实验四 实验 电子商务网站的数据库设计

一、实验目的  
(1) 理解并掌握数据库设计过程  
(2) 熟悉数据库相关操作的命令  
二、实验内容

1、根据教材P95-P109，完成一个电子商务网站的数据库建设。

2、根据教材P109-P1113，商品购物流程数据库的设计与实现。

3、完成相关数据库以及表的创建。

三、相关操作

参看实验材料内容完成。

**完成一个电子商务网站的数据库建设**

**步骤1：**要设计电子商务网站的数据库设计，首先需要创建商品分类表。创建商品分类表的语句如下图1所示。

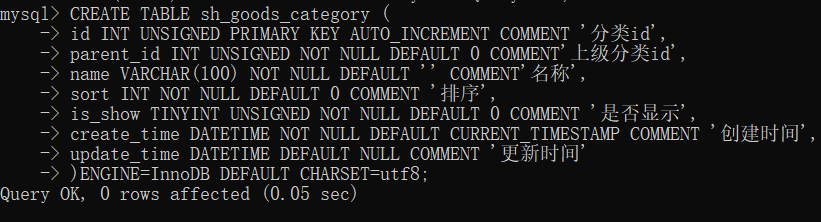


图1 创建商品分类表

**步骤2：**创建商品分类表后，紧接着要做的就是添加测试数据。添加测试数据的语句和结果如下图2所示。

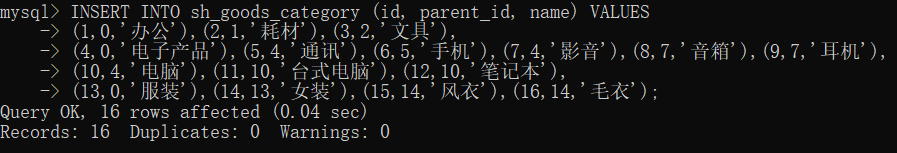


图2 添加测试数据

**步骤3：**添加数据之后，紧接着就是创建商品表。创建商品表的语句和结果如下图3所示。

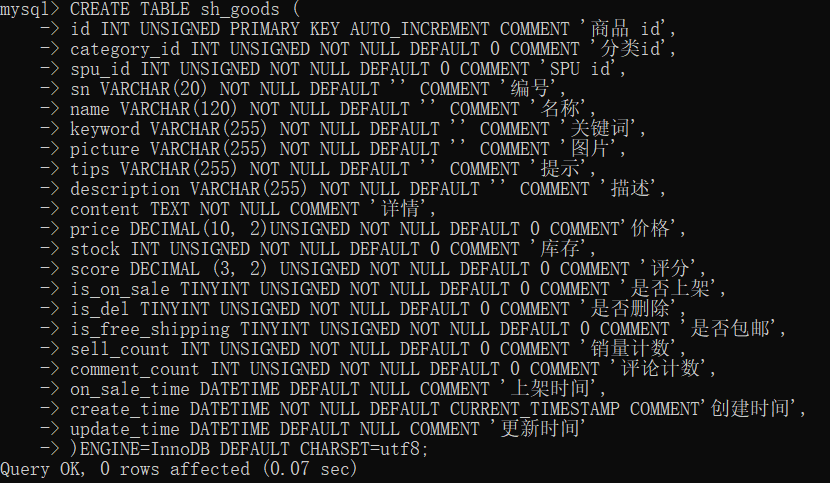


图3 创建商品表

**步骤4：**创建商品表之后，紧接着就是插入测试数据。插入测试数据的语句和结果如下图4所示。

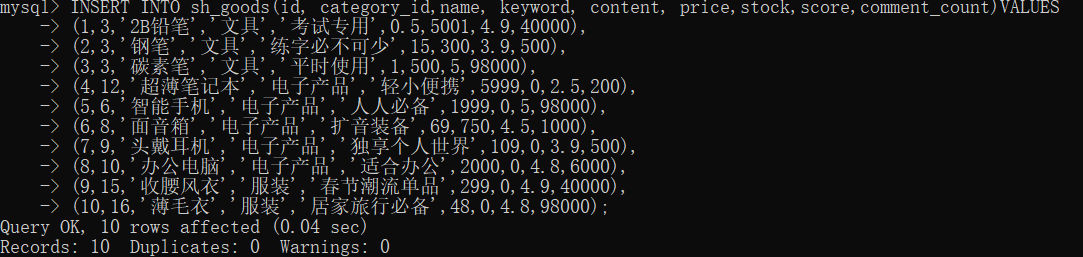


图4 插入测试数据

