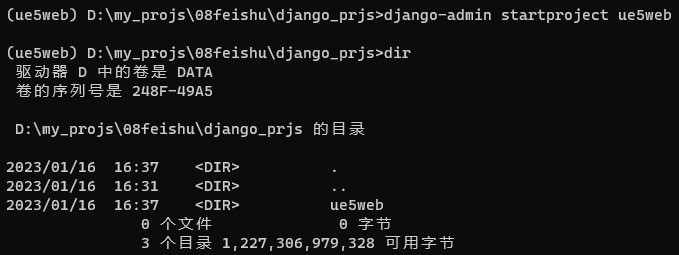
**用Django创建ue5的教程网站**

**创建虚拟环境**

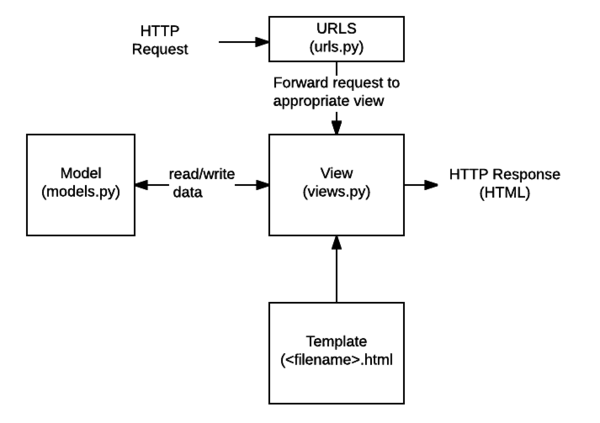
* √在Pycharm中创建虚拟环境 ue5web
* 在虚拟环境中 conda install安装 django
* √激活环境、创建项目

|  |
| --- |
| Bash D:\my\_projs\08feishu\django\_prjs>conda activate ue5web # 激活虚拟环境 (ue5web) D:\my\_projs\08feishu\django\_prjs>django-admin startproject ue5web # 创建文件夹和项目 |





**urls.py, views.py, models.py 和 Template 的关系**



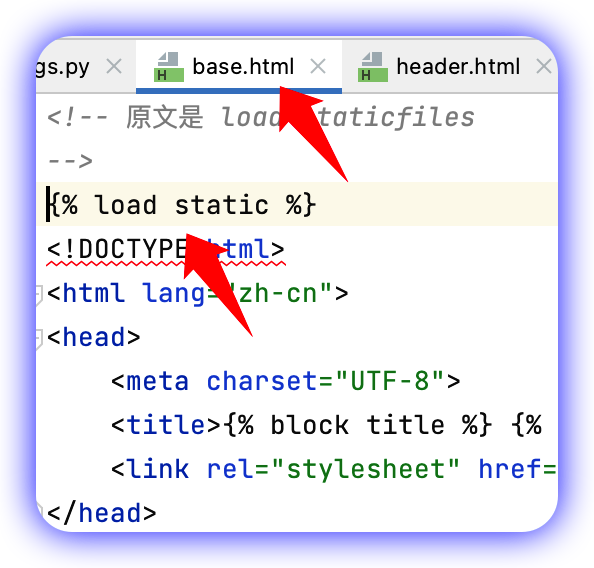
* HTTP Request: 用户访问一个网址, Request 中常常带有参数
* urls.py 根据一定的规则, 把网址传递给匹配的 views.py 函数.
* View 函数会读取 Model 中的数据 + Template, 把页面呈现给客户.
* Model 可以理解为数据库.

**模板**



* 注意↑, 这种写法很聪明.

**静态文件**



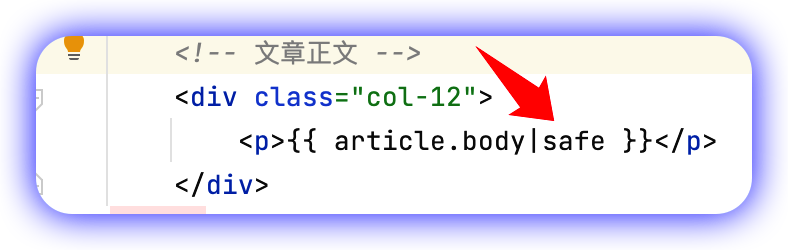
* 在 Django 3 和 4 中, 加载静态文件的语法是 load static, 而不是 load staticfiles .

**HTML页面Source的乱码问题**

**用safe过滤**

* html代码

|  |
| --- |
| HTML <p class="card-text">{{ article.body|slice:'100'|safe }}...</p> |



* ↑ 这里要用上 safe , 才能让 Django 正确渲染Body 中的 HTML 标记.

**slice的问题**



|  |
| --- |
| HTML <p class="card-text">{{ article.body|slice:'100'|safe }}...</p> |

**CheckBox 中的对钩不变色**

* 里面的对钩用了一个特殊的字符, 这个字符不支持改变颜色. 可以暂时把对钩用字母 V 表示.
* 对钩问题是 CheckGet的一个大重点。以后可以进一步完善.
* 例如, 可以在后台统计学员是否 Check 了对钩.

**mammoth**

* 可以用Python 的 mammoth把 docx 文档转换为 html, 要在虚拟环境的 terminal 下执行命令↓

|  |
| --- |
| Bash # 虚拟环境/路径/命令 **(ue5web) maxmacboookpro2019@myPro doc2html %** mammoth 'blueprints初级教程-1-崇礼.docx' --output-dir='imgs' |