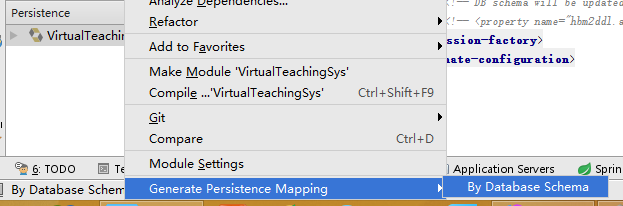
Service 包：里面存放一个Service接口，和其实现类。Service所提供的方法都是对数据库进行操作的方法，Dao 就在这里面用到，后面细讲。

**第三步、创建一个Entity类！**

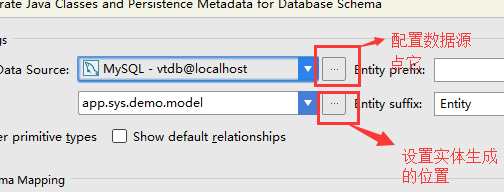
每个Entity类都与一张表相对应着，Entity类不需要自己写，我们可以使用IDEA帮我们反向生成它。

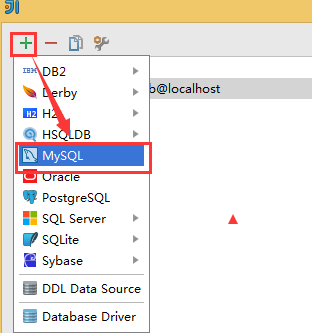
首先，

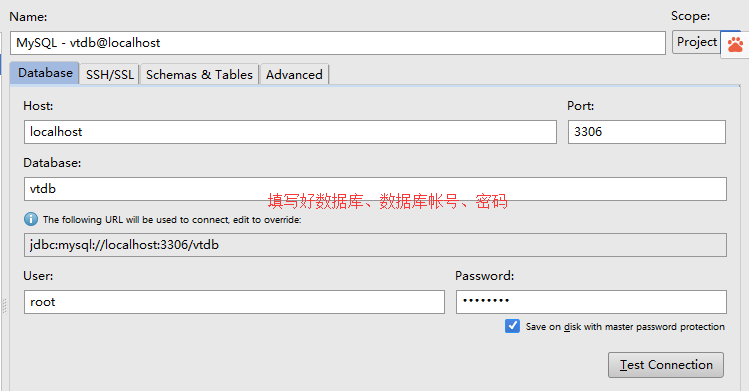


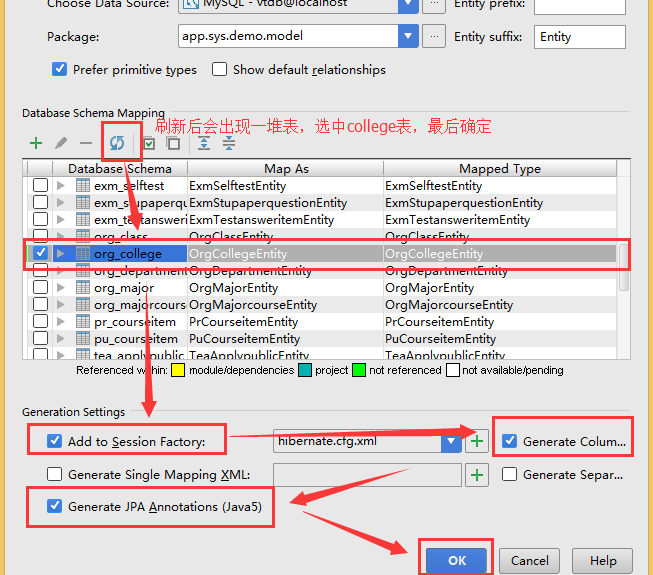


接着，

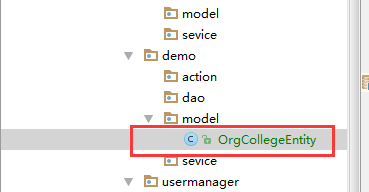








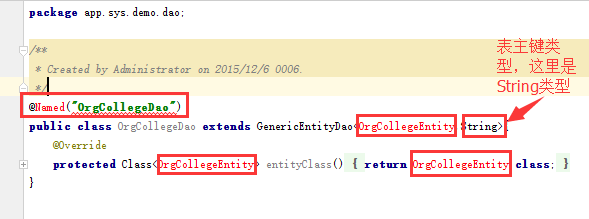
一阵疯狂运行过后 ，实体生成了



**第四步：创建一个OrgCollegeEntity的 专属个人Dao 类，取个外号叫OrgCollegeDao**

****

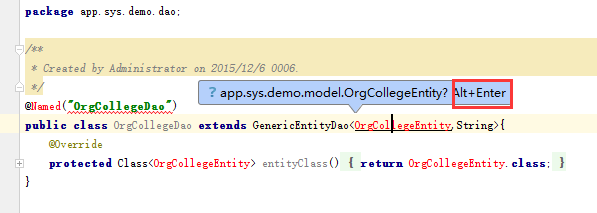
里面的内容是什么呢？

****

就是这么写的，不要问我为什么，直接从其它的Dao拷过来，改！记得把红框里的内容改成对应的东西

看到那鲜艳的红色了吗？

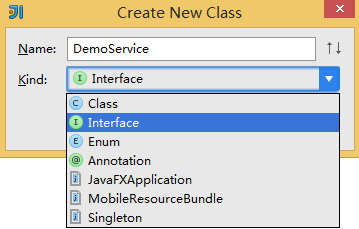
肯定是因为包没import进来，把光标放到那个红色字体上，**Alt+Enter**就搞定了

