

# BookStore 第二次大作业实验报告

Github地址: <https://github.com/Lhtie/book-store.git>

日期: 2023年6月11日

## 实验要求

在上次使用MongoDB实现的BookStore代码框架上, 重新设计关系数据库, 利用ORM使用postgresql实现原来的代码逻辑。

其中需要合理使用索引、事务处理等。

## 实验过程

- 安装PostgreSQL, 配置用户密码, 添加数据库  
运行Postgres服务

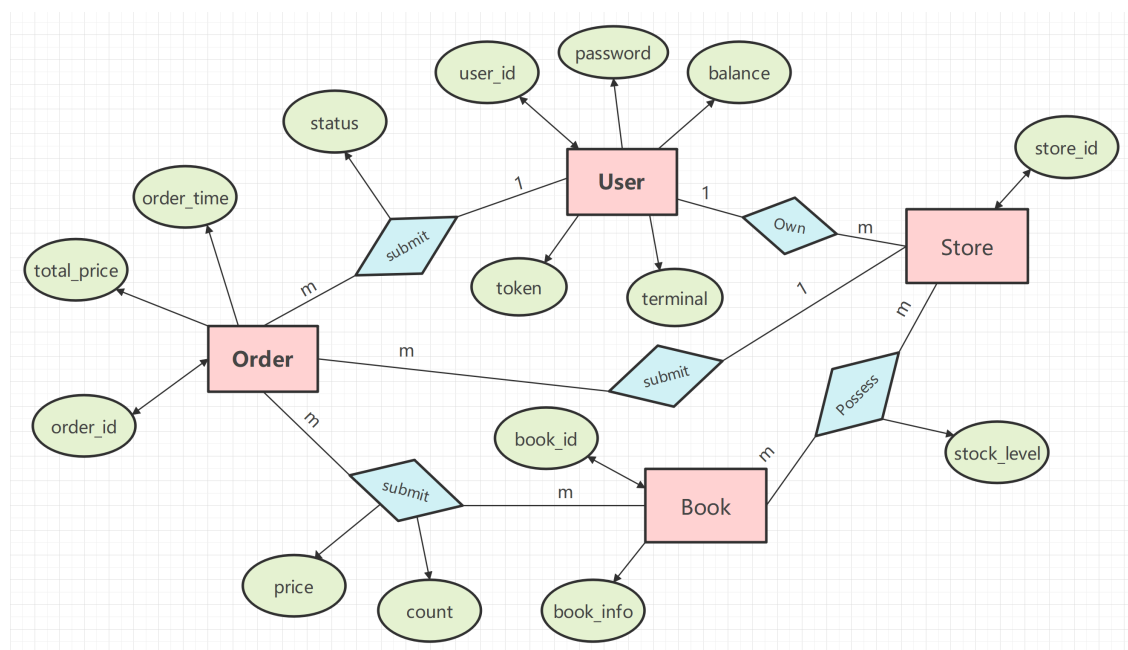
```
sudo service postgresql start
```

- 重新设计数据库关系模式, 使用ORM实现关系表的创建
- 修改所有原来调用MongoDB的接口, 使用postgresql重新改写
- 沿用上次前60%提供的以及后40%另外开发的测试代码, 对重写的代码做测试
- 报告测试结果

## 具体内容

### 1. 关系数据库设计

ER关系图设计



关系表设计

- user和store之间的own关系是一对多, 将此关系记录到store表中 (即后面的user\_store表)
- store和book之间possess是多对多的关系, 额外一张表记录此关系 (即后面的store表)

- user、store与order之间是一对多的关系，所以user和store的信息都记在了order表中（即后面的new\_order表）
- order和book之间是多对多的关系，此关系记录在额外的表中（即后面的new\_order\_detail表）

关系表具体构建

**Table user:**

user_id	password	balance	token	terminal
Text	Text	Integer	Text	Text

Create Index on user\_id  
Primary key on user\_id

**Table user\_store:**

user_id	store_id
Text	Text

Create Index on (user\_id, store\_id)  
Primary key on (user\_id, store\_id)

**Table store:**

store_id	book_id	book_info	stock_level
Text	Text	Text	Integer

Create Index on (store\_id, book\_id)  
Primary key on (store\_id, book\_id)

**Table new\_order:**

order_id	user_id	store_id	status	total_price	order_time
Text	Text	Text	Integer	Integer	Integer

status表示订单的状态

- 1表示已下单但未付款
- 2表示已付款但未发货
- 3表示已付款未发货
- 4表示已收货
- 0表示订单已取消

Create Index on order\_id  
Primary key on order\_id

**Table new\_order\_detail**

id	order_id	book_id	count	price
uuid	Text	Text	Integer	Integer

Create Index on id  
Primary key on id

**Table history\_order**

order_id	user_id	store_id	total_price	status	order_time
Text	Text	Text	Integer	Integer	Integer

Create Index on order\_id  
Primary key on order\_id

**Table history\_order\_books**

id	order_id	book_id	count	price
uuid	Text	Text	Integer	Integer

Create Index on id  
Primary key on id

**Table inverted\_index**

id	key_ctx	store_id	book_id
uuid	Text	Text	Text

Create Index on id  
Primary key on id

## 使用ORM代码构建

```
from sqlalchemy.ext.declarative import declarative_base

Base = declarative_base()
class TABLE_NAME(Base):
    __tablename__ = 'TABLE_NAME'
    entity = Column(...)
    # primary=True, index=True if needed
```

```
# drop all tables
Base.metadata.drop_all(self.engine)
# create tables
Base.metadata.create_all(self.engine)
```

