数据库文件：

由于Django框架的特性，开发后端与数据库时并不直接操作数据库或数据表，而是通过在Django的各个models.py中编写模型并执行迁移数据库以建立数据库，但通过PyCharm的数据库管理插件可以从现有数据库生成SQL代码，因此除去SQL代码外，还附加了后端的模型层Django模型代码。

**一、数据库SQL代码：**

create table pindu.auth\_group

(

id int auto\_increment

primary key,

name varchar(150) not null,

constraint name

unique (name)

);

create table pindu.book\_book

(

isbn varchar(100) not null

primary key,

name varchar(100) not null,

writer varchar(200) not null,

intro varchar(1000) not null,

category varchar(20) not null,

presshouse varchar(50) not null,

pressdate varchar(20) not null,

price double not null,

picture varchar(300) not null,

isflow tinyint(1) not null

);

create table pindu.book\_conditionrate

(

id int auto\_increment

primary key,

`condition` varchar(10) not null,

buy\_rate double not null,

sell\_rate double not null

);

create table pindu.book\_inventory

(

id int auto\_increment

primary key,

book\_num int not null,

book\_isbn varchar(100) not null,

condition\_rate\_id int not null,

constraint book\_inventory\_book\_isbn\_12610312\_fk\_book\_book\_isbn

foreign key (book\_isbn) references pindu.book\_book (isbn),

constraint book\_inventory\_condition\_rate\_id\_0d94baf1\_fk\_book\_cond

foreign key (condition\_rate\_id) references pindu.book\_conditionrate (id)

);

create table pindu.django\_content\_type

(

id int auto\_increment

primary key,

app\_label varchar(100) not null,

model varchar(100) not null,

constraint django\_content\_type\_app\_label\_model\_76bd3d3b\_uniq

unique (app\_label, model)

);

create table pindu.auth\_permission

(

id int auto\_increment

primary key,

name varchar(255) not null,

content\_type\_id int not null,

codename varchar(100) not null,

constraint auth\_permission\_content\_type\_id\_codename\_01ab375a\_uniq

unique (content\_type\_id, codename),

constraint auth\_permission\_content\_type\_id\_2f476e4b\_fk\_django\_co

foreign key (content\_type\_id) references pindu.django\_content\_type (id)

);

create table pindu.auth\_group\_permissions

(

id bigint auto\_increment

primary key,

group\_id int not null,

permission\_id int not null,

constraint auth\_group\_permissions\_group\_id\_permission\_id\_0cd325b0\_uniq

unique (group\_id, permission\_id),

constraint auth\_group\_permissio\_permission\_id\_84c5c92e\_fk\_auth\_perm

foreign key (permission\_id) references pindu.auth\_permission (id),

constraint auth\_group\_permissions\_group\_id\_b120cbf9\_fk\_auth\_group\_id

foreign key (group\_id) references pindu.auth\_group (id)

);

create table pindu.django\_migrations

(

id bigint auto\_increment

primary key,

app varchar(255) not null,

name varchar(255) not null,

applied datetime(6) not null

);

create table pindu.django\_session

(

session\_key varchar(40) not null

primary key,

session\_data longtext not null,

expire\_date datetime(6) not null

);

create index django\_session\_expire\_date\_a5c62663

on pindu.django\_session (expire\_date);

create table pindu.people\_admin

(

id int auto\_increment

primary key,

name varchar(20) not null,

pwd varchar(300) not null,

level int not null,

gender varchar(2) not null,

birth varchar(10) not null,

intro varchar(1000) not null,

phone varchar(11) not null,

username varchar(20) not null

);

create table pindu.people\_user

(

id bigint auto\_increment

primary key,

password varchar(128) not null,

last\_login datetime(6) null,

is\_superuser tinyint(1) not null,

username varchar(150) not null,

first\_name varchar(150) not null,

last\_name varchar(150) not null,

email varchar(254) not null,

is\_staff tinyint(1) not null,

is\_active tinyint(1) not null,

date\_joined datetime(6) not null,

balance double not null,

birth date null,

phone varchar(11) not null,

gender varchar(2) not null,

constraint username

unique (username)

);

create table pindu.authtoken\_token

(

`key` varchar(40) not null

primary key,

created datetime(6) not null,

user\_id bigint not null,

constraint user\_id

unique (user\_id),

constraint authtoken\_token\_user\_id\_35299eff\_fk\_people\_user\_id

foreign key (user\_id) references pindu.people\_user (id)