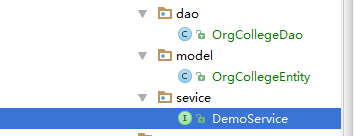


Alt+Enter快捷键是个好帮手，能快速智能地帮你解决碍眼的红色的错误。

**第五步：创建一个Service接口和它的实现类**

创建接口：

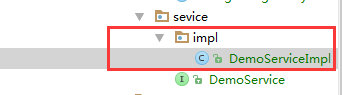




进入，并写下这样的代码



创建实现类：

****

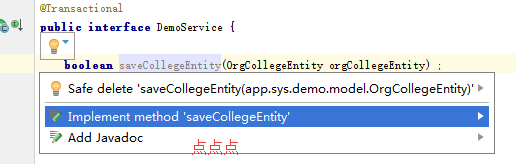
****

就是要写成这样，照着Copy，改

接下来就要在这个空空的DemoServiceImpl.java里面实现接口里的方法了，先别急着敲，我们先**回到**DemoService接口这个文件里面



（当然也可以不用点灯泡，直接Alt+Enter）





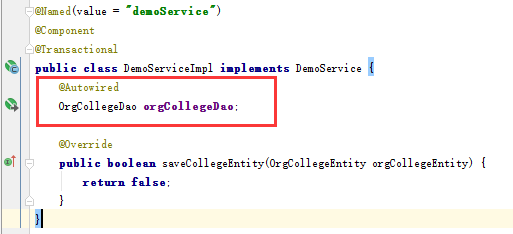
自动跳到这里面来了，帮你生成了些基本代码



看到左边小红框里的绿色小图标没？点它就能轻松返回DemoService接口文件的对应**方法（函数）**处，同理，在DemoService的左边也有类似的小图标，能够让你快速地回到这个实现类里的**方法实现处**，方便了一些。

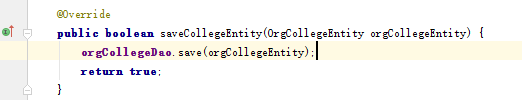
**第六步：完成saveCollegeEntity()这个方法的内容**

在这个实现类里继续敲打出红框里的代码

****

刀（Dao）磨了那么久，终于要拿来砍了，红框里使用了Spring的依赖注入特性，反正就是，那个紫色的orgCollegeDao已经帮你初始化好了，待会儿直接用。

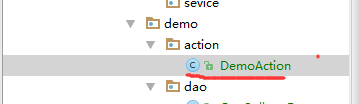
继续写



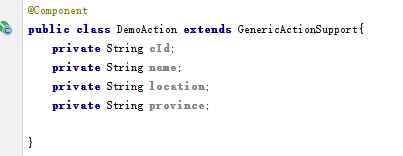
PS：dao里面有一些封装好的基本的增删改查方法，每个Dao都继承自GenericEntityDao这个公共Dao类，Ctrl+B快捷键可以直接跳到类里面查看

**第七步：写一个Action类**

1. 创建



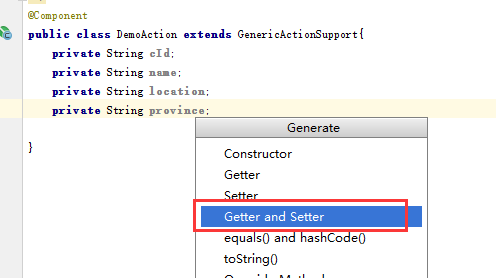
1. 声明属性

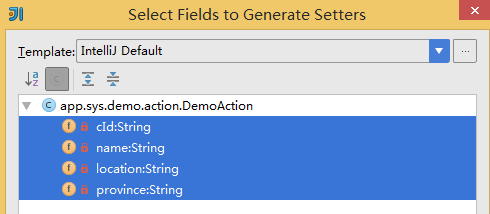


属性是用来接收前台传过来的参数的，无论前端访问了这个Action的哪个方法，只要前台的**参数名和属性名一样**，Struts2会自动把前台传过来的参数赋值给属性（其实是调用了属性的Set()方法来赋值的,后面讲），这样就能拿属性里的值随便乱用了！我们待会要把它们存进数据库里