## 2. 事务处理

### PostgreSQL 代码框架

```
# Begin Transaction
session = self.db.DbSession()

# Actions
session.add(...)
session.query(...).filter(...).update()
session.query(...).filter(...).delete()

# Commit & End
session.commit
session.close()
```

### 细节逻辑

- 尽量将对一条数据的查询、访问、修改、删除放在一个事务处理内，并且在操作最后commit
- 如果是插入多条类似数据，则收集所有数据，在同一个事务处理内逐个插入
- 注意 (possible bugs)：如果一个ORM实例被插入进SQL，那么之后访问该实例都需要包括在事务处理内

## 实验结果

- 通过所有前60%数据，以及后40%自行开发的测试数据
- 测试覆盖率92%，几乎覆盖所有功能逻辑
- Bash Outputs for Test Scripts

```
===== test session starts
=====
platform linux -- Python 3.9.16, pytest-7.3.1, pluggy-1.0.0 --
/home/lhtie/anaconda3/envs/book_store/bin/python
cachedir: .pytest_cache
rootdir: /mnt/c/Users/lhtie/Documents/ACM Class/大三下/数据库系统/第二次大作业/book-store/bookstore
collecting ... frontend begin test
2023-06-11 10:42:35,682 INFO sqlalchemy.engine.Engine select
pg_catalog.version()
2023-06-11 10:42:35,684 INFO sqlalchemy.engine.Engine [raw sql] {}
2023-06-11 10:42:35,692 INFO sqlalchemy.engine.Engine select
current_schema()
2023-06-11 10:42:35,694 INFO sqlalchemy.engine.Engine [raw sql] {}
2023-06-11 10:42:35,698 INFO sqlalchemy.engine.Engine show
standard_conforming_strings
2023-06-11 10:42:35,699 INFO sqlalchemy.engine.Engine [raw sql] {}
2023-06-11 10:42:35,703 INFO sqlalchemy.engine.Engine BEGIN (implicit)
2023-06-11 10:42:35,718 INFO sqlalchemy.engine.Engine SELECT
pg_catalog.pg_class.relname
FROM pg_catalog.pg_class JOIN pg_catalog.pg_namespace ON
pg_catalog.pg_namespace.oid = pg_catalog.pg_class.relnamespace
```

```

WHERE pg_catalog.pg_class.relname = %(table_name)s AND
pg_catalog.pg_class.relkind = ANY (ARRAY[%(param_1)s, %(param_2)s, %
(param_3)s, %(param_4)s, %(param_5)s]) AND
pg_catalog.pg_table_is_visible(pg_catalog.pg_class.oid) AND
pg_catalog.pg_namespace.nspname != %(nspname_1)s
2023-06-11 10:42:35,719 INFO sqlalchemy.engine.Engine [generated in
0.00157s] {'table_name': 'user', 'param_1': 'r', 'param_2': 'p', 'param_3':
'f', 'param_4': 'v', 'param_5': 'm', 'nspname_1': 'pg_catalog'}
2023-06-11 10:42:35,725 INFO sqlalchemy.engine.Engine SELECT
pg_catalog.pg_class.relname
FROM pg_catalog.pg_class JOIN pg_catalog.pg_namespace ON
pg_catalog.pg_namespace.oid = pg_catalog.pg_class.relnamespace
WHERE pg_catalog.pg_class.relname = %(table_name)s AND
pg_catalog.pg_class.relkind = ANY (ARRAY[%(param_1)s, %(param_2)s, %
(param_3)s, %(param_4)s, %(param_5)s]) AND
pg_catalog.pg_table_is_visible(pg_catalog.pg_class.oid) AND
pg_catalog.pg_namespace.nspname != %(nspname_1)s
2023-06-11 10:42:35,726 INFO sqlalchemy.engine.Engine [cached since
0.008769s ago] {'table_name': 'user_store', 'param_1': 'r', 'param_2': 'p',
'param_3': 'f', 'param_4': 'v', 'param_5': 'm', 'nspname_1': 'pg_catalog'}
2023-06-11 10:42:35,728 INFO sqlalchemy.engine.Engine SELECT
pg_catalog.pg_class.relname
FROM pg_catalog.pg_class JOIN pg_catalog.pg_namespace ON
pg_catalog.pg_namespace.oid = pg_catalog.pg_class.relnamespace
WHERE pg_catalog.pg_class.relname = %(table_name)s AND
pg_catalog.pg_class.relkind = ANY (ARRAY[%(param_1)s, %(param_2)s, %
(param_3)s, %(param_4)s, %(param_5)s]) AND
pg_catalog.pg_table_is_visible(pg_catalog.pg_class.oid) AND
pg_catalog.pg_namespace.nspname != %(nspname_1)s
2023-06-11 10:42:35,729 INFO sqlalchemy.engine.Engine [cached since 0.01165s
ago] {'table_name': 'store', 'param_1': 'r', 'param_2': 'p', 'param_3': 'f',
'param_4': 'v', 'param_5': 'm', 'nspname_1': 'pg_catalog'}
2023-06-11 10:42:35,731 INFO sqlalchemy.engine.Engine SELECT
pg_catalog.pg_class.relname
FROM pg_catalog.pg_class JOIN pg_catalog.pg_namespace ON
pg_catalog.pg_namespace.oid = pg_catalog.pg_class.relnamespace
WHERE pg_catalog.pg_class.relname = %(table_name)s AND
pg_catalog.pg_class.relkind = ANY (ARRAY[%(param_1)s, %(param_2)s, %
(param_3)s, %(param_4)s, %(param_5)s]) AND
pg_catalog.pg_table_is_visible(pg_catalog.pg_class.oid) AND
pg_catalog.pg_namespace.nspname != %(nspname_1)s
2023-06-11 10:42:35,732 INFO sqlalchemy.engine.Engine [cached since 0.01516s
ago] {'table_name': 'new_order', 'param_1': 'r', 'param_2': 'p', 'param_3':
'f', 'param_4': 'v', 'param_5': 'm', 'nspname_1': 'pg_catalog'}
2023-06-11 10:42:35,734 INFO sqlalchemy.engine.Engine SELECT
pg_catalog.pg_class.relname
FROM pg_catalog.pg_class JOIN pg_catalog.pg_namespace ON
pg_catalog.pg_namespace.oid = pg_catalog.pg_class.relnamespace
WHERE pg_catalog.pg_class.relname = %(table_name)s AND
pg_catalog.pg_class.relkind = ANY (ARRAY[%(param_1)s, %(param_2)s, %
(param_3)s, %(param_4)s, %(param_5)s]) AND
pg_catalog.pg_table_is_visible(pg_catalog.pg_class.oid) AND
pg_catalog.pg_namespace.nspname != %(nspname_1)s

