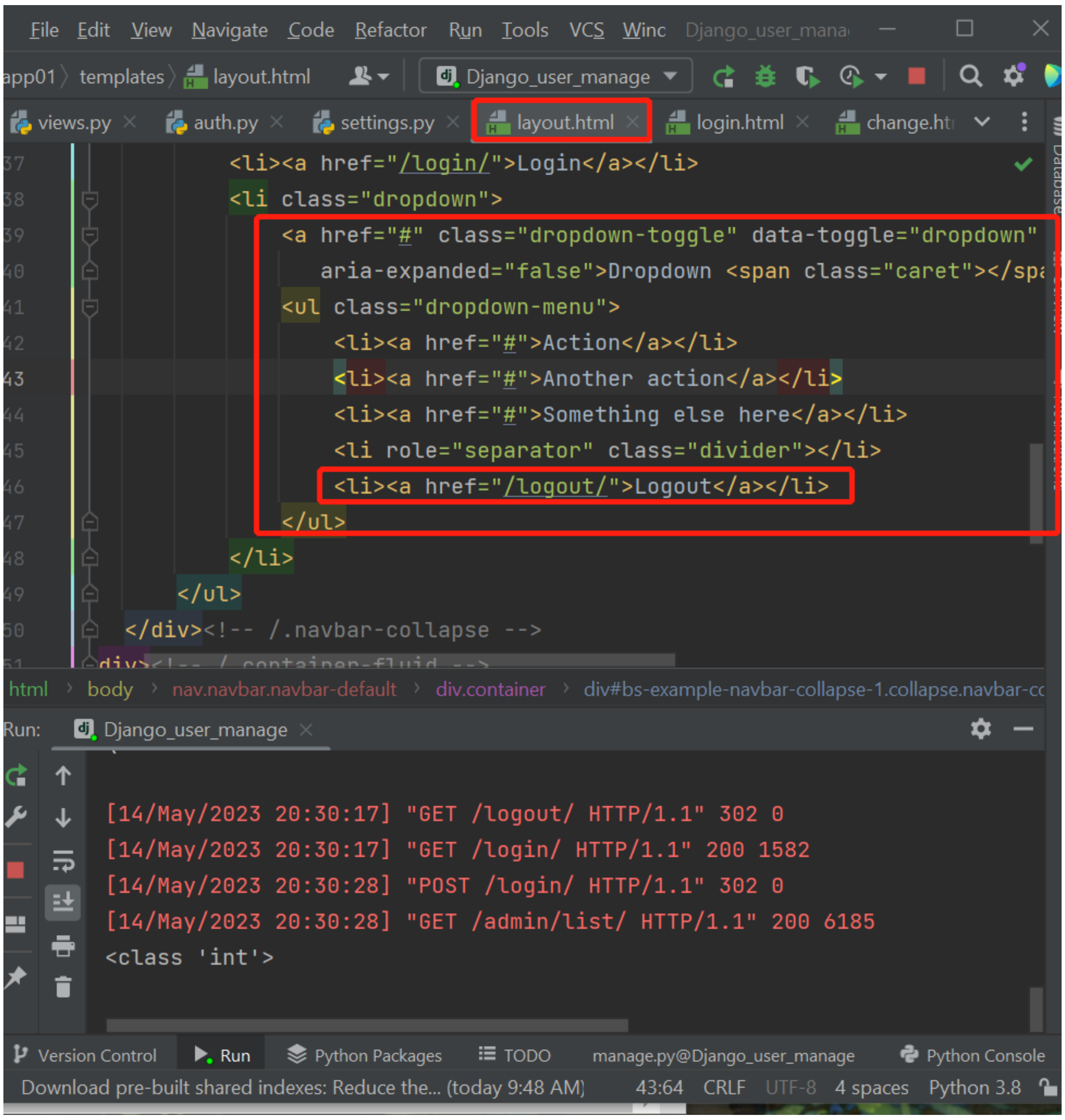
User Management - continue - logout

```
def logout(request):
    request.session.clear()
    return redirect('/login/')
```