**步骤5：**插入测试数据之后，紧接着就是创建含有商品属性多维度的表。创建含有商品属性多维度的表的语句和结果如下图5-8所示。

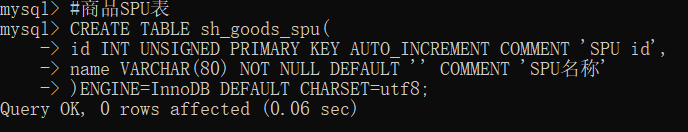


图5 创建商品SPU表

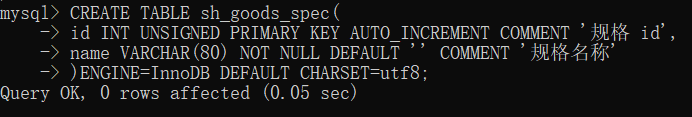


图6 创建商品规格表

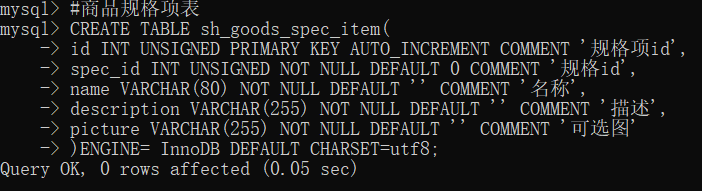


图7 创建商品规格表

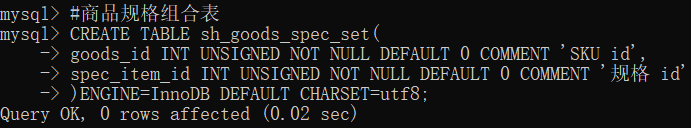


图8 创建商品规格组合表

**步骤6：**创建商品规格组合表之后，紧接着就是向上述四张表添加测试数据。添加测试数据的语句和结果如下图9所示。

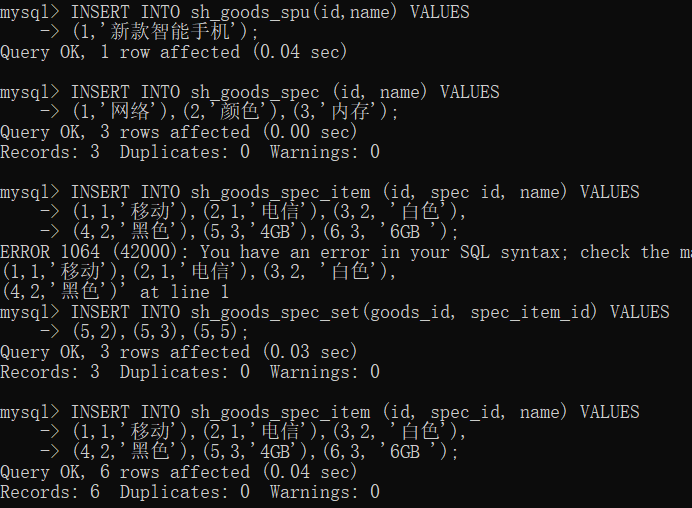


图9 向商品规格组合表添加测试数据

**步骤7：**向商品规格组合表添加测试数据之后，紧接着就是要考虑不同分类的商品应该具有不同的属性模板。创建含有商品不同属性模板的语句和结果如下图10-13所示。

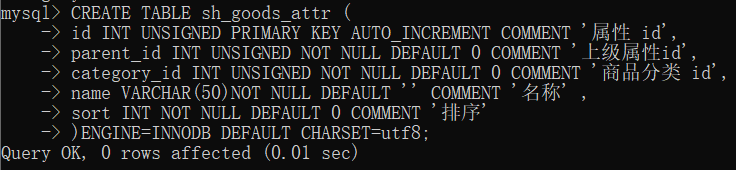


图10 创建商品属性表

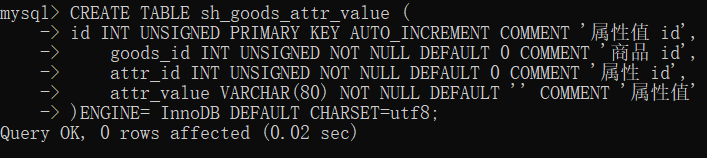


图11 创建商品属性值表

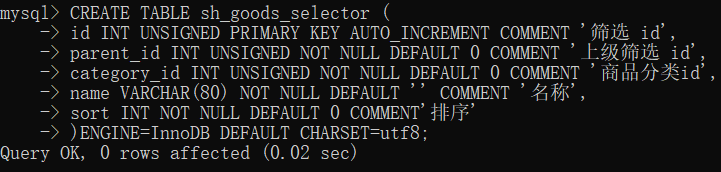


图12 创建商品筛选表

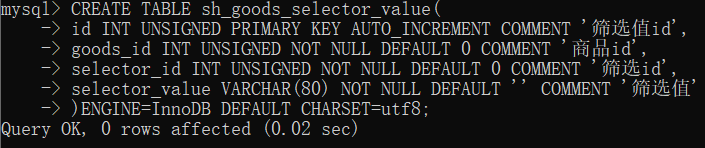