```

```

2023-06-11 10:42:35,736 INFO sqlalchemy.engine.Engine [cached since 0.01856s
ago] {'table_name': 'new_order_detail', 'param_1': 'r', 'param_2': 'p',
'param_3': 'f', 'param_4': 'v', 'param_5': 'm', 'nspname_1': 'pg_catalog'}
2023-06-11 10:42:35,738 INFO sqlalchemy.engine.Engine SELECT
pg_catalog.pg_class.relname
FROM pg_catalog.pg_class JOIN pg_catalog.pg_namespace ON
pg_catalog.pg_namespace.oid = pg_catalog.pg_class.relnamespace
WHERE pg_catalog.pg_class.relname = %(table_name)s AND
pg_catalog.pg_class.relkind = ANY (ARRAY[%(param_1)s, %(param_2)s, %
(param_3)s, %(param_4)s, %(param_5)s]) AND
pg_catalog.pg_table_is_visible(pg_catalog.pg_class.oid) AND
pg_catalog.pg_namespace.nspname != %(nspname_1)s
2023-06-11 10:42:35,739 INFO sqlalchemy.engine.Engine [cached since 0.02222s
ago] {'table_name': 'history_order', 'param_1': 'r', 'param_2': 'p',
'param_3': 'f', 'param_4': 'v', 'param_5': 'm', 'nspname_1': 'pg_catalog'}
2023-06-11 10:42:35,742 INFO sqlalchemy.engine.Engine SELECT
pg_catalog.pg_class.relname
FROM pg_catalog.pg_class JOIN pg_catalog.pg_namespace ON
pg_catalog.pg_namespace.oid = pg_catalog.pg_class.relnamespace
WHERE pg_catalog.pg_class.relname = %(table_name)s AND
pg_catalog.pg_class.relkind = ANY (ARRAY[%(param_1)s, %(param_2)s, %
(param_3)s, %(param_4)s, %(param_5)s]) AND
pg_catalog.pg_table_is_visible(pg_catalog.pg_class.oid) AND
pg_catalog.pg_namespace.nspname != %(nspname_1)s
2023-06-11 10:42:35,743 INFO sqlalchemy.engine.Engine [cached since 0.02622s
ago] {'table_name': 'history_order_books', 'param_1': 'r', 'param_2': 'p',
'param_3': 'f', 'param_4': 'v', 'param_5': 'm', 'nspname_1': 'pg_catalog'}
2023-06-11 10:42:35,745 INFO sqlalchemy.engine.Engine SELECT
pg_catalog.pg_class.relname
FROM pg_catalog.pg_class JOIN pg_catalog.pg_namespace ON
pg_catalog.pg_namespace.oid = pg_catalog.pg_class.relnamespace
WHERE pg_catalog.pg_class.relname = %(table_name)s AND
pg_catalog.pg_class.relkind = ANY (ARRAY[%(param_1)s, %(param_2)s, %
(param_3)s, %(param_4)s, %(param_5)s]) AND
pg_catalog.pg_table_is_visible(pg_catalog.pg_class.oid) AND
pg_catalog.pg_namespace.nspname != %(nspname_1)s
2023-06-11 10:42:35,746 INFO sqlalchemy.engine.Engine [cached since 0.02931s
ago] {'table_name': 'inverted_index', 'param_1': 'r', 'param_2': 'p',
'param_3': 'f', 'param_4': 'v', 'param_5': 'm', 'nspname_1': 'pg_catalog'}
2023-06-11 10:42:35,750 INFO sqlalchemy.engine.Engine
DROP TABLE inverted_index
2023-06-11 10:42:35,751 INFO sqlalchemy.engine.Engine [no key 0.00102s] {}
2023-06-11 10:42:35,765 INFO sqlalchemy.engine.Engine
DROP TABLE history_order_books
2023-06-11 10:42:35,766 INFO sqlalchemy.engine.Engine [no key 0.00147s] {}
2023-06-11 10:42:35,770 INFO sqlalchemy.engine.Engine
DROP TABLE history_order
2023-06-11 10:42:35,772 INFO sqlalchemy.engine.Engine [no key 0.00146s] {}
2023-06-11 10:42:35,775 INFO sqlalchemy.engine.Engine
DROP TABLE new_order_detail
2023-06-11 10:42:35,777 INFO sqlalchemy.engine.Engine [no key 0.00136s] {}
2023-06-11 10:42:35,781 INFO sqlalchemy.engine.Engine
DROP TABLE new_order
2023-06-11 10:42:35,782 INFO sqlalchemy.engine.Engine [no key 0.00118s] {}
2023-06-11 10:42:35,785 INFO sqlalchemy.engine.Engine

