

我们来看错误页面的提示，它提示我们不能为`post_edit`路由生成链接，因为我们删除了路由`post_edit`对应的action方法。最简单的解决方法就是在`index.html.twig`文件中，我们删除使用`post_edit`路由的代码就可以了。同样的我们删除使用`post_new`路由的这行代码，回到浏览器刷新。

现在我们的文章就已经显示了，文章的所有状态都是已发布的状态。回到项目，现在我们来学习一下Twig模板引擎怎么使用。

在第1行使用`extends`关键字可以实现对模板的继承，我们按着command键鼠标点击`base.html.twig`文件，在`base.html.twig`文件中，我们看到最典型的HTML5代码，非常简单。但是在代码中间有一些使用`{% block title %}`，这样的方式定义的代码，这些代码就是Twig的模板代码。

在Twig模板中使用`block`关键字来定义代码块，我们可以将一些特定功能的HTML代码块定义成一个`block`区块，然后再使用`extends`关键字在子模板中对`block`定义的区块进行覆盖，来让不同的模板显示不同的内容。回到`base.html.twig`文件，这里定义了4个区块，首先是标题区块，然后是样式代码区块，然后JS代码区块，最后是body区块。这样在`index.html.twig`中，通过`extends`关键字继承`base.html.twig`文件后，我们可以在子模板中定义一些区块来覆盖父模板区块中的内容。

在body区块中，我们看第23行代码，Twig模板引擎也提供了循环遍历的关键字，使用`for in`来进行循环遍历。在`for`循环中也可以使用`else`，如果`for`循环中没有内容的话，就会执行`else`的代码，最后使用`endfor`来结束这个循环。

我们来修改`index.html.twig`文件，我们让首页的标题为`Teebblog`，然后h1标签这里我们输入`Teebblog List`，我们来看`for`循环的循环体，对于Twig文件中的常规类型的变量，比如说字符串类型或者数值类型，使用双大括号的方式来显示变量的值，为了方便在Twig模板中对页面的变量进行查看，Twig引擎提供了一个`{{ dump() }}`方法，我们可以在`dump()`方法中传入要检查的变量。

我们这里传入`post`，回到浏览器刷新页面，这样`for`循环中的`post`变量都挨个显示了。注释这行代码，在前面的课程中，我们使用`make:test`命令行创建了单元测试代码和集成测试代码。

在下节课中我们将使用`make:test`命令行创建功能测试代码，来测试我们的Twig模板代码是否可以按照我们的预期进行显示。