

首页搭建



导航栏

1、功能需求：

- 用户未登录时右边栏展现以下字样：
 - 注册
 - 登录
- 如果用户登录，右边栏展示：
 - 用户名
 - 更多操作 #点击更多操作，可以展开
 - 修改密码
 - 修改头像
 - 后台登录
 - 退出登录
 - 点击博客可以跳转到个人站点首页，如果用户未登录，跳转到登录页面

2、技术实现-前端：

- Bootstrap框架的导航条组件：<nav class="navbar navbar-default">
- 下拉菜单中：修改密码使用了Bootstrap中的JavaScript插件中的模态框。
 - 在模态框中使用了Ajax向后端提交数据
- 通过{{ request.auth.is_authenticated }}的值判断用户是否登录。
 - 如果登录显示用户名，以及下拉菜单。否则显示注册和登录字样。

3、技术实现-后端：

- 更改密码：
 - is_right=request.user.check_password(old_password) #判断原密码是否正确
 - request.user.set_password(new_password) #更改用户的密码,但是数据没有更改
 - request.user.save() #同时更改数据库，相当于将现在user的属性同步到数据库

- `return JsonResponse(bakc_dic)`
- 退出登录：
 - `auth.logout(request)`

Django后台管理admin使用

1、登录需要使用超级管理员才可以登录

- `python manage.py createsuperuser` #创建超级管理员

2、django的admin后台管理，提供了对数据库的基本操作功能，比如增删改查。

- 但是要对数据库进行管理，需要在admin.py中进行数据库表的注册才行。
- admin.py:
 - `admin.site.register(models.UserInfo)` #将UserInfo表注册到admin后台管理
 - `admin.site.register(models.Blog)` #将Blog表注册到admin后台管理
 - `admin.site.register(models.Category)` #将category表注册得到admin后台管理
- 注册后即可在后台对数据库表进行增删改查。
- 后台增加一个表记录时，有段的字段会要求必须填写，如果不想填写，需要的后端对应的model中的字段进行标注：
 - `phone=models.BigIntegerField(verbose_name="手机号",null=True,blank=True)`
 - `null=True` #表示数据库字段可以为空
 - `blank=True` #表示admin后台字段可以为空
- 为了后面页面搭建，有良好的展示效果，我们需要实现向数据库中加入一些文章内容；

这时候就可以利用admin后台管理了。添加内容时，主要有相关的外键，和表都需要完善清楚。

3、注册完后admin中显示的表名，一般会默认加s.如果想自定义显示表名：

- `class UserInfo(models.Model):`
 - `username=models.CharField(max_length=32)`

- class Meta:
 - verbose_name_plural="用户信息" #定义admin后台管理中默认显示的表名

网站首页搭建

1、功能需要:

- 总体布局使用Bootstrap框架，整体使用2-8-2的布局方式
- 两边个站两列使用bootstrap中面板组件，可用于存放广告
- 中间是首页的主要内容
- 使用分页器显示主要内容
- 内容展示包含：
 - 文章标题
 - 文章描述（简介）
 - 显示作者和发布时间
 - 显示评论数和点赞数
 - 显示作者头像

2、技术实现-前端:

- 广告部分使用Bootstrap面板组件：<div class="panel panel-danger">
- 主要内容使用Bootstrap媒体对象组件：<ul class="media-list">
- 头像展示，使用media文件下的头像文件，这需要后端进行相应media配置（媒体配置）
 -
- 作者、发布时间、评论数，点赞数等均使用标签实现
- 发布日期使用了date过滤器。
- 点赞和评论图标使用了Bootstrap的字体图标

3、技术实现-后端:

- url配置：
 - re_path(r"^home/",views.home,name="home")
 - re_path(r"^\$",views.home) #网站默认首页

- media配置：
 - 网页的一些静态资源(css文件, js文件), 需要放在单独的静态文件下, 才可以进行网络访问。
 - 那么客户的一些静态资源(头像文件,)也需要放在特定的文件下, 这就需要进行media配置。
 - media配置, 就是将后台的某一个文件夹, 开放权限, 可以通过网络直接进行访问。
 - media配置步骤:
 - MEDIA_ROOT=os.path.join(BASE_DIR,"media")
 - 在settings.py中, 注册需要开放权限的文件夹。
 - re_path(r"^media/(?P<path>.*)",serve, {"document_root":settings.MEDIA_ROOT}),
 - 在urls.py中, 为文件夹配置好对用的url访问规则。
 - 重启django服务, 系统就会在项目下自动创建一个名为media的文件。
- 配置好media文件后, 将头像文件夹, 移动到此文件夹下。
- 使用分页器显示文章内容
- def home(request):
 - article_query_set=models.Article.objects.all() #获取所有的文章内容
 - paginator=Paginator(article_query_set,3) #将所有的文章内容封装在分页器内
 - current_page=paginator.page(num) #获取到其中的某一页的数据, 然后传给前端。