```

```

DROP TABLE store
2023-06-11 10:42:35,786 INFO sqlalchemy.engine.Engine [no key 0.00121s] {}
2023-06-11 10:42:35,790 INFO sqlalchemy.engine.Engine
DROP TABLE user_store
2023-06-11 10:42:35,791 INFO sqlalchemy.engine.Engine [no key 0.00121s] {}
2023-06-11 10:42:35,795 INFO sqlalchemy.engine.Engine
DROP TABLE "user"
2023-06-11 10:42:35,796 INFO sqlalchemy.engine.Engine [no key 0.00122s] {}
2023-06-11 10:42:35,799 INFO sqlalchemy.engine.Engine COMMIT
2023-06-11 10:42:35,804 INFO sqlalchemy.engine.Engine BEGIN (implicit)
2023-06-11 10:42:35,806 INFO sqlalchemy.engine.Engine SELECT
pg_catalog.pg_class.relname
FROM pg_catalog.pg_class JOIN pg_catalog.pg_namespace ON
pg_catalog.pg_namespace.oid = pg_catalog.pg_class.relnamespace
WHERE pg_catalog.pg_class.relname = %(table_name)s AND
pg_catalog.pg_class.relkind = ANY (ARRAY[%(param_1)s, %(param_2)s, %
(param_3)s, %(param_4)s, %(param_5)s]) AND
pg_catalog.pg_table_is_visible(pg_catalog.pg_class.oid) AND
pg_catalog.pg_namespace.nspname != %(nspname_1)s
2023-06-11 10:42:35,806 INFO sqlalchemy.engine.Engine [cached since 0.08945s
ago] {'table_name': 'user', 'param_1': 'r', 'param_2': 'p', 'param_3': 'f',
'param_4': 'v', 'param_5': 'm', 'nspname_1': 'pg_catalog'}
2023-06-11 10:42:35,809 INFO sqlalchemy.engine.Engine SELECT
pg_catalog.pg_class.relname
FROM pg_catalog.pg_class JOIN pg_catalog.pg_namespace ON
pg_catalog.pg_namespace.oid = pg_catalog.pg_class.relnamespace
WHERE pg_catalog.pg_class.relname = %(table_name)s AND
pg_catalog.pg_class.relkind = ANY (ARRAY[%(param_1)s, %(param_2)s, %
(param_3)s, %(param_4)s, %(param_5)s]) AND
pg_catalog.pg_table_is_visible(pg_catalog.pg_class.oid) AND
pg_catalog.pg_namespace.nspname != %(nspname_1)s
2023-06-11 10:42:35,811 INFO sqlalchemy.engine.Engine [cached since 0.09415s
ago] {'table_name': 'user_store', 'param_1': 'r', 'param_2': 'p', 'param_3':
'f', 'param_4': 'v', 'param_5': 'm', 'nspname_1': 'pg_catalog'}
2023-06-11 10:42:35,815 INFO sqlalchemy.engine.Engine SELECT
pg_catalog.pg_class.relname
FROM pg_catalog.pg_class JOIN pg_catalog.pg_namespace ON
pg_catalog.pg_namespace.oid = pg_catalog.pg_class.relnamespace
WHERE pg_catalog.pg_class.relname = %(table_name)s AND
pg_catalog.pg_class.relkind = ANY (ARRAY[%(param_1)s, %(param_2)s, %
(param_3)s, %(param_4)s, %(param_5)s]) AND
pg_catalog.pg_table_is_visible(pg_catalog.pg_class.oid) AND
pg_catalog.pg_namespace.nspname != %(nspname_1)s
2023-06-11 10:42:35,816 INFO sqlalchemy.engine.Engine [cached since 0.09888s
ago] {'table_name': 'store', 'param_1': 'r', 'param_2': 'p', 'param_3': 'f',
'param_4': 'v', 'param_5': 'm', 'nspname_1': 'pg_catalog'}
2023-06-11 10:42:35,821 INFO sqlalchemy.engine.Engine SELECT
pg_catalog.pg_class.relname
FROM pg_catalog.pg_class JOIN pg_catalog.pg_namespace ON
pg_catalog.pg_namespace.oid = pg_catalog.pg_class.relnamespace
WHERE pg_catalog.pg_class.relname = %(table_name)s AND
pg_catalog.pg_class.relkind = ANY (ARRAY[%(param_1)s, %(param_2)s, %
(param_3)s, %(param_4)s, %(param_5)s]) AND
pg_catalog.pg_table_is_visible(pg_catalog.pg_class.oid) AND
pg_catalog.pg_namespace.nspname != %(nspname_1)s

