

在本节课开始之前,我已经在我的github主页上创建了一个项目,我们课程所编写的所有代码我都将会上传到github这个仓库。

```
#github仓库地址  
https://github.com/teebbstudios/teebblog
```

现在我们提交一次代码,当提交好代码之后,Phpstorm会将我们代码的更改都会变成白色已提交的状态。刷新,我们的提交已经更新到了仓库。

Symfony是MVC框架,M就是model。大多数MVC框架都会使用ORM框架,来对model进行对象和数据库表的关系映射。Symfony使用doctrine作为ORM框架。

我们打开[symfony.sh](https://symfony.sh)这个网站,搜索ORM,在最后一行,我们可以通过别名的方式来安装这个组件。

打开底部控制台,输入`composer req orm`回车,组件安装完成后会对我们的代码进行一些更改。我们已经创建了一个git提交,现在我们看一下组件对哪些文件进行了更改。

输入`git status`,首先他在配置文件夹中创建了doctrine配置文件,另外添加了一个migrations目录,并且在src目录中创建Entity目录,Entity目录中存放的是MVC设计模式中的model类,对应着ORM就是O,object对象。

然后我们查看.env文件,在.env文件中添加了DATABASE\_URL这个参数(环境变量),我们使用的是mysql数据库,这里需要修改一下。#号是行前的注释,mysql的URL中用户是root,密码是password,本机的8888端口,数据库名是teebblog。

我们看一下config目录中的配置文件,doctrine.yaml文件,它会解析我们的.env文件配置的DATABASE\_URL,来进行数据库的连接。再往下,我们看orm键下的配置,它自动映射的是App\Entity命名空间下的model类,对应的就是我们src目录下的Entity目录中的类。

现在我们查看一下命令行的更改,输入`symfony console`,orm组件给我们安装了很多doctrine的命令,我们可以使用这些命令对数据库进行一些操作。比如说数据库的创建、删除或者数据库数据的导入,在后面课程中我们会大量用到这些命令行。

我们想测试一下Symfony与数据库的连接,我们可以使用doctrine的数据库创建命令,拉到底部输入`symfony console doctrine:database:create`,他这里提示我们不能创建teebblog数据库,错误原因是我们的数据库已经存在了。这就说明我们的数据库连接已经成功了。这样我们的orm组件就已经安装,并且配置完成了。

在下一节,我们将讲解一下测试驱动开发的一些流程。