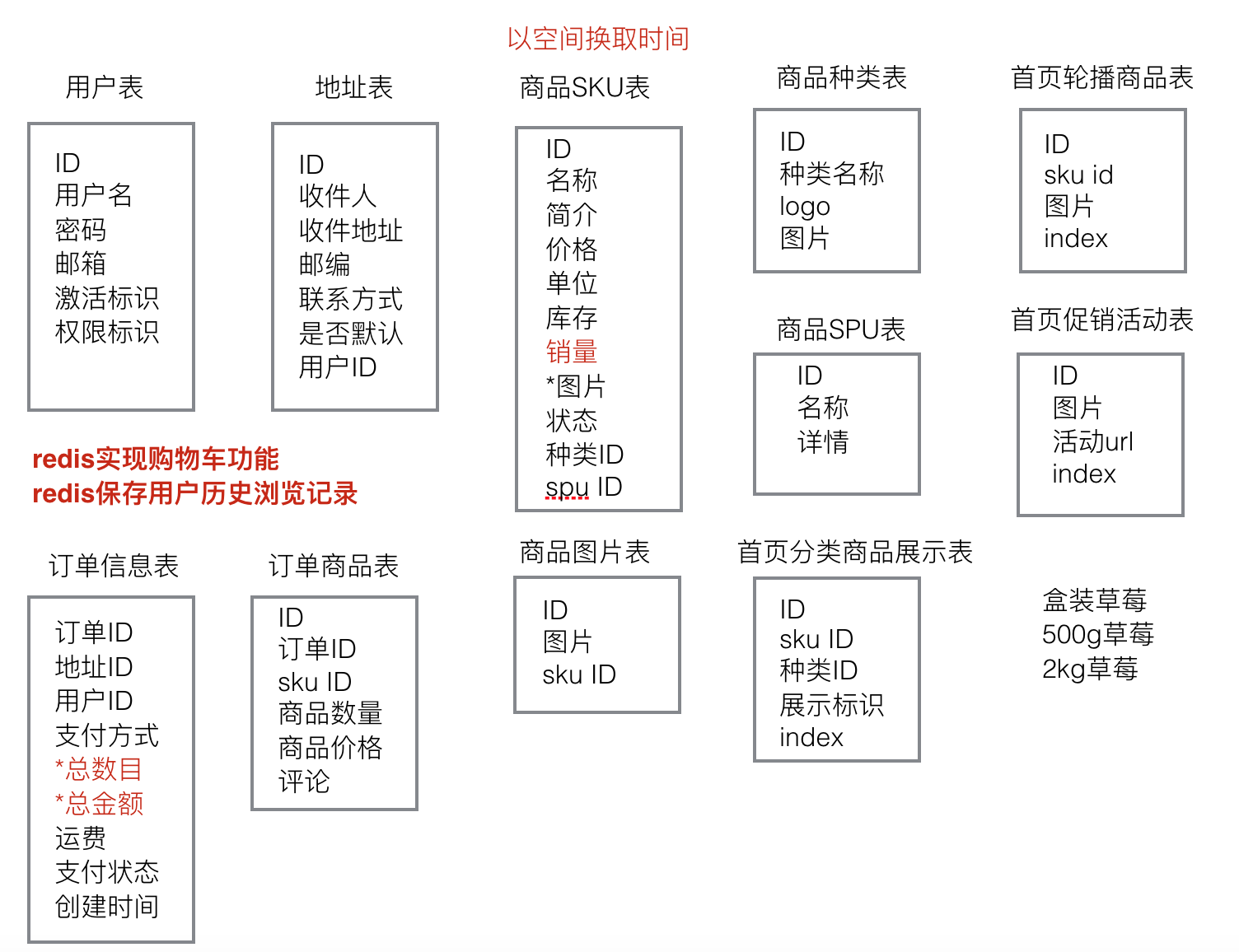
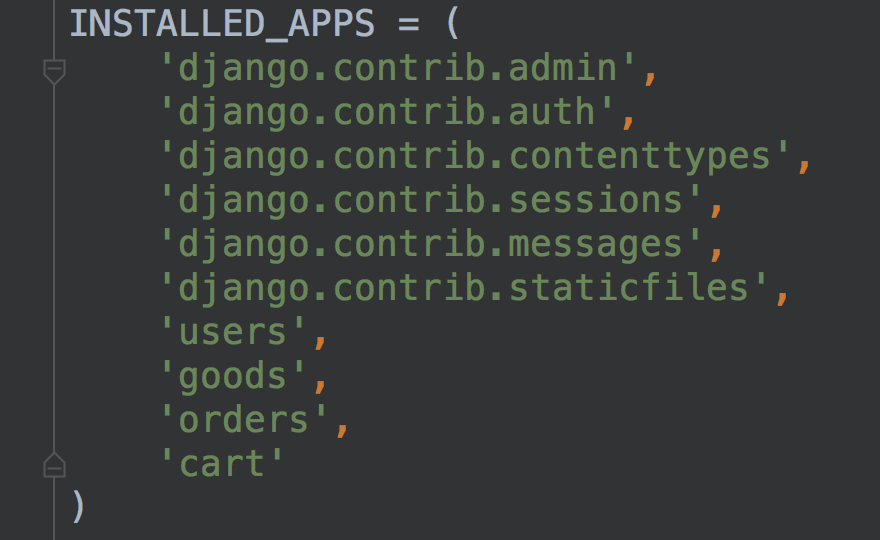
# 项目架构