);

create table pindu.buy\_cart

(

id int auto\_increment

primary key,

book\_num int not null,

inventory\_id int not null,

user\_id bigint not null,

constraint buy\_cart\_inventory\_id\_beab59ba\_fk\_book\_inventory\_id

foreign key (inventory\_id) references pindu.book\_inventory (id),

constraint buy\_cart\_user\_id\_b074e327\_fk\_people\_user\_id

foreign key (user\_id) references pindu.people\_user (id)

);

create table pindu.django\_admin\_log

(

id int auto\_increment

primary key,

action\_time datetime(6) not null,

object\_id longtext null,

object\_repr varchar(200) not null,

action\_flag smallint unsigned not null,

change\_message longtext not null,

content\_type\_id int null,

user\_id bigint not null,

constraint django\_admin\_log\_content\_type\_id\_c4bce8eb\_fk\_django\_co

foreign key (content\_type\_id) references pindu.django\_content\_type (id),

constraint django\_admin\_log\_user\_id\_c564eba6\_fk\_people\_user\_id

foreign key (user\_id) references pindu.people\_user (id),

check (`action\_flag` >= 0)

);

create table pindu.feedback\_feedback

(

id int auto\_increment

primary key,

que\_condition int not null,

que\_content\_user2aftersaleman varchar(300) not null,

que\_content\_aftersaleman2user varchar(300) not null,

que\_content\_assetman2aftersaleman varchar(300) not null,

que\_content\_aftersaleman2assetman varchar(300) not null,

aftersaleman\_id int null,

assetman\_id int null,

user\_id bigint not null,

constraint feedback\_feedback\_aftersaleman\_id\_c246b94f\_fk\_people\_admin\_id

foreign key (aftersaleman\_id) references pindu.people\_admin (id),

constraint feedback\_feedback\_assetman\_id\_f51ecf3f\_fk\_people\_admin\_id

foreign key (assetman\_id) references pindu.people\_admin (id),

constraint feedback\_feedback\_user\_id\_f7dd5014\_fk\_people\_user\_id

foreign key (user\_id) references pindu.people\_user (id)

);

create table pindu.people\_user\_groups

(

id bigint auto\_increment

primary key,

user\_id bigint not null,

group\_id int not null,

constraint people\_user\_groups\_user\_id\_group\_id\_60378d34\_uniq

unique (user\_id, group\_id),

constraint people\_user\_groups\_group\_id\_1f0237d0\_fk\_auth\_group\_id

foreign key (group\_id) references pindu.auth\_group (id),

constraint people\_user\_groups\_user\_id\_290df956\_fk\_people\_user\_id

foreign key (user\_id) references pindu.people\_user (id)

);

create table pindu.people\_user\_user\_permissions

(

id bigint auto\_increment

primary key,

user\_id bigint not null,

permission\_id int not null,

constraint people\_user\_user\_permissions\_user\_id\_permission\_id\_a7ac6748\_uniq

unique (user\_id, permission\_id),

constraint people\_user\_user\_per\_permission\_id\_55f01ef0\_fk\_auth\_perm

foreign key (permission\_id) references pindu.auth\_permission (id),

constraint people\_user\_user\_permissions\_user\_id\_7d19557b\_fk\_people\_user\_id

foreign key (user\_id) references pindu.people\_user (id)

);

create table pindu.people\_useraddress

(

id bigint auto\_increment

primary key,

receiver\_name varchar(20) not null,

receiver\_phone varchar(20) not null,

receiver\_region varchar(20) not null,

receiver\_place varchar(20) not null,

user\_id bigint not null,

constraint people\_useraddress\_user\_id\_13dc8b58\_fk\_people\_user\_id

foreign key (user\_id) references pindu.people\_user (id)

);

create table pindu.buy\_order

(

id int auto\_increment

primary key,

status int not null,

submit\_time datetime(6) not null,

finish\_time datetime(6) null,

user\_id bigint not null,

useraddress\_id bigint not null,

constraint buy\_order\_user\_id\_98a44b79\_fk\_people\_user\_id

foreign key (user\_id) references pindu.people\_user (id),

constraint buy\_order\_useraddress\_id\_462c5a5c\_fk\_people\_useraddress\_id

foreign key (useraddress\_id) references pindu.people\_useraddress (id)

