

5. django上传文件

通过编写WEB前端代码， 将文件通过网页上传至服务器。

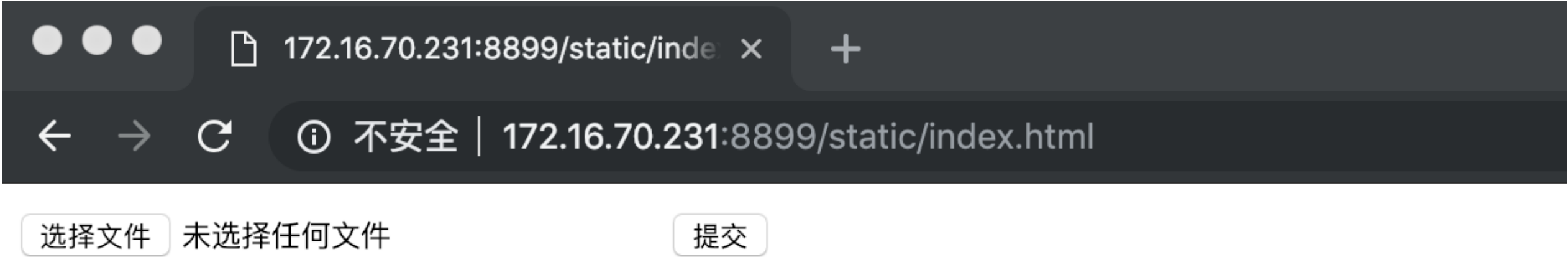
一. django上传文件

1.1 临时csrf, 在项目配置文件Settings中, 找到如下配置, 将CSRF部分内容注释掉.

```
MIDDLEWARE = [  
    'django.middleware.security.SecurityMiddleware',  
    'django.contrib.sessions.middleware.SessionMiddleware',  
    'django.middleware.common.CommonMiddleware',  
    # 'django.middleware.csrf.CsrfViewMiddleware',  
    'django.contrib.auth.middleware.AuthenticationMiddleware',  
    'django.contrib.messages.middleware.MessageMiddleware',  
    'django.middleware.clickjacking.XFrameOptionsMiddleware',  
]
```

1.2 开发WEB前端代码.

```
<form action="/upload/" method="post" enctype="multipart/form-data">  
    <input type="file" name="files">  
    <input type="submit">  
</form>
```



html表单请求url /upload/, HTTP方法为POST

1.3 django后台代码开发:

```
from django.conf import settings  
def file_upload(request):  
    if request.method == 'POST':  
        static_folder = os.path.join(settings.STATIC_ROOT, 'multipart')  
        if not os.path.exists(static_folder):  
            os.makedirs(static_folder)  
        obj = request.FILES.get('files')  
        f = open(os.path.join(static_folder, obj.name), 'wb')  
        for chunk in obj.chunks():  
            f.write(chunk)  
        f.close()  
        return HttpResponse('OK')
```

1.4 将URL与视图函数映射, 修改项目urls.py文件

```
urlpatterns = [  
    path('admin/', admin.site.urls),  
    path('', include('login.urls')),  
    path('addclass/', login_views.AddClass.as_view()),  
    path('upload/', login_views.file_upload), ##新增  
]
```

1.5 测试静态文件上传:



1.5.1 查看django WEB项目静态资源目录

```
(webapps) [root@harbor-a multipart]# ls  
1111.jpg Python课程模板.xlsx  
(webapps) [root@harbor-a multipart]# pwd  
/app_shell/python/webapps/webapps/static/multipart
```

静态资源上传成功!

1.5.2 访问静态资源, 通过以下url可以访问到刚刚上传的静态资源文件1111.jpg
http1://172.16.70.231:8899/static/multipart/1111.jpg