图片防盗链

- # 如何避免别的网站直接通过本网站的url访问本网站资源
- # 简单的防盗
 - 我可以做到请求来的时候先看看当前请求是从哪个网站过来的
 - 如果是本网站那么正常访问
 - 如果是其他网站直接拒绝
 - 请求头里面有一个专门记录请求来自于哪个网址的参数

- Referer: <http://127.0.0.1:8000/xxx/>
- # 如何避免
 - 1.要么修改请求头referer
 - 2.直接写爬虫把对方网址的所有资源直接下载到我们的服务器上

内容回顾

1) 首页搭建

```
1
2 # django admin后台管理
3 """
4 urls.py自带的那个url
5
6 特点:
7 能够自动生成注册了的模型表的增删改查四条url及相应的界面
8
9 使用
10 在对应的应用下的admin.py文件中注册模型表即可
11     admin.site.register(models.UserInfo)
12     ...
13 class UserInfo(models.Model):
14     username = models.CharField(verbose_name='字段名')
15     class Meta:
16         verbose_name_plural = '表名'
17
18 针对bbs在绑定数据的时候一定要细心
19     1.先去文章表里面绑定数据
20     2.个人站点
21     3.文章分类
22     4.将用户和个人站点绑定关系
23     5.文章与文章标签
24
25 扩展:admin路由分发的本质
26     # 路由分发本质    include    可以无限制的嵌套N多层
27     url(r'^index/', ([], None, None))
28
29     # url(r'^index/', ([
```

```

30     #         url(r'^index_1/',([
31         #             url(r'^index_1_1',index),
32         #             url(r'^index_1_2',index),
33         #             url(r'^index_1_3',index),
34         #             ],None,None)),
35     #         url(r'^index_2/',index),
36     #         url(r'^index_3/',index),
37     #         ],None,None)),
38     """

```

2) media配置

```

1     """
2     网址所使用到的静态资源默认都是放在static文件夹下
3     用户上传的静态文件资源也应该单独找个位置存放
4
5     media配置
6         settings.py
7             MEDIA_ROOT = os.path.join(BASE_DIR,'media')
8     用户上传文件会自动创建media文件夹然后在该文件夹内存储数据
9     """

```

3) 自定义暴露后端资源

```

1     # 你需要自己在urls.py中书写代码
2     from django.views.static import serve
3     from BBS14 import settings
4
5     # 固定写法 不要自己改动
6     url(r'^media/(?P<path>.*)',serve,{ 'document_root':settings.MEDIA_ROOT}),

```

urls.py

```

1     path('admin/', admin.site.urls),
2     #首页
3     re_path(r"^home/",views.home,name="home"),

```

```
4 re_path(r"'^$',views.home),
5 #暴露后端指定文件夹资源
6 re_path(r"^media/(?P<path>.*)",serve,{"document_root":settings.MEDIA_ROOT}),
7 #个人站点页面搭建
8 re_path(r"^(?P<username>\w+)/$",views.site,name="site"),
```