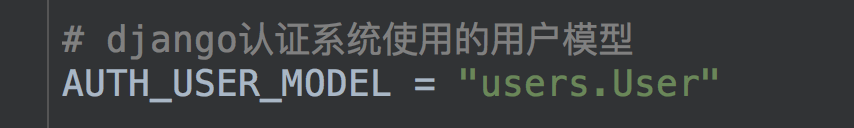


# 数据库表结构



# 用户认证模型





注意： AUTH\_USER\_MODEL配置参数要在第一次迁移数据库之前配置，否则可能django的认证系统工作不正常

# 类视图

将视图view以类的形式定义

通用类视图基类：

django.views.generic.View ( 与django.views.generic.base.View 是同一个)

urls.py中配置路由使用类视图的as\_view()方法

由dispatch()方法具体将请求request分发至对应请求方式的处理方法中（get、post等）

扩展阅读：

<http://python.usyiyi.cn/translate/django_182/topics/class-based-views/intro.html>

类视图资料：

http://python.usyiyi.cn/documents/django\_182/ref/class-based-views/base.html#django.views.generic.base.View.as\_view

# 用户模块开发

## Django认证系统文档

<http://python.usyiyi.cn/documents/django_182/topics/auth/default.html>

**预习方法：**

**create\_user**

**authenticate**

**login**

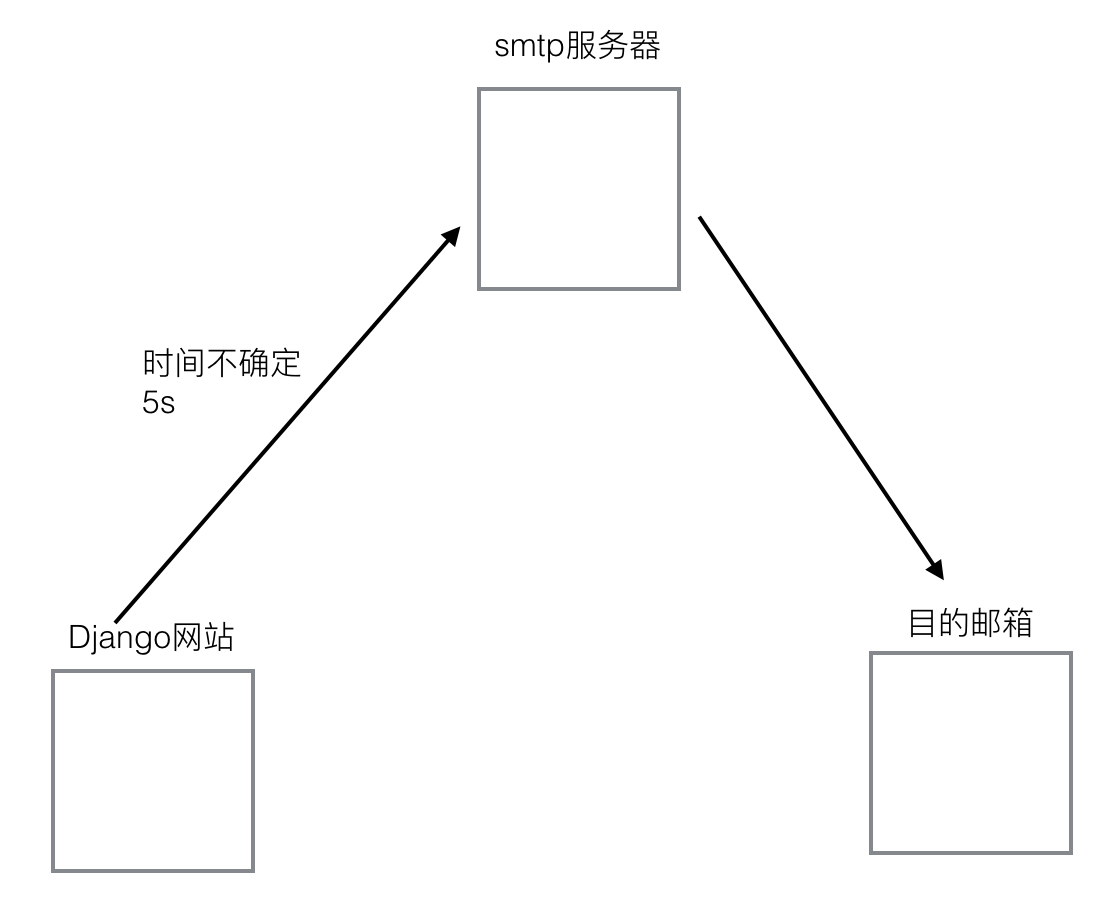
logout

is\_authenticated

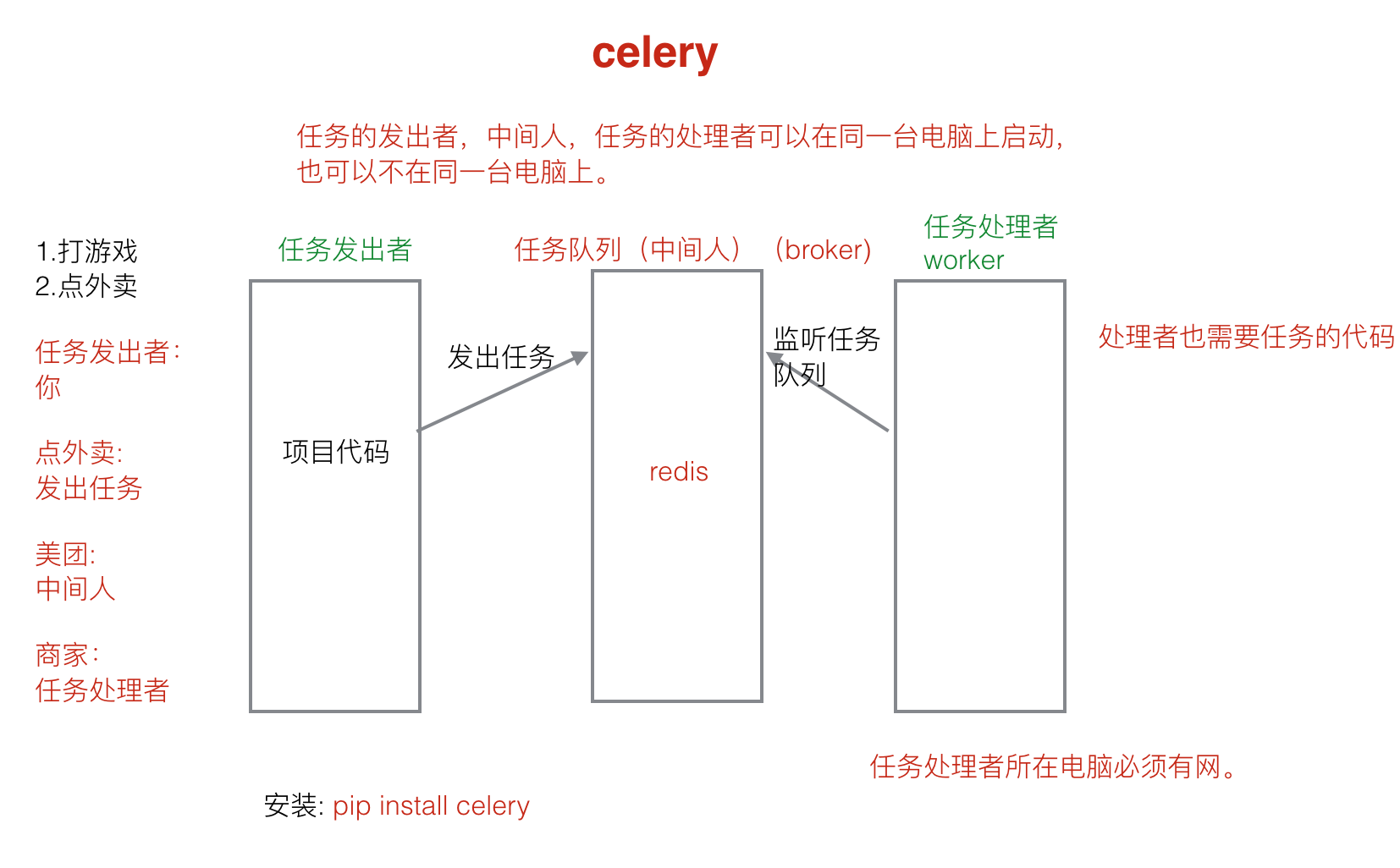
login\_required装饰器

## 用户注册

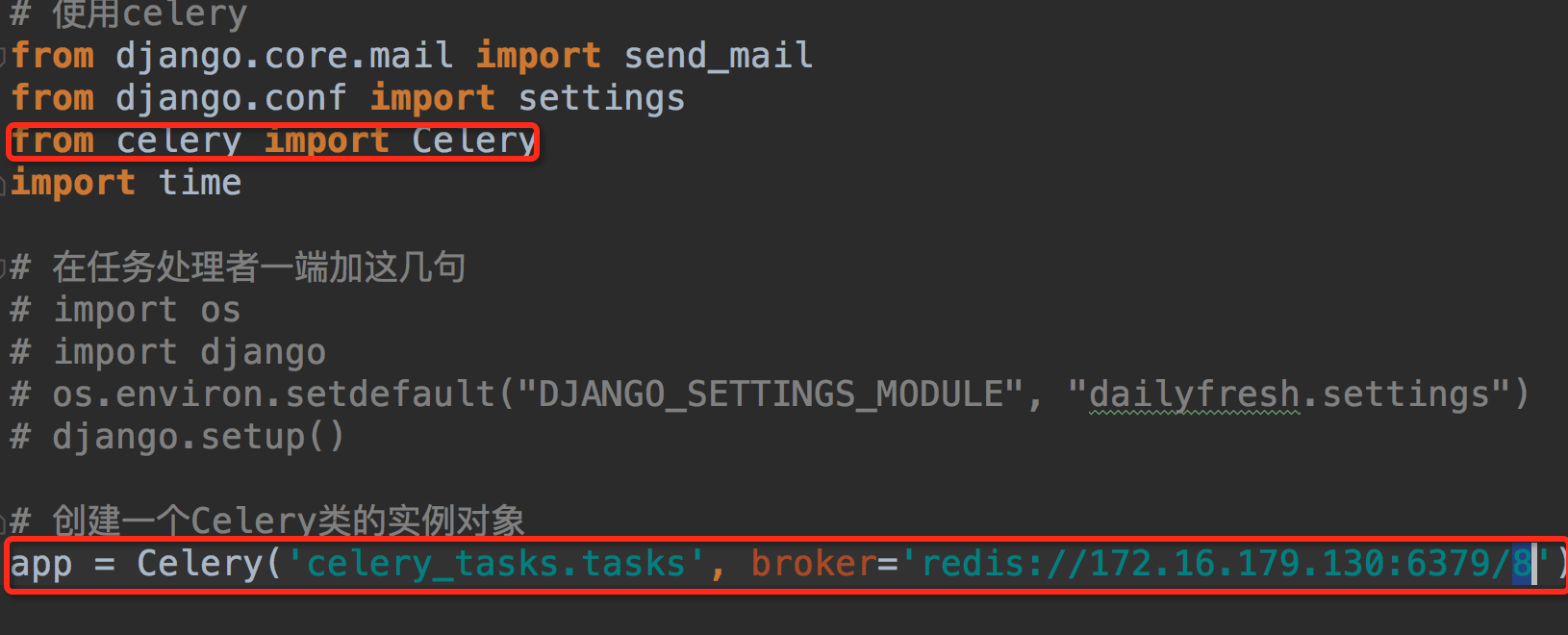
django发送邮件