);

create table pindu.buy\_orderdetail

(

id int auto\_increment

primary key,

book\_num int not null,

order\_book\_price double not null,

inventory\_id int not null,

order\_id int not null,

constraint buy\_orderdetail\_inventory\_id\_2c66e3cc\_fk\_book\_inventory\_id

foreign key (inventory\_id) references pindu.book\_inventory (id),

constraint buy\_orderdetail\_order\_id\_3bb7c86c\_fk\_buy\_order\_id

foreign key (order\_id) references pindu.buy\_order (id)

);

create table pindu.sell\_order

(

id int auto\_increment

primary key,

status int not null,

user\_id bigint not null,

finish\_time datetime(6) null,

submit\_time datetime(6) not null,

useraddress\_id bigint not null,

constraint sell\_order\_user\_id\_6e3f5ddb\_fk\_people\_user\_id

foreign key (user\_id) references pindu.people\_user (id),

constraint sell\_order\_useraddress\_id\_e7c1090f\_fk\_people\_useraddress\_id

foreign key (useraddress\_id) references pindu.people\_useraddress (id)

);

create table pindu.sell\_orderdetail

(

id int auto\_increment

primary key,

predict\_price double null,

final\_price double null,

book\_isbn varchar(100) not null,

customer\_condition\_id int not null,

final\_condition\_id int null,

order\_id int not null,

constraint sell\_orderdetail\_book\_isbn\_5d452110\_fk\_book\_book\_isbn

foreign key (book\_isbn) references pindu.book\_book (isbn),

constraint sell\_orderdetail\_customer\_condition\_i\_4fcaacce\_fk\_book\_cond

foreign key (customer\_condition\_id) references pindu.book\_conditionrate (id),

constraint sell\_orderdetail\_final\_condition\_id\_0df4d0c3\_fk\_book\_cond

foreign key (final\_condition\_id) references pindu.book\_conditionrate (id),

constraint sell\_orderdetail\_order\_id\_73b413f5\_fk\_sell\_order\_id

foreign key (order\_id) references pindu.sell\_order (id)

);

create table pindu.sell\_warehouse

(

id int auto\_increment

primary key,

book\_isbn varchar(100) not null,

customer\_condition\_id int not null,

user\_id bigint not null,

constraint sell\_warehouse\_book\_isbn\_9e457a18\_fk\_book\_book\_isbn

foreign key (book\_isbn) references pindu.book\_book (isbn),

constraint sell\_warehouse\_customer\_condition\_i\_4d8c6fc7\_fk\_book\_cond

foreign key (customer\_condition\_id) references pindu.book\_conditionrate (id),

constraint sell\_warehouse\_user\_id\_7d1a8599\_fk\_people\_user\_id

foreign key (user\_id) references pindu.people\_user (id)

);

**二、Django模型层代码**

**（1）book**

class Inventory(models.Model):

id = models.AutoField(primary\_key=True)

book = models.ForeignKey('Book', on\_delete=models.CASCADE, db\_column='book\_isbn')

condition\_rate = models.ForeignKey('ConditionRate', on\_delete=models.CASCADE, related\_name="+")

book\_num = models.IntegerField(default=0)

class Book(models.Model):

isbn = models.CharField(max\_length=100, primary\_key=True)

name = models.CharField(max\_length=100, blank=True)

writer = models.CharField(max\_length=200, blank=True)

intro = models.CharField(max\_length=1000, blank=True)

category = models.CharField(max\_length=20, blank=True)

presshouse = models.CharField(max\_length=50, blank=True)

pressdate = models.CharField(max\_length=20, blank=True)

price = models.FloatField(default=0, blank=True)

picture = models.CharField(max\_length=300, blank=True)

isflow = models.BooleanField(default=True)

class ConditionRate(models.Model):

id = models.AutoField(primary\_key=True)

condition = models.CharField(max\_length=10)

buy\_rate = models.FloatField(default=1)

sell\_rate = models.FloatField(default=1)