```

```

2023-06-11 10:42:35,823 INFO sqlalchemy.engine.Engine [cached since 0.1058s
ago] {'table_name': 'new_order', 'param_1': 'r', 'param_2': 'p', 'param_3':
'f', 'param_4': 'v', 'param_5': 'm', 'nspname_1': 'pg_catalog'}
2023-06-11 10:42:35,827 INFO sqlalchemy.engine.Engine SELECT
pg_catalog.pg_class.relname
FROM pg_catalog.pg_class JOIN pg_catalog.pg_namespace ON
pg_catalog.pg_namespace.oid = pg_catalog.pg_class.relnamespace
WHERE pg_catalog.pg_class.relname = %(table_name)s AND
pg_catalog.pg_class.relkind = ANY (ARRAY[%(param_1)s, %(param_2)s, %
(param_3)s, %(param_4)s, %(param_5)s]) AND
pg_catalog.pg_table_is_visible(pg_catalog.pg_class.oid) AND
pg_catalog.pg_namespace.nspname != %(nspname_1)s
2023-06-11 10:42:35,828 INFO sqlalchemy.engine.Engine [cached since 0.1112s
ago] {'table_name': 'new_order_detail', 'param_1': 'r', 'param_2': 'p',
'param_3': 'f', 'param_4': 'v', 'param_5': 'm', 'nspname_1': 'pg_catalog'}
2023-06-11 10:42:35,832 INFO sqlalchemy.engine.Engine SELECT
pg_catalog.pg_class.relname
FROM pg_catalog.pg_class JOIN pg_catalog.pg_namespace ON
pg_catalog.pg_namespace.oid = pg_catalog.pg_class.relnamespace
WHERE pg_catalog.pg_class.relname = %(table_name)s AND
pg_catalog.pg_class.relkind = ANY (ARRAY[%(param_1)s, %(param_2)s, %
(param_3)s, %(param_4)s, %(param_5)s]) AND
pg_catalog.pg_table_is_visible(pg_catalog.pg_class.oid) AND
pg_catalog.pg_namespace.nspname != %(nspname_1)s
2023-06-11 10:42:35,834 INFO sqlalchemy.engine.Engine [cached since 0.1172s
ago] {'table_name': 'history_order', 'param_1': 'r', 'param_2': 'p',
'param_3': 'f', 'param_4': 'v', 'param_5': 'm', 'nspname_1': 'pg_catalog'}
2023-06-11 10:42:35,837 INFO sqlalchemy.engine.Engine SELECT
pg_catalog.pg_class.relname
FROM pg_catalog.pg_class JOIN pg_catalog.pg_namespace ON
pg_catalog.pg_namespace.oid = pg_catalog.pg_class.relnamespace
WHERE pg_catalog.pg_class.relname = %(table_name)s AND
pg_catalog.pg_class.relkind = ANY (ARRAY[%(param_1)s, %(param_2)s, %
(param_3)s, %(param_4)s, %(param_5)s]) AND
pg_catalog.pg_table_is_visible(pg_catalog.pg_class.oid) AND
pg_catalog.pg_namespace.nspname != %(nspname_1)s
2023-06-11 10:42:35,839 INFO sqlalchemy.engine.Engine [cached since 0.1219s
ago] {'table_name': 'history_order_books', 'param_1': 'r', 'param_2': 'p',
'param_3': 'f', 'param_4': 'v', 'param_5': 'm', 'nspname_1': 'pg_catalog'}
2023-06-11 10:42:35,844 INFO sqlalchemy.engine.Engine SELECT
pg_catalog.pg_class.relname
FROM pg_catalog.pg_class JOIN pg_catalog.pg_namespace ON
pg_catalog.pg_namespace.oid = pg_catalog.pg_class.relnamespace
WHERE pg_catalog.pg_class.relname = %(table_name)s AND
pg_catalog.pg_class.relkind = ANY (ARRAY[%(param_1)s, %(param_2)s, %
(param_3)s, %(param_4)s, %(param_5)s]) AND
pg_catalog.pg_table_is_visible(pg_catalog.pg_class.oid) AND
pg_catalog.pg_namespace.nspname != %(nspname_1)s
2023-06-11 10:42:35,845 INFO sqlalchemy.engine.Engine [cached since 0.128s
ago] {'table_name': 'inverted_index', 'param_1': 'r', 'param_2': 'p',
'param_3': 'f', 'param_4': 'v', 'param_5': 'm', 'nspname_1': 'pg_catalog'}
2023-06-11 10:42:35,852 INFO sqlalchemy.engine.Engine
CREATE TABLE "user" (
    user_id TEXT NOT NULL,
    password TEXT NOT NULL,

