

1. 代码运行说明

1. 通过[官网](#)或Anaconda安装 python3。
2. 安装 flask 包，打开终端输入：

```
pip install flask
```

3. 在 TradingPlatform 文件夹里打开终端，输入

```
python main.py
```

4. 运行成功后，打开浏览器输入<http://127.0.0.1:8000/>即可访问平台。

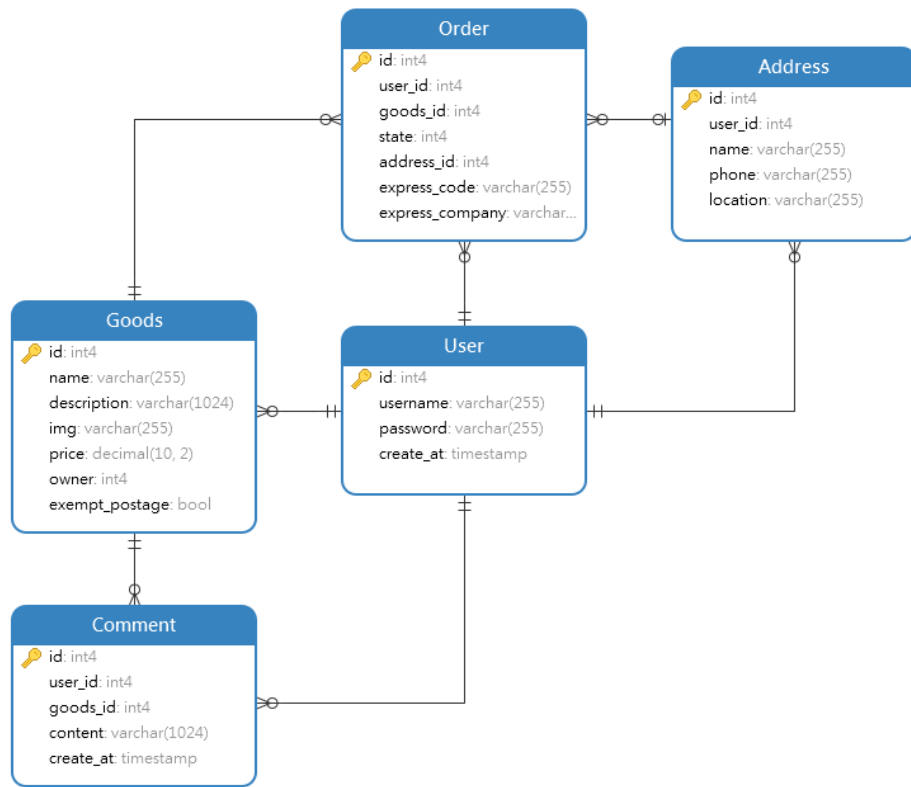
2. 代码结构说明

```
TradingPlatform
├─ db_api.py  # 实现数据库对接逻辑，待补充完善
├─ main.py   # 实现网站的前后端对接逻辑，可根据需要（如想使用自己的数据库设计）进行调整
├─ static    # 存放网站依赖的静态文件
├─ templates # 存放网页模板的文件夹
│   ├─ base.html  # 网页基模板，包括导航栏，网页结构等
│   ├─ goods_detail.html  # 商品详情页
│   ├─ index.html  # 交易大厅页
│   ├─ login.html  # 登录页
│   ├─ manage_address.html  # 收货地址管理页
│   ├─ manage_goods.html  # 商品管理页
│   ├─ manage_order.html  # 订单管理页
│   └─ signup.html  # 注册页
```

作业中需要实现的接口都在 db_api.py 文件中，具体要求见 db_api.py 里的函数注释。

3. 数据库设计说明

平台的代码基于如下物理设计实现，同学们可以根据自身情况选择如下数据库设计或自己的设计。



各数据表的说明如下：

用户表：User

字段	类型	说明
id	int4	用户id，主键
username	varchar(255)	用户名，不允许重名，为unique字段
password	varchar(255)	密码，建议使用md5加密后再存储
create_at	timestamp	用户创建时间

收货地址表：Address

字段	类型	说明
id	int4	收货地址id，主键
user_id	int4	收货地址所属的用户id，外键，指向user表的id
name	varchar(255)	收件人姓名
phone	varchar(255)	收件人手机号码
location	varchar(255)	收件人详细地址

商品表：Goods

字段	类型	说明
id	int4	商品id, 主键
name	varchar(255)	商品名称
description	varchar(1024)	商品描述
img	varchar(255)	商品图片路径
price	decimal(10, 2)	商品价格
owner	int4	商品主人id, 外键, 指向User表的id
exempt_postage	bool	商品是否包邮

订单表: Order

字段	类型	说明
id	int4	订单id, 主键
user_id	int4	发起订单的用户id, 外键, 指向User表的id
goods_id	int4	订单关联的商品id, 外键, 指向Goods表的id
state	int4	订单状态, 包括待同意、待补充信息、待发货、待收货、已完成、被买走
address_id	int4	订单收货地址id, 外键, 指向Address表的id, 在某些订单状态下允许为空
express_code	varchar(255)	订单快递单号, 在某些订单状态下允许为空
express_company	varchar(255)	订单快递公司, 在某些订单状态下允许为空

评论表: Comment

字段	类型	说明
id	int4	评论id, 主键
user_id	int4	发表评论的用户id, 外键, 指向User表的id
goods_id	int4	评论关联的商品id, 外键, 指向Goods表的id
content	varchar(1024)	评论内容
create_at	timestamp	评论发表时间