（2）buy

class Cart(models.Model):

id = models.AutoField(primary\_key=True)

user = models.ForeignKey('people.User', related\_name='+', on\_delete=models.CASCADE)

inventory = models.ForeignKey('book.Inventory', on\_delete=models.CASCADE) # 一本某个品相的书

book\_num = models.IntegerField(default=1)

class Order(models.Model):

id = models.AutoField(primary\_key=True)

user = models.ForeignKey('people.User', related\_name='+', on\_delete=models.CASCADE)

useraddress = models.ForeignKey('people.UserAddress', related\_name="buy", on\_delete=models.CASCADE)

status = models.IntegerField(default=0)

submit\_time = models.DateTimeField(auto\_now\_add=True)

finish\_time = models.DateTimeField(null=True)

class OrderDetail(models.Model):

id = models.AutoField(primary\_key=True)

order = models.ForeignKey('Order', on\_delete=models.CASCADE)

inventory = models.ForeignKey('book.Inventory', on\_delete=models.CASCADE)

book\_num = models.IntegerField(default=1)

order\_book\_price = models.FloatField(default=0.0)

（3）feedback

class Feedback(models.Model):

id = models.AutoField(primary\_key=True)

user = models.ForeignKey('people.User', on\_delete=models.CASCADE)

aftersaleman = models.ForeignKey('people.Admin', on\_delete=models.CASCADE, related\_name="aftersaleman", blank=True,

null=True)

assetman = models.ForeignKey('people.Admin', on\_delete=models.CASCADE, related\_name="assetman", blank=True,

null=True)

# 0:未处理,1:售后反馈资金,2:资金反馈售后,3:售后反馈用户

que\_condition = models.IntegerField(default=0)

# A给B的问题内容

que\_content\_user2aftersaleman = models.CharField(max\_length=300, blank=True)

que\_content\_aftersaleman2user = models.CharField(max\_length=300, blank=True)

que\_content\_assetman2aftersaleman = models.CharField(max\_length=300, blank=True)

que\_content\_aftersaleman2assetman = models.CharField(max\_length=300, blank=True)

（4）people

class User(AbstractUser):

balance = models.FloatField(default=0)

birth = models.DateField(null=True, blank=True)

phone = models.CharField(max\_length=11, blank=True)

gender = models.CharField(max\_length=2, blank=True)

class UserAddress(models.Model):

user = models.ForeignKey('User', on\_delete=models.CASCADE)

receiver\_name = models.CharField(max\_length=20, blank=True)

receiver\_phone = models.CharField(max\_length=20, blank=True)

receiver\_region = models.CharField(max\_length=20, blank=True)

receiver\_place = models.CharField(max\_length=20, blank=True)

class Admin(models.Model):

id = models.AutoField(primary\_key=True)

username = models.CharField(max\_length=20)

name = models.CharField(max\_length=20)

pwd = models.CharField(max\_length=300)

level = models.IntegerField(default=0)

gender = models.CharField(max\_length=2, blank=True)

birth = models.CharField(max\_length=10, blank=True)

intro = models.CharField(max\_length=1000, blank=True)

phone = models.CharField(max\_length=11, blank=True)

（5）sell

class WareHouse(models.Model):

user = models.ForeignKey('people.User', related\_name='+', on\_delete=models.CASCADE)

book = models.ForeignKey('book.Book', on\_delete=models.CASCADE, db\_column='book\_isbn')

id = models.AutoField(primary\_key=True)

customer\_condition = models.ForeignKey('book.ConditionRate', on\_delete=models.CASCADE, related\_name="+")

class Order(models.Model):

user = models.ForeignKey('people.User', related\_name='+', on\_delete=models.CASCADE)

useraddress = models.ForeignKey('people.UserAddress', related\_name="sell", on\_delete=models.CASCADE)

id = models.AutoField(primary\_key=True)

status = models.IntegerField(default=0)

submit\_time = models.DateTimeField(auto\_now\_add=True)

finish\_time = models.DateTimeField(null=True)

class OrderDetail(models.Model):

order = models.ForeignKey('Order', on\_delete=models.CASCADE)

book = models.ForeignKey('book.Book', on\_delete=models.CASCADE, db\_column='book\_isbn')

id = models.AutoField(primary\_key=True)

customer\_condition = models.ForeignKey('book.ConditionRate', on\_delete=models.CASCADE, related\_name="+")

final\_condition = models.ForeignKey('book.ConditionRate', on\_delete=models.CASCADE, related\_name="+", null=True)

predict\_price = models.FloatField(null=True)

final\_price = models.FloatField(null=True)