```



```

        balance INTEGER NOT NULL,
        token TEXT NOT NULL,
        terminal TEXT NOT NULL,
        PRIMARY KEY (user_id)
    )

2023-06-11 10:42:35,854 INFO sqlalchemy.engine.Engine [no key 0.00179s] {}
2023-06-11 10:42:35,882 INFO sqlalchemy.engine.Engine CREATE UNIQUE INDEX
ix_user_user_id ON "user" (user_id)
2023-06-11 10:42:35,884 INFO sqlalchemy.engine.Engine [no key 0.00153s] {}
2023-06-11 10:42:35,895 INFO sqlalchemy.engine.Engine
CREATE TABLE user_store (
    user_id TEXT NOT NULL,
    store_id TEXT NOT NULL,
    PRIMARY KEY (user_id, store_id)
)

2023-06-11 10:42:35,896 INFO sqlalchemy.engine.Engine [no key 0.00143s] {}
2023-06-11 10:42:35,907 INFO sqlalchemy.engine.Engine CREATE INDEX
ix_user_store_store_id ON user_store (store_id)
2023-06-11 10:42:35,909 INFO sqlalchemy.engine.Engine [no key 0.00171s] {}
2023-06-11 10:42:35,917 INFO sqlalchemy.engine.Engine CREATE INDEX
ix_user_store_user_id ON user_store (user_id)
2023-06-11 10:42:35,919 INFO sqlalchemy.engine.Engine [no key 0.00156s] {}
2023-06-11 10:42:35,926 INFO sqlalchemy.engine.Engine
CREATE TABLE store (
    store_id TEXT NOT NULL,
    book_id TEXT NOT NULL,
    book_info TEXT NOT NULL,
    stock_level INTEGER NOT NULL,
    PRIMARY KEY (store_id, book_id)
)

2023-06-11 10:42:35,927 INFO sqlalchemy.engine.Engine [no key 0.00141s] {}
2023-06-11 10:42:35,937 INFO sqlalchemy.engine.Engine CREATE INDEX
ix_store_store_id ON store (store_id)
2023-06-11 10:42:35,938 INFO sqlalchemy.engine.Engine [no key 0.00199s] {}
2023-06-11 10:42:35,956 INFO sqlalchemy.engine.Engine CREATE INDEX
ix_store_book_id ON store (book_id)
2023-06-11 10:42:35,957 INFO sqlalchemy.engine.Engine [no key 0.00145s] {}
2023-06-11 10:42:35,967 INFO sqlalchemy.engine.Engine
CREATE TABLE new_order (
    order_id TEXT NOT NULL,
    user_id TEXT NOT NULL,
    store_id TEXT NOT NULL,
    status INTEGER NOT NULL,
    total_price INTEGER NOT NULL,
    order_time INTEGER NOT NULL,
    PRIMARY KEY (order_id)
)

```

```
2023-06-11 10:42:35,969 INFO sqlalchemy.engine.Engine [no key 0.00203s] {}
2023-06-11 10:42:35,979 INFO sqlalchemy.engine.Engine CREATE UNIQUE INDEX
ix_new_order_order_id ON new_order (order_id)
2023-06-11 10:42:35,980 INFO sqlalchemy.engine.Engine [no key 0.00147s] {}
2023-06-11 10:42:35,990 INFO sqlalchemy.engine.Engine
CREATE TABLE new_order_detail (
    id VARCHAR NOT NULL,
    order_id TEXT NOT NULL,
    book_id TEXT NOT NULL,
    count INTEGER NOT NULL,
    price INTEGER NOT NULL,
    PRIMARY KEY (id)
)
```

```
2023-06-11 10:42:35,991 INFO sqlalchemy.engine.Engine [no key 0.00144s] {}
2023-06-11 10:42:36,000 INFO sqlalchemy.engine.Engine CREATE INDEX
ix_new_order_detail_order_id ON new_order_detail (order_id)
2023-06-11 10:42:36,001 INFO sqlalchemy.engine.Engine [no key 0.00108s] {}
2023-06-11 10:42:36,007 INFO sqlalchemy.engine.Engine
CREATE TABLE history_order (
    order_id TEXT NOT NULL,
    user_id TEXT NOT NULL,
    store_id TEXT NOT NULL,
    total_price INTEGER NOT NULL,
    status INTEGER NOT NULL,
    order_time INTEGER NOT NULL,
    PRIMARY KEY (order_id)
)
```

```
2023-06-11 10:42:36,008 INFO sqlalchemy.engine.Engine [no key 0.00115s] {}
2023-06-11 10:42:36,016 INFO sqlalchemy.engine.Engine CREATE UNIQUE INDEX
ix_history_order_order_id ON history_order (order_id)
2023-06-11 10:42:36,018 INFO sqlalchemy.engine.Engine [no key 0.00230s] {}
2023-06-11 10:42:36,034 INFO sqlalchemy.engine.Engine
CREATE TABLE history_order_books (
    id VARCHAR NOT NULL,
    order_id TEXT NOT NULL,
    book_id TEXT NOT NULL,
    count TEXT NOT NULL,
    PRIMARY KEY (id)
)
```

```
2023-06-11 10:42:36,036 INFO sqlalchemy.engine.Engine [no key 0.00161s] {}
2023-06-11 10:42:36,046 INFO sqlalchemy.engine.Engine CREATE INDEX
ix_history_order_books_order_id ON history_order_books (order_id)
2023-06-11 10:42:36,047 INFO sqlalchemy.engine.Engine [no key 0.00148s] {}
2023-06-11 10:42:36,055 INFO sqlalchemy.engine.Engine
CREATE TABLE inverted_index (
    id VARCHAR NOT NULL,
    key_ctx TEXT NOT NULL,
    store_id TEXT NOT NULL,
    book_id TEXT NOT NULL,
```

```

PRIMARY KEY (id)
)

2023-06-11 10:42:36,056 INFO sqlalchemy.engine.Engine [no key 0.00130s] {}
2023-06-11 10:42:36,068 INFO sqlalchemy.engine.Engine CREATE INDEX
ix_inverted_index_key_ctx ON inverted_index (key_ctx)
2023-06-11 10:42:36,070 INFO sqlalchemy.engine.Engine [no key 0.00229s] {}
2023-06-11 10:42:36,078 INFO sqlalchemy.engine.Engine COMMIT
Settings: Auto Cancel Out Of Time Orders
* Serving Flask app 'be.serve' (lazy loading)
* Environment: production
  WARNING: This is a development server. Do not use it in a production
  deployment.
  Use a production WSGI server instead.
* Debug mode: off
collected 58 items

fe/test/test_add_book.py::TestAddBook::test_ok PASSED [
  1%]
fe/test/test_add_book.py::TestAddBook::test_error_non_exist_store_id PASSED
[  3%]
fe/test/test_add_book.py::TestAddBook::test_error_exist_book_id PASSED [
  5%]
fe/test/test_add_book.py::TestAddBook::test_error_non_exist_user_id PASSED [
  6%]
fe/test/test_add_funds.py::TestAddFunds::test_ok PASSED [
  8%]
fe/test/test_add_funds.py::TestAddFunds::test_error_user_id PASSED [
 10%]
fe/test/test_add_funds.py::TestAddFunds::test_error_password PASSED [
 12%]
fe/test/test_add_stock_level.py::TestAddStockLevel::test_error_user_id
PASSED [ 13%]
fe/test/test_add_stock_level.py::TestAddStockLevel::test_error_store_id
PASSED [ 15%]
fe/test/test_add_stock_level.py::TestAddStockLevel::test_error_book_id
PASSED [ 17%]
fe/test/test_add_stock_level.py::TestAddStockLevel::test_ok PASSED [
 18%]
fe/test/test_bench.py::test_bench PASSED [
 20%]
fe/test/test_create_store.py::TestCreateStore::test_ok PASSED [
 22%]
fe/test/test_create_store.py::TestCreateStore::test_error_exist_store_id
PASSED [ 24%]
fe/test/test_find_books.py::TestFindBooks::test_find_books_ok PASSED [
 25%]
fe/test/test_find_books.py::TestFindBooks::test_find_books_sep_ok PASSED [
 27%]
fe/test/test_find_books.py::TestFindBooks::test_find_books_empty PASSED [
 29%]