图13 创建商品筛选值表

**步骤8：**创建含有商品不同属性模板表之后，紧接着就是向上述四张表添加测试数据。添加测试数据的语句和结果如下图14所示。

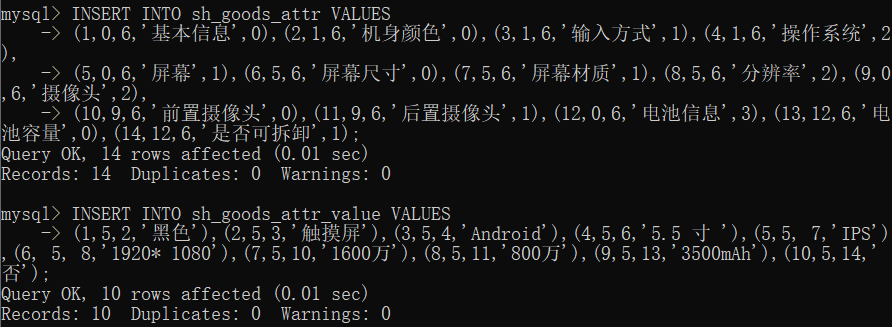


图14 向含有商品不同属性模板表添加测试数据

**步骤9：**向含有商品不同属性模板表添加测试数据之后，紧接着就是创建用户表。创建用户表的语句和结果如下图15所示。

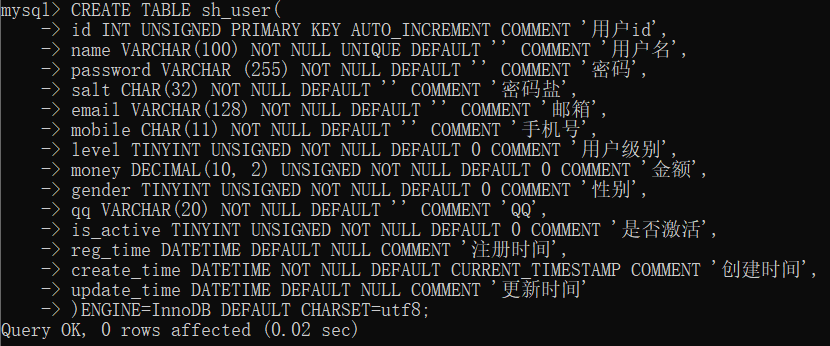


图15 创建用户表

**步骤10：**创建用户表之后，考虑到Web项目的安全性，应对用户的密码进行加密，不能明文存储，否则一旦泄露会造成很大的损失。用MD5加盐算法加密方式的语句和结果如下图16所示。

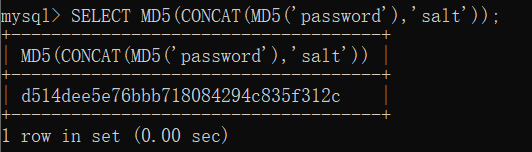


图16 利用MD5加盐算法加密

**步骤11：**用户表创建完并利用MD5加盐算法加密之后，紧接着就是为用户表添加测试数据，添加数据的语句和结果如下图17所示。

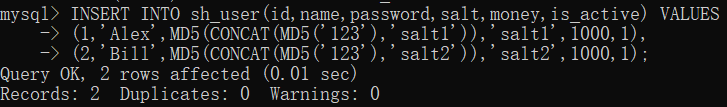


图17 为用户表添加测试数据

**步骤12：**可以通过SELECT命令来查看sh\_user表中的记录，观察密码加密的结果。其使用的方式如下图18所示。

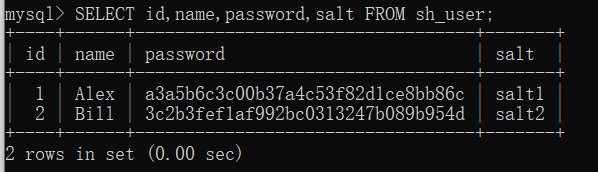