File Edit View Navigate Code Refactor Run Tools VCS Winc Django_user_mana
app01 〉 templates 〉 layout.html
Django_user_manage
views.py
auth.py
settings.py
layout.html
login.html
change.ht
html 〉 body 〉 nav.navbar.navbar-default 〉 div.container 〉 div#bs-example-navbar-collapse-1.collapse.navbar-cc
Run: Django_user_manage
Version Control
Run
Python Packages
TODO
manage.py@Django_user_manage
Python Console
Download pre-built shared indexes: Reduce the... (today 9:48 AM)
43:64 CRLF UTF-8 4 spaces Python 3.8

```
                <li><a href="/login/">Login</a></li>
                <li class="dropdown">
                    <a href="#" class="dropdown-toggle" data-toggle="dropdown"
                       aria-expanded="false">Dropdown <span class="caret"></spa
                    <ul class="dropdown-menu">
                        <li><a href="#">Action</a></li>
                        <li><a href="#">Another action</a></li>
                        <li><a href="#">Something else here</a></li>
                        <li role="separator" class="divider"></li>
                        <li><a href="/logout/">Logout</a></li>
                    </ul>
                </li>
            </ul>
        </div><!-- /.navbar-collapse -->
```

```
[14/May/2023 20:30:17] "GET /logout/ HTTP/1.1" 302 0
[14/May/2023 20:30:17] "GET /login/ HTTP/1.1" 200 1582
[14/May/2023 20:30:28] "POST /login/ HTTP/1.1" 302 0
[14/May/2023 20:30:28] "GET /admin/list/ HTTP/1.1" 200 6185
<class 'int'>
```

verification code for login

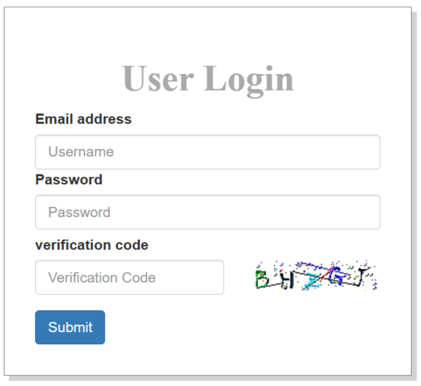
Python生成随机验证码 - 武沛齐 - 博客园

Python生成随机验证码，需要使用PIL模块. 安装： 基本使用 1. 创建图片 2. 创建画笔，用于在图片上画任意内容 3. 画点 4. 画线 5.&#160;画圆 6. 写文本 7. 特殊字体文字 图片验证码 注意：字体文件下载，猛击这里

https://www.cnblogs.com/wupeiqi/articles/5812291.html

```
from app01.utils.code import check_code
from io import BytesIO

def image_code(request):
    img, code_string = check_code()
    print(code_string)
    stream = BytesIO()
    img.save(stream, 'png')
    stream.getvalue()
    return HttpResponse(stream.getvalue())
```

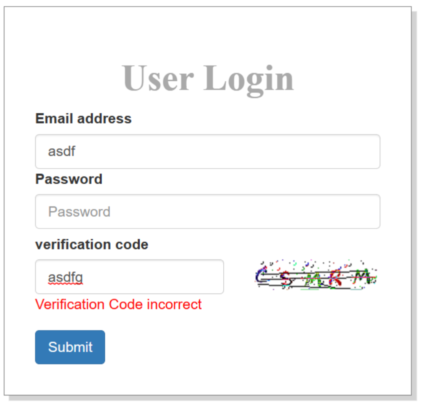
```
<div class="form-group">
            <label for="id_code">verification code</label>
            <div class="row">
                <div class="col-xs-7">
                    <input type="text" name="code" class="form-control" placeholder="Verification Code">
                    <span style="color: red;"></span>
                </div>
                <div class="col-xs-5">
                    <img id="image_code" src="/image/code/">
                </div>
            </div>
        </div>
```



verify

add to session

```
def login(request):
    if request.method == 'GET':
        form = LoginForm()
        return render(request, 'login.html', {'form': form})

    form = LoginForm(data=request.POST)

    if form.is_valid():
        user_input_code = form.cleaned_data.pop('code')
        code = request.session.get('image_code', "")
        if code.upper() != user_input_code.upper():
            form.add_error("code", "Verification Code incorrect")
            return render(request, 'login.html', {'form': form})

        # form.cleaned_data (type:dict)
        # find record in dict where username, password = username, password
        admin_object = models.Admin.objects.filter(**form.cleaned_data).first()

        if not admin_object:
            form.add_error("password", "Username or Password incorrect")
            return render(request, 'login.html', {'form': form})

        # add cookie & session
        request.session['info'] = {'id': admin_object.id, 'name': admin_object.username}
        # reset expiry time
        request.session.set_expiry(60 * 60 * 24 * 7)
        return redirect("/admin/list/")
    return render(request, 'login.html', {'form': form})
```


User Login

Email address

asdf

Password

Password

verification code

asdfg

Verification Code incorrect