```

```
fe/test/test_find_books.py::TestFindBooks::test_find_books_out_of_pages
2023-06-11 11:01:11,616 INFO sqlalchemy.engine.Engine UPDATE store SET
stock_level=%(stock_level)s WHERE store.store_id = %(store_store_id)s AND
store.book_id = %(store_book_id)s
PASSED [ 31%]2023-06-11 11:01:11,728 INFO sqlalchemy.engine.Engine BEGIN
(implicit)
2023-06-11 11:01:11,731 INFO sqlalchemy.engine.Engine SELECT store.store_id
AS store_store_id, store.book_id AS store_book_id, store.book_info AS
store_book_info, store.stock_level AS store_stock_level
FROM store
WHERE store.store_id = %(store_id_1)s AND store.book_id = %(book_id_1)s
LIMIT %(param_1)s
2023-06-11 11:01:11,734 INFO sqlalchemy.engine.Engine [cached since 1055s
ago] {'store_id_1': 'store_s_1_1_c3abaf6a-0801-11ee-a64f-00155deb4660',
'book_id_1': '1009368', 'param_1': 1}
2023-06-11 11:01:11,744 INFO sqlalchemy.engine.Engine UPDATE store SET
stock_level=%(stock_level)s WHERE store.store_id = %(store_store_id)s AND
store.book_id = %(store_book_id)s

fe/test/test_find_books.py::TestFindBooks::test_find_in_store_ok PASSED [
32%]
fe/test/test_find_books.py::TestFindBooks::test_find_in_store_sep_ok PASSED
[ 34%]
fe/test/test_find_books.py::TestFindBooks::test_find_in_store_empty PASSED [
36%]
fe/test/test_find_books.py::TestFindBooks::test_find_in_store_out_of_pages
PASSED [ 37%]
fe/test/test_find_books.py::TestFindBooks::test_find_in_store_non_exist_stor
e_id PASSED [ 39%]
fe/test/test_login.py::TestLogin::test_ok PASSED [
41%]
fe/test/test_login.py::TestLogin::test_error_user_id PASSED [
43%]
fe/test/test_login.py::TestLogin::test_error_password PASSED [
44%]
fe/test/test_new_order.py::TestNewOrder::test_non_exist_book_id PASSED [
46%]
fe/test/test_new_order.py::TestNewOrder::test_low_stock_level PASSED [
48%]
fe/test/test_new_order.py::TestNewOrder::test_ok PASSED [
50%]
fe/test/test_new_order.py::TestNewOrder::test_non_exist_user_id PASSED [
51%]
fe/test/test_new_order.py::TestNewOrder::test_non_exist_store_id PASSED [
53%]
fe/test/test_orders.py::TestOrders::test_cancel_order_ok PASSED [
55%]
fe/test/test_orders.py::TestOrders::test_cancel_order_repeat PASSED [
56%]
fe/test/test_orders.py::TestOrders::test_cancel_order_after_payment PASSED [
58%]
fe/test/test_orders.py::TestOrders::test_cancel_order_after_send PASSED [
60%]
fe/test/test_orders.py::TestOrders::test_cancel_order_after_receive PASSED [
62%]
```

```

fe/test/test_orders.py::TestOrders::test_cancel_non_exist_buyer_id PASSED [
63%]
fe/test/test_orders.py::TestOrders::test_cancel_non_exist_order_id PASSED [
65%]
fe/test/test_orders.py::TestOrders::test_auto_cancel_ok PASSED [
67%]
fe/test/test_password.py::TestPassword::test_ok PASSED [
68%]
fe/test/test_password.py::TestPassword::test_error_password PASSED [
70%]
fe/test/test_password.py::TestPassword::test_error_user_id PASSED [
72%]
fe/test/test_payment.py::TestPayment::test_ok PASSED [
74%]
fe/test/test_payment.py::TestPayment::test_authorization_error PASSED [
75%]
fe/test/test_payment.py::TestPayment::test_not_suff_funds PASSED [
77%]
fe/test/test_payment.py::TestPayment::test_repeat_pay PASSED [
79%]
fe/test/test_register.py::TestRegister::test_register_ok PASSED [
81%]
fe/test/test_register.py::TestRegister::test_unregister_ok PASSED [
82%]
fe/test/test_register.py::TestRegister::test_unregister_error_authorization
PASSED [ 84%]
fe/test/test_register.py::TestRegister::test_register_error_exist_user_id
PASSED [ 86%]
fe/test/test_send_receive.py::TestSendReceive::test_send_books_ok PASSED [
87%]
fe/test/test_send_receive.py::TestSendReceive::test_send_before_payment
PASSED [ 89%]
fe/test/test_send_receive.py::TestSendReceive::test_send_non_exist_store_id
PASSED [ 91%]
fe/test/test_send_receive.py::TestSendReceive::test_send_non_exist_order_id
PASSED [ 93%]
fe/test/test_send_receive.py::TestSendReceive::test_receive_books_ok PASSED
[ 94%]
fe/test/test_send_receive.py::TestSendReceive::test_receive_before_send
PASSED [ 96%]
fe/test/test_send_receive.py::TestSendReceive::test_receive_non_exist_buyer_
id PASSED [ 98%]
fe/test/test_send_receive.py::TestSendReceive::test_receive_non_exist_order_
id PASSED [100%]