图18 查看密码

**步骤13：**创建用户表之后，还需要创建商品评论表。创建商品评论表的SQL语句和结果如下图19所示。

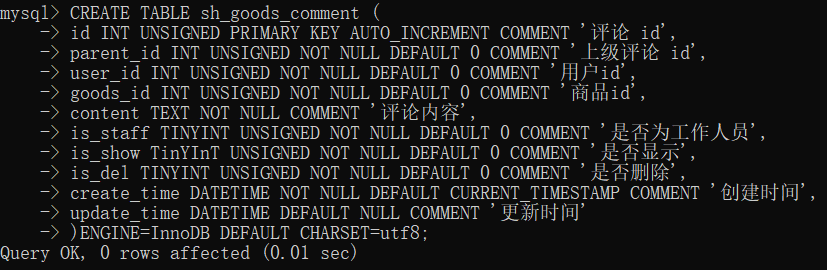


图19 创建商品评论表

**步骤14：**创建商品评论表之后，还需要为商品添加测试数据。添加测试数据的SQL语句和结果如下图20所示。

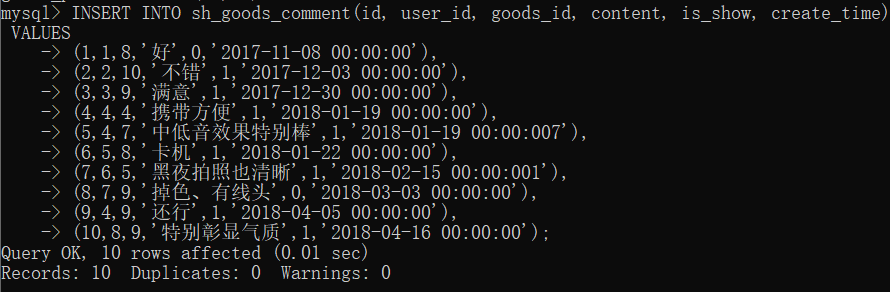


图20 添加测试数据

**商品购物流程数据库的设计与实现**

**步骤1：**在商品购物过程中，用户可以将想购买的商品添加到购物车，从而方便一次购买多件商品，此外，将商品添加到购物车后不会影响商品的库存。因此给出如下图21所示的SQL语句和结果。

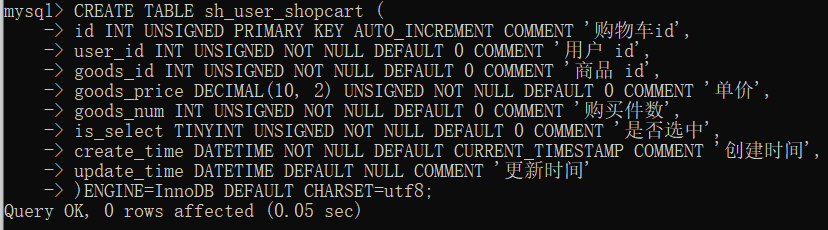
****

图21 创建购物车表

**步骤2：**创建购物车表之后，还需要创建存储收货地址的表。创建存储收货地址的表的SQL语句和结果如下图22所示。

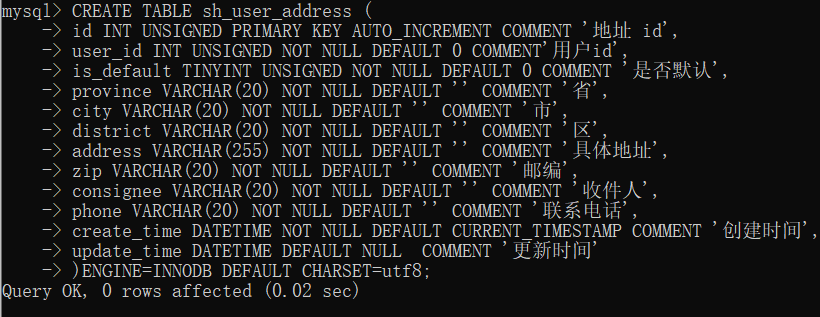
****

图22 创建收货地址表

**步骤3：**创建存储收货地址表之后，还需要创建存订单表和订单商品表。创建存订单表和订单商品表的SQL语句和结果如下图23、24所示。

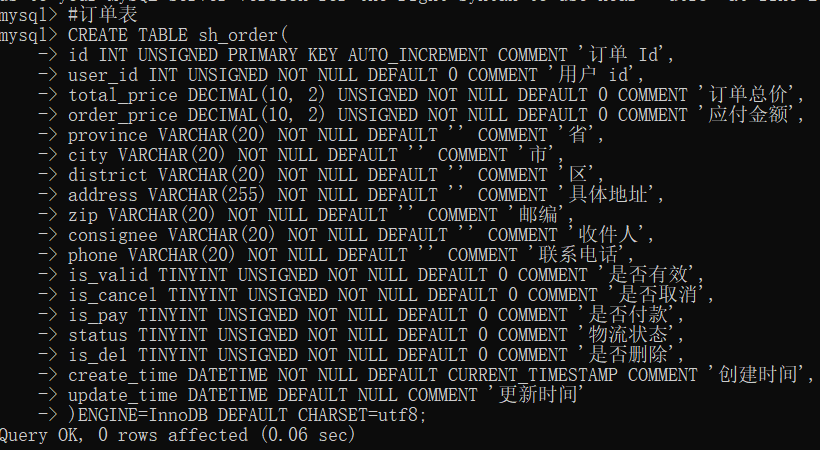
****

图23 创建订单表