1、导航栏

1)templates.home.html

```
1 <nav class="navbar navbar-inverse">
2     <div class="container-fluid">
3         <!-- Brand and toggle get grouped for better mobile display -->
4         <div class="navbar-header">
5             <button type="button" class="navbar-toggle collapsed" data-toggle="collapse"
6 data-target="#bs-example-navbar-collapse-1" aria-expanded="false">
7                 <span class="sr-only">Toggle navigation</span>
8                 <span class="icon-bar"></span>
9                 <span class="icon-bar"></span>
10                <span class="icon-bar"></span>
11            </button>
12            <a class="navbar-brand" href="#">BBS</a>
13        </div>
14
15        <!-- Collect the nav links, forms, and other content for toggling -->
16        <div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">
17            <ul class="nav navbar-nav">
18                <li class="active"><a href="#">博客 <span class="sr-only">(current)
19</span></a></li>
20                <li><a href="#">文章</a></li>
21                <li class="dropdown">
22                    <a href="#" class="dropdown-toggle" data-toggle="dropdown"
23role="button" aria-haspopup="true" aria-expanded="false">更多 <span class="caret">
24</span></a>
25                    <ul class="dropdown-menu">
26                        <li><a href="#">Action</a></li>
27                        <li><a href="#">Another action</a></li>
28                        <li><a href="#">Something else here</a></li>
29                        <li role="separator" class="divider"></li>
30                        <li><a href="#">Separated link</a></li>
```



```
62         <label for="">原密码</label>&nbsp;&nbsp;&nbsp;</span>  
    <span id="error_password" style="color: red"></span>  
63         <input type="password"  
id="id_old_password" class="form-control">  
64     </div>  
65     <div class="form-group">  
66         <label for="">新密码</label>  
67         <input type="password"  
id="id_new_password" class="form-control">  
68     </div>  
69     <div class="form-group">  
70         <label for="">确认密码71         <input type="password"  
id="id_confirm_password" class="form-control">  
72     </div>  
73     <div class="modal-footer">  
74         <button type="button" class="btn btn-  
default" data-dismiss="modal">取消</button>  
75         <button type="button" class="btn btn-  
primary" id="id_edit">修改</button>  
76     </div>  
77     <br>  
78     <br>  
79 </div>  
  
81 </div>  
  
83 </div>  
84 </div>  
85 </div>  
  
86 </li>  
87 {% else %} <!--如果没有登录，显示注册和登录字样-->  
88     <li><a href="{% url "reg" %}">注册</a></li> <!--使用的是url别名-->  
89     <li><a href="{% url "login" %}">登录</a></li>  
90 </li>  
91 {% endif %}  
92 </ul>  
93 </div><!-- /.navbar-collapse -->  
94 </div><!-- /.container-fluid -->  
95 </nav>
```

```

97 <script>
98     //*****修改密码
99     *****8
100     $("#id_edit").click(function () {
101         $.ajax({
102             url:"/set_password/",
103             type:"post",
104             data:{
105                 "old_password":$("#id_old_password").val(),
106                 "new_password":$("#id_new_password").val(),
107                 "confirm_password":$("#id_confirm_password").val(),
108                 "csrfmiddlewaretoken":"{{ csrf_token }}"
109             },
110             success:function (args) {
111                 if(args.code==1000){
112                     window.location.reload()    //刷新当前页面
113                 }
114                 if (args.code==2000){
115                     $("#error_confirm_password").text(args.msg)
116                 }else {
117                     $("#error_password").text(args.msg)
118                 }
119             })
120         })
121 </script>

```

2)app01.views.py

```

1 #####更改密码#####
2 @login_required()
3 def set_password(request):
4     if True:
5         back_dic={"code":1000,"msg":""}
6         if request.method=="POST":
7             old_password=request.POST.get("old_password")
8             new_password=request.POST.get("new_password")
9             confirm_password=request.POST.get("confirm_password")
10            is_right=request.user.check_password(old_password)
11            if is_right:

```

```

12         if new_password==confirm_password:
13             request.user.set_password(new_password)
14             request.user.save()
15             back_dic["msg"]="密码修改成功"
16         else:
17             back_dic["code"]=2000
18             back_dic["msg"]="两次密码不一致"
19     else:
20         back_dic["code"]=3000
21         back_dic["msg"]="原密码错误"
22     return JsonResponse(back_dic)
23     return HttpResponse("set_pwd")
24
25 #####退出登录#####
26 @login_required()
27 def logout(request):
28     auth.logout(request)
29     return redirect("/home/")