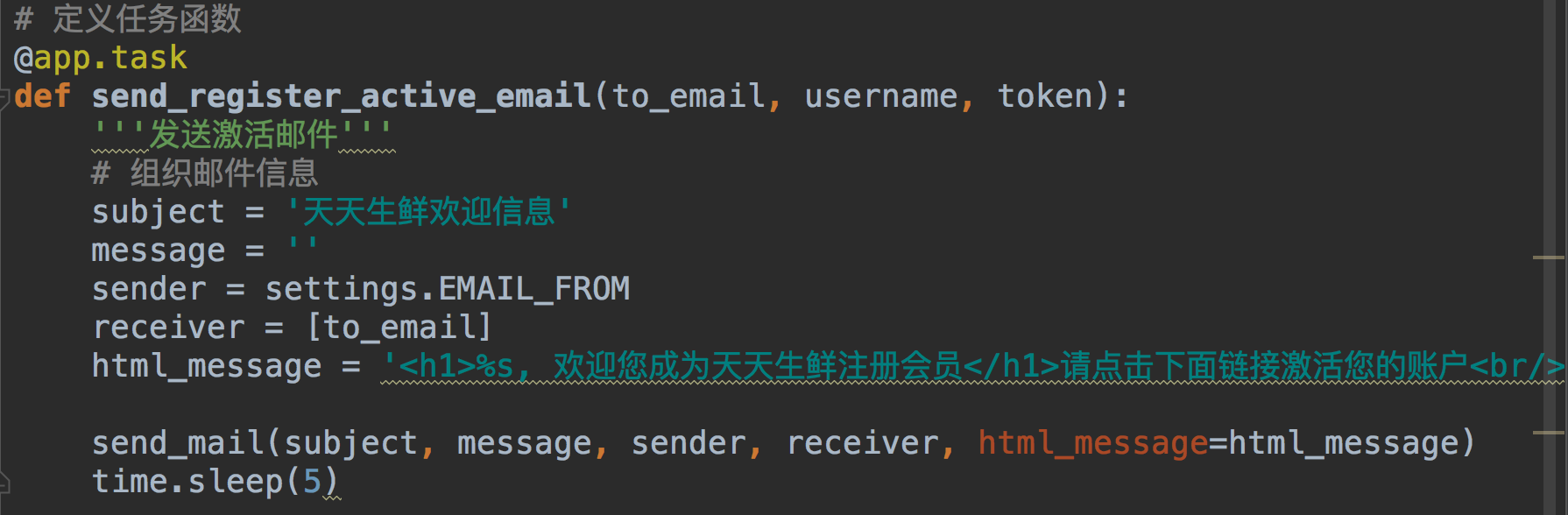


celery

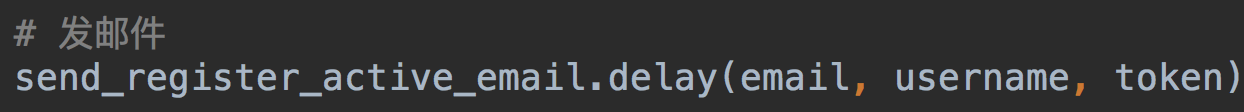


使用：

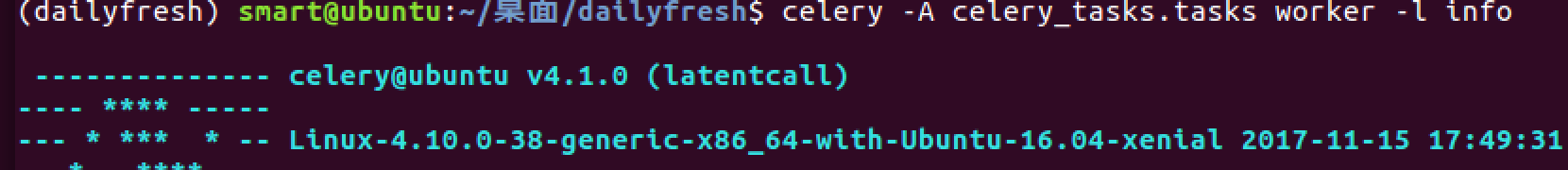




发出任务：



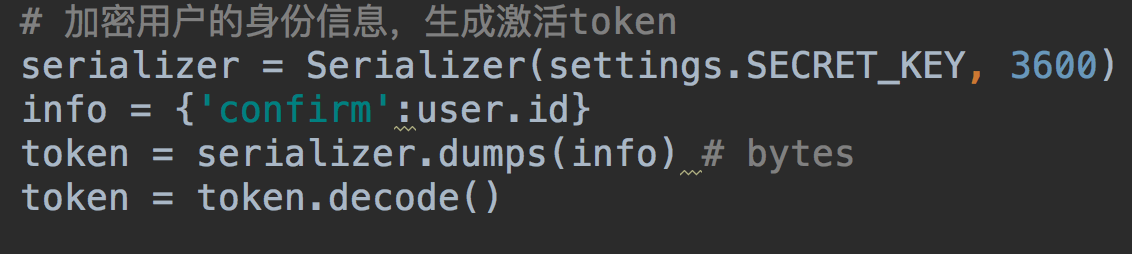
启动worker:



## 用户激活

itsdangerous

加密用户身份信息：



解密用户身份信息:



## 用户登录

## 用户退出

## 用户地址

## 用户个人信息