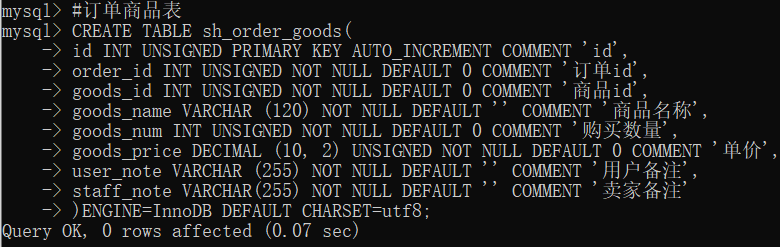
****

图24 创建订单商品表

**步骤4：**创建存订单表和订单商品表之后，还需要创建存商品评分表。创建存商品评分表的SQL语句和结果如下图25所示。

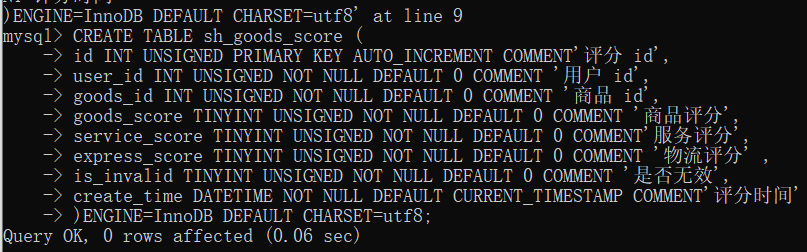
****

图25 创建存商品评分表

1. 问题思考与总结  
    1、数据库设计的过程有哪些步骤？

数据库设计一般要经过需求分析、概要设计、逻辑设计、代码编写、软件测试、安装部署六步。需求分析是数据库设计的第一步，是最困难、最耗费时间的一步，也是整个设计过程的基础。需求分析是否做的充分和准确，直接决定了在其上构建数据库大厦的速度与质量。需求分析做的不好，会导致整个数据库设计返工重做。概要设计概要设计是数据库设计的关键，通过综合、归纳与抽象用户需求，形成一个具体 DBMS 的概念模型，也就是绘制数据库的 E-R 图；逻辑结构设计将 E-R 图转换为多张表，进行逻辑设计，确认各表的主外键，并应用数据库设计的三大范式进行审核，对其优化；物理设计阶段经项目组开会讨论确定 E-R 图后，根据项目的技术实现，团队开发能力及项目的成本预算，选择具体的数据库（如 MySQL 或 Oracle 等）进行物理实现。数据库实施阶段运用 DBMS 提供的数据语言（例如 SQL）、工具及宿主语言，根据逻辑设计和物理设计的结果建立数据库，编制与调试应用程序，组织数据入库，并进行试运行。 数据库运行和维护阶段