```

```

===== warnings summary

```

```

=====

```

```

../../../../../../../../../../../../home/lhtie/anaconda3/envs/book_store/lib/py
thon3.9/site-packages/pkg_resources/__init__.py:121

```

```

/home/lhtie/anaconda3/envs/book_store/lib/python3.9/site-
packages/pkg_resources/__init__.py:121: DeprecationWarning: pkg_resources is
deprecated as an API

```

```

    warnings.warn("pkg_resources is deprecated as an API",
DeprecationWarning)

```

```

be/model/store.py:8
  /mnt/c/Users/lhtie/Documents/ACM Class/大三下/数据库系统/第二次大作业/book-
store/bookstore/be/model/store.py:8: MovedIn20Warning: The
  ``declarative_base()`` function is now available as
  sqlalchemy.orm.declarative_base(). (deprecated since: 2.0) (Background on
  SQLAlchemy 2.0 at: https://sqlalche.me/e/b8d9)
    Base = declarative_base()

```

-- Docs: <https://docs.pytest.org/en/stable/how-to/capture-warnings.html>

===== 58 passed, 2 warnings in 1723.49s (0:28:43)

=====

frontend end test

No data to combine

Name	Stmts	Miss	Branch	BrPart	Cover
-----					
be/__init__.py	0	0	0	0	100%
be/app.py	3	3	2	0	0%
be/model/__init__.py	0	0	0	0	100%
be/model/buyer.py	271	51	90	17	80%
be/model/db_conn.py	24	4	0	0	83%
be/model/error.py	25	1	0	0	96%
be/model/order.py	41	5	8	2	86%
be/model/seller.py	111	8	56	14	87%
be/model/stopwords.py	1	0	0	0	100%
be/model/store.py	85	2	8	0	98%
be/model/times.py	32	2	8	0	95%
be/model/tokenize.py	34	1	20	1	96%
be/model/user.py	124	15	32	6	87%
be/serve.py	41	1	2	1	95%
be/view/__init__.py	0	0	0	0	100%
be/view/auth.py	42	0	0	0	100%
be/view/buyer.py	78	0	2	0	100%
be/view/seller.py	38	0	0	0	100%
fe/__init__.py	0	0	0	0	100%
fe/access/__init__.py	0	0	0	0	100%
fe/access/auth.py	31	0	0	0	100%
fe/access/book.py	70	1	12	2	96%
fe/access/buyer.py	76	0	2	0	100%
fe/access/new_buyer.py	8	0	0	0	100%
fe/access/new_seller.py	8	0	0	0	100%
fe/access/seller.py	37	0	0	0	100%
fe/bench/__init__.py	0	0	0	0	100%
fe/bench/run.py	13	0	6	0	100%
fe/bench/session.py	47	0	12	1	98%
fe/bench/workload.py	125	1	22	2	98%
fe/conf.py	11	0	0	0	100%
fe/conftest.py	17	0	0	0	100%
fe/test/gen_book_data.py	50	0	16	0	100%
fe/test/test_add_book.py	36	0	10	0	100%
fe/test/test_add_funds.py	23	0	0	0	100%
fe/test/test_add_stock_level.py	39	0	10	0	100%
fe/test/test_bench.py	6	2	0	0	67%
fe/test/test_create_store.py	20	0	0	0	100%
fe/test/test_find_books.py	57	0	0	0	100%
fe/test/test_login.py	28	0	0	0	100%

fe/test/test_new_order.py	40	0	0	0	100%
fe/test/test_orders.py	135	45	4	1	67%
fe/test/test_password.py	33	0	0	0	100%
fe/test/test_payment.py	60	1	4	1	97%
fe/test/test_register.py	31	0	0	0	100%
fe/test/test_send_receive.py	89	1	4	1	98%
<hr/>					
TOTAL	2040	144	330	49	92%
wrote HTML report to htmlcov/index.html					

## Git代码管理

- 从第一次小组作业提交的Github Repo上fork个人版本
- 使用Git提交修改的代码