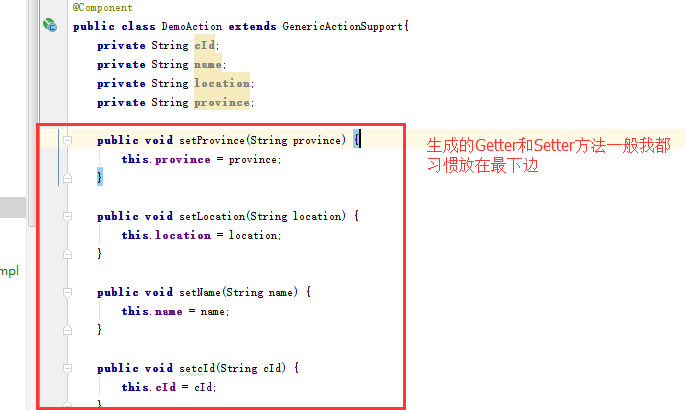
1. 为属性生成Get()、Set()方法

把光标移到最下边的属性旁边，右键选择Generate（或者Alt+Insert）



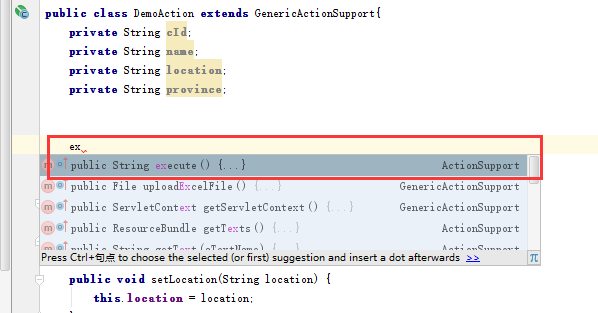


按Ctrl把四个属性都选上，确定



1. 添加默认的execute方法

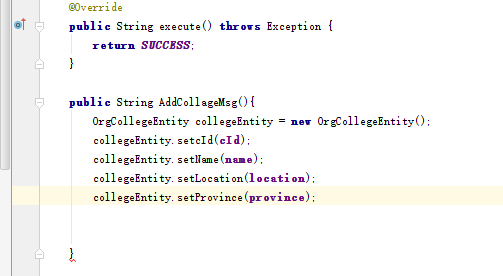
输入ex马上就会有智能提示，选第一个



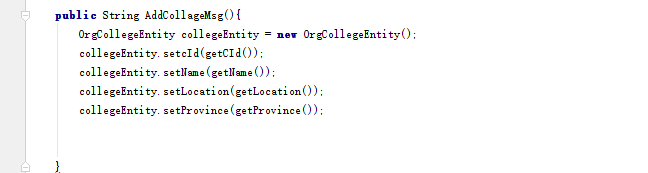
这个execute方法是每个Action方法都要有的，不过可以不管它

1. 添加addCollageMsg方法(就是这个方法了！添加一条学院的信息)

同时把属性数据填到OrgCollegeEntity里面去

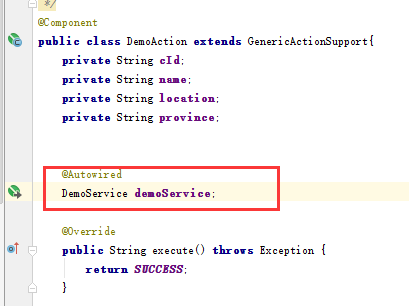


**不过约定一下，最好用Getter方法来获取属性值，改成这样吧**

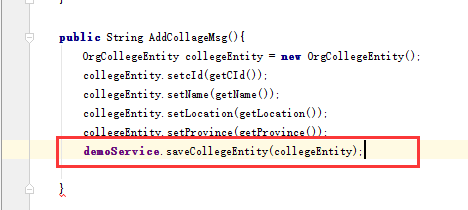
****

1. Service的使用

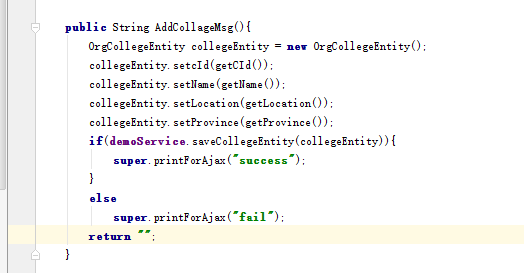
此时此刻，前台数据已经获取到了，怎么存呢？想起来我们前面写好的服务了吧，这时候就要用到它了，先在前面声明它！



这里还是用了Spring的依赖注入，demoService已经帮你初始化好了，直接用



如果前台是使用Ajax来访问的，需要给前台一个反馈



PrintForAjax是我封装在父类里的一个方法，如果要给前台返回Json数据，可以直接调用**super**.writeJsonArray();可以往括号里面塞入List \Map \Entity 都行

常用快捷键：“Alt + Enter”智能提示，错误解决；

“Ctrl + B ”跳转到指定类里；

“Ctrl + Shift + / ” 整段注释

“Tab” 智能提示选择，当出现智能提示时，用Tab选择 ，最好别用Enter选择（否则会换行）

“ /\*\* + Enter ”函数注释说明，如图

