

Django 自帶加密方式的使用

- 导入django自带的加密算法模块

```
from django.contrib.auth.hashers import make_password, check_password
```

- make_password(参数1, 参数2, 参数3)
 - 参数1: 需要加密的字符串
 - 参数2: 是否每次都生成不同的加密串, 默认为None, 如果给定任意一个字符串, 则表示每次生成相同的加密串。
 - 参数3: 表示加密算法, 常见的加密算法如下: 'pbkdf2_sha256', 'pbkdf2_sha1', 'bcrypt_sha256', 'unsalted_md5' 等...
- 加密使用:

```
sha_pwd = make_password('1234567890', None, 'pbkdf2_sha256')
print(sha_pwd)
# 类似加密后的结果如下:
pbkdf2_sha256$12000$xzMLhCNvQbb8$i1XDnJIpb/cRRGRX2x7Ym74RNfPRCUp5pbU6Sn+V3J0
=
```

- 密码验证使用:
 - 方法: check_password (参数1, 参数2)
 - 返回值: True 或 False
 - 参数: 参数1: 原始密码, 参数2: 数据库查询出的密码
 - 使用:

```
isSame = check_password('1234567890', sha_pwd )
print(isSame)
# 如果为: True: 表示密码相同, 如果为: False, 则表示密码不相同。
```

Reference

[django自带的加密方法](#)

2019.1.14