```

• 导航条展示



• 点击修改密码，模态框展示

修改密码

用户名

jason

原密码

新密码

确认密码

取消

修改

2、网站首页

1)templates.home.html

```
1 <div class="container-fluid">  <!-- 总体布局 2-8-2 -->
2 <!--#####左边广告部分
   ##### -->
3     <div class="col-md-2">
4         <div class="panel panel-danger">
5             <div class="panel-heading">
6                 <h3 class="panel-title">重金求子</h3>
7             </div>
8             <div class="panel-body">
9                 cskdcjsk几点开始
10            </div>
11        </div>
12        <div class="panel panel-info">
13            <div class="panel-heading">
14                <h3 class="panel-title">千万大奖</h3>
15            </div>
16            <div class="panel-body">
17                抓紧联系：182132823
```

```

18         </div>
19     </div>
20     <div class="panel panel-default">
21         <div class="panel-heading">
22             <h3 class="panel-title">美女直播</h3>
23         </div>
24         <div class="panel-body">
25             性感荷官在线发牌，你还在等什么
26         </div>
27     </div>
28 </div>
29
30 <!--#####中间首页主要内容##### -->
31 <div class="col-md-8">
32     <!--#####展示利用分页器展示网页内容#####-->
33     {% for article_obj in current_page %}         <!--current_page是一个分页-->
34         <!--媒体对象-->
35         <ul class="media-list">
36             <li class="media">
37                 <h4 class="media-heading"><a href="">{{ article_obj.title }}</a>
38 </h4>
39                 <div class="media-left">
40                     <a href="#">
41                         
43                     </a>
44                 </div>
45                 <div class="media-body">
46                     {{ article_obj.desc }}
47                 </div>
48                 <br>
49                 <div>
50                     <span><a href="#">{{ article_obj.blog.userinfo.username
51 }}&nbsp;&nbsp;&nbsp;</a></span>
52                     <span>发布于</span>
53                     <span>{{ article_obj.create_time|date:"Y-m-d"}}&nbsp;&nbsp;&nbsp;
54 </span>
55                     <span class="glyphicon glyphicon-comment">评论数{{
56 article_obj.comment_num }}&nbsp;&nbsp;&nbsp;</span>
57                     <span class="glyphicon glyphicon-thumbs-up">点赞数{{
58 article_obj.up_num }}&nbsp;&nbsp;&nbsp;</span>

```

```

53         </div>
54     </li>
55 </ul>
56 <hr>
57 {% endfor %}
58
59 <!--#####分页器标签显示#####-->
60 <nav aria-label="Page navigation">
61     <ul class="pagination">
62         <li>                                <!--显示上一页-->
63             {% if current_page.has_previous %}
64                 <a href="?page={{ current_page.previous_page_number }}" aria-
label="Previous">
65             {% else %}
66                 <a class="disabled" href="" aria-label="Previous">
67             {% endif %}
68             <span aria-hidden="true">上一页</span>
69             </a>
70         </li>
71
72         {% for item in page_range %}    <!--显示中间的页码-->
73             {% if num == item %}
74                 <li class="active"><a href="?page={{ item }}">{{ item }}</a>
</li>
75             {% else %}
76                 <li><a href="?page={{ item }}">{{ item }}</a> </li>
77             {% endif %}
78         {% endfor %}
79
80         <li>                                <!--显示下一页-->
81             {% if current_page.has_next %}    <!--判断当前页面是否有下一页-->
82                 <a href="?page={{ current_page.next_page_number }}" aria-
label="Next">    <!--有则加上下一页的链接-->
83             {% else %}
84                 <a class="disabled" href="" aria-label="Next">        <!--没有则链
接失效-->
85             {% endif %}
86             <span aria-hidden="true">下一页</span>
87             </a>
88         </li>
89     </ul>

```

```

90         </nav>
91     </div>
92
93     <!--#####右边广告部分##### -->
94     <div class="col-md-2">
95         <div class="panel panel-danger">
96             <div class="panel-heading">
97                 <h3 class="panel-title">重金求子</h3>
98             </div>
99             <div class="panel-body">
100                 cskdcjsk几点开始
101             </div>
102         </div>
103         <div class="panel panel-info">
104             <div class="panel-heading">
105                 <h3 class="panel-title">千万大奖</h3>
106             </div>
107             <div class="panel-body">
108                 抓紧联系：182132823
109             </div>
110         </div>
111         <div class="panel panel-default">
112             <div class="panel-heading">
113                 <h3 class="panel-title">美女直播</h3>
114             </div>
115             <div class="panel-body">
116                 性感荷官在线发牌，你还在等什么
117             </div>
118         </div>
119     </div>
120 </div>

```

2)app01.views.py

```

1  def home(request):
2      #查询本网站的所有的文章数据展示到前端页面，这里可以使用分页器
3      article_query_set=models.Article.objects.all()
4
5      #使用分页器

```

```

6     num=request.GET.get("page")
7     if not num: #如果num为空
8         num="1"
9     num=int(num)
10    paginator=Paginator(article_query_set,3)
11    page_range=paginator.page_range    #获取到分页范围
12    if paginator.num_pages>11:
13        right=num+6
14        left=num-5
15        if right>paginator.num_pages:
16            right=paginator.num_pages
17        if left<=0:
18            left=1
19        page_range=range(left,right)
20    try:
21        current_page=paginator.page(num)    #获取到当前的分页内容
22    except EmptyPage as e:
23        current_page=paginator.page(1)    #如果出错默认显示第一页的内容
24    return render(request,"home.html",locals())

```

• 网页展示