1. 确定数据类型需要注意哪些问题？
2. INT数据类型可以用来保存那些不包含小数点的数字。INT代表整数。其中：

TINYINT——这个类型最多可容纳三位数 8bit

SMALLINT——最多可容纳五位数 16bit

MEDIUMINT——最多可容纳八位数 24bit

INT——可以容纳十位数 32bit

BIGINT——最多可容纳二十位数 64bit

1. 整数类型有可选的unsigned（无符号）属性，它表示不允许为负数，并大致把正上限提高了一倍，例如：tinyint unsigned保存的翻围为0到255，而不是-127到128。
2. 实数有分数部分，然而，它们并不仅仅是分数。可以使用decimal保存比出bigint还大的整数
3. 比较起decimal类型，浮点类型保存同样大小的值使用的空间通常更小，float类型占用4个字节，double占用8个字节，而且精度更大，范围更广。只有在需要对小数进行精确的时候才使用decimal
4. varchar保存了可变长度的字符串，是使用得最多的字符串类型，它能比固定类型占用更少的存储空间，因为它只占用了自已需要的空间（也就是说较短的值占用的空间更小）。它使用额外的1-2个字节来存储值的长度, 在存取值时候，MySQL不会去掉字符串末尾的空格。
5. char固定长度，char存取值时候，MySQL会去掉末尾的空格。Char在存储很短的字符串或长度近似相同的字符的时候很有用，MySQL使用\0（0字节）填充binary值，而不是空格，并且不会在获取数据的时候把填充的值截掉。
6. TEXT保存的是字符数据，有字符集和排序规则
7. DATETIME：能够保存大范围的值，从1001年到9999年，精度为秒，它把日期和时间封装到一个格式为yyyy:MM:dd:HH:mm:ss的整数当中，与时区无关。它使用了8个字节存储空间。
8. TIMESTAMP显示的值依赖于时区，MySQL服务器、操作系统及客户端连接都有时区设置。
9. 其他自己遇到的问题整理总计

由于本次实验是照着书上代码写的，因此除了粗瓷你打错字母之外，没有遇到棘手的问题。唯一遇到的非粗心问题见下述：

**问题1：**当我登陆成功后，不管运行任何命令，总是提示如下面图26所示的错误。

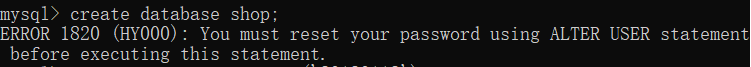


图26 报错1截图

遇到这个问题的原因是从MySQL版本5.6.6版本起，添加了password\_expired功能，它允许设置用户的过期时间。这个特性已经添加到mysql.user数据表，但是它的默认值是“N”，可以使用SET PASSWORD语句来修改这个值。，解决成功截图如下图27所示。

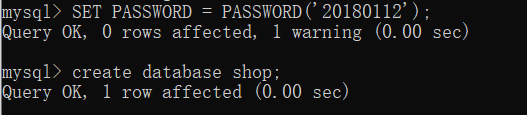


图27 报错1解决截图

1. 本章重要命令的总结。

创建数据表：CREATE TABLE+数据表名

插入数据：INSERT INTO + 数据表名 + 属性 + values

查看数据：SELECT +属性 + from +表名

非空约束：name VARCHAR(20) NOT NULL

主键约束：alter table UserId add constraint PK\_UserId primary key (UserId)

唯一约束：alter table UserInfo add constraint UQ\_IDNumber unique(IdentityCardNumber)

默认约束：alter table UserInfo add constraint DF\_UserAddress default (‘地址不详’) for UserAddress

检查约束：alter table UserInfo add constraint CK\_UserAge check (UserAge between 20 and 40)

外键约束：alter table UserOrder add constraint FK\_UserId\_UserId foreign key(UserId)references UserInfo(UserId)