072 使用Flask\_Login实现用户登录和权限验证

* 核心技术

Flask\_Login 是Flask框架的一个插件，可以非常方便的管理用户对网站的访问。Flask\_login的常见操作如下：

（1）提供user\_loader()回调函数。使用Flask\_Login时，需要为其提供一个user\_loader()回调函数。user\_loader()函数主要是通过获取user对象存储到session中。

（2）定义User类的属性和方法。如下所示：

* is\_authenticated：用来判断是否是已经授权了，如果通过授权就会返回true
* is\_active：判断是否已经激活
* is\_anonymous：判断是否是匿名用户
* get\_id()：返回用户的唯一标识

这些属性和方法也可以直接继承于userMixin的默认方法和属性，示例代码如下：

1. **class** User(db.Model,UserMixin):
2. id = db.Column(db.Integer, autoincrement=**True**, primary\_key=**True**)
3. username = db.Column(db.String(125), nullable=**False**)
4. email = db.Column(db.String(125), nullable=**False**)
5. password = db.Column(db.String(125), nullable=**False**)

（3）自定义登录过程。当游客访问需要登录的页面时，应提示登录信息，并跳转到的登录页面。Flask\_login提供了配置属性，如下所示：

1. # 实例化LoginManager类
2. login\_manager = LoginManager(app)
3. # 跳转的页面
4. login\_manager.login\_view = 'login'
5. # 提示信息
6. login\_manager.login\_message = "请先登录"
7. # 提示样式
8. login\_manager.login\_message\_category = 'danger'

（4）“记住我”操作。默认情况下，当用户关闭浏览器时，Flask会话被删除，用户注销。 “记住我”可以防止用户在关闭浏览器时意外退出。这并不意味着在用户注销后记住或预先填写登录表单中的用户名或密码。只需将remember = True传递给login\_user调用即可。示例代码如下：

login\_user(user, remember=form.remember.data)