**实验4数据库的安全性、完整性**

**1、实验目的要求**

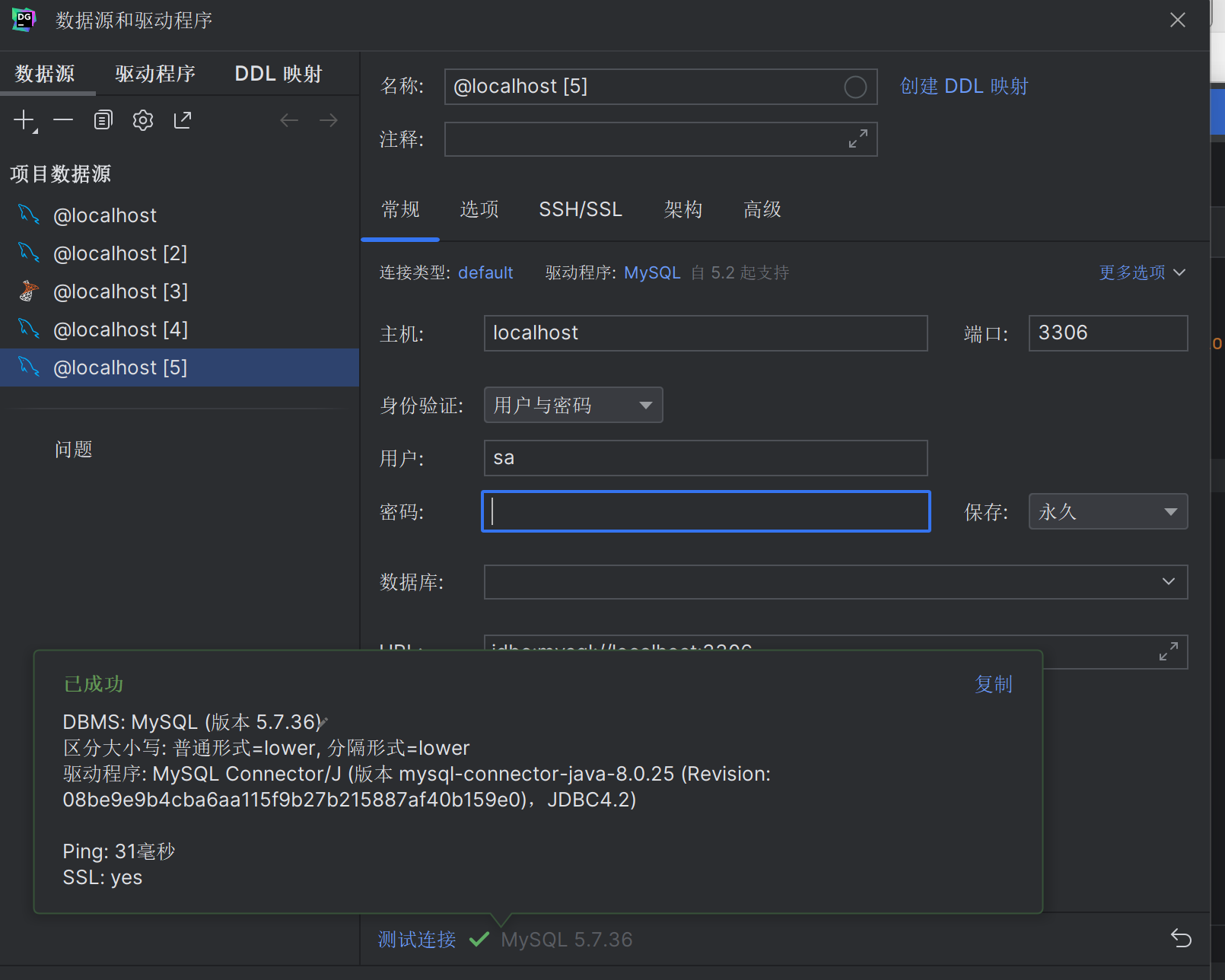
1. 掌握自主存取控制权限的定义和维护方法；
2. 掌握实体完整性的定义和维护方法；
3. 掌握参照完整性的定义和维护方法；
4. 掌握用户自定义完整性的定义和维护方法。

**2、实验主要内容**

1. 定义用户、角色，分配权限给用户、角色，回收权限，以相应的用户名登录数据库验证权限分配是否正确。选择一个应用场景，使用自主存取控制机制设计权限分配。可以采用两种方案。方案一：采用sa超级用户登录数据库，完成所有权限分配工作，然后用相应用户名登陆数据库以验证权限分配正确性；方案二：采用sa用户登陆数据库创建三个部门经理用户，并分配相应的权限，然后分别用三个经理用户名登陆数据库，创建相应部门的USER, ROLE，并分配相应权限。验证权限分配之前，请备份好数据库；针对不同用户所具有的权限，分别设计相应的SQL语句加以验证。

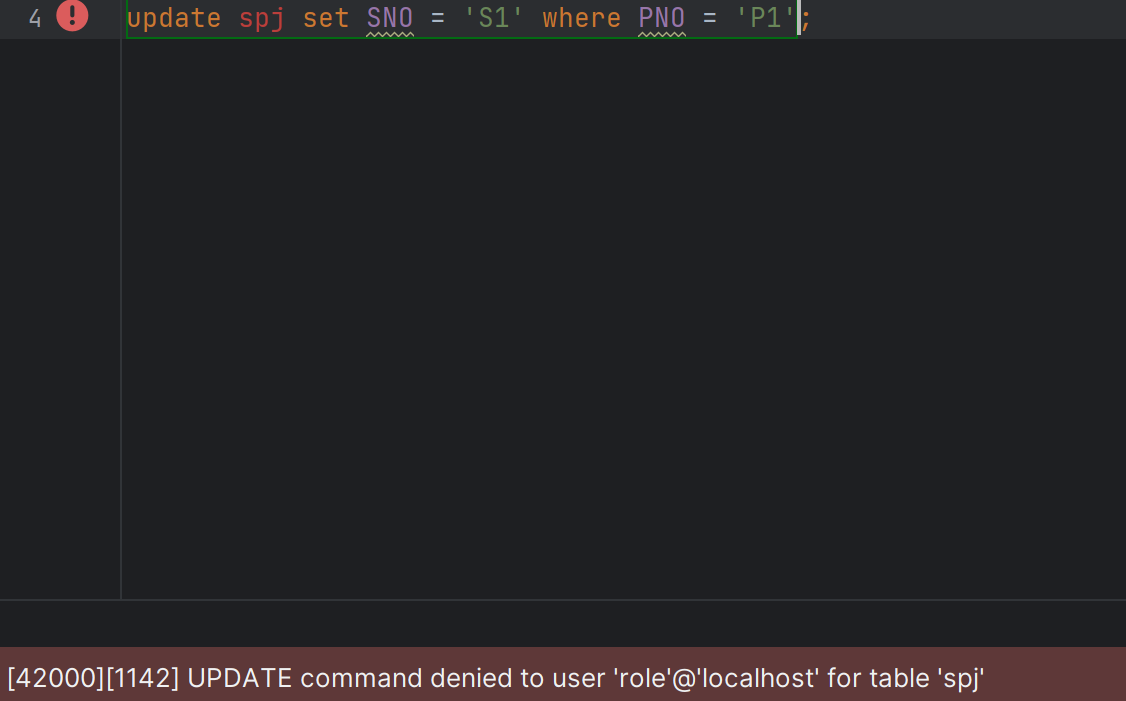
create user "sa" identified by 'qwe';  
  
grant all privileges on \*.\* to sa with grant option ;

使用创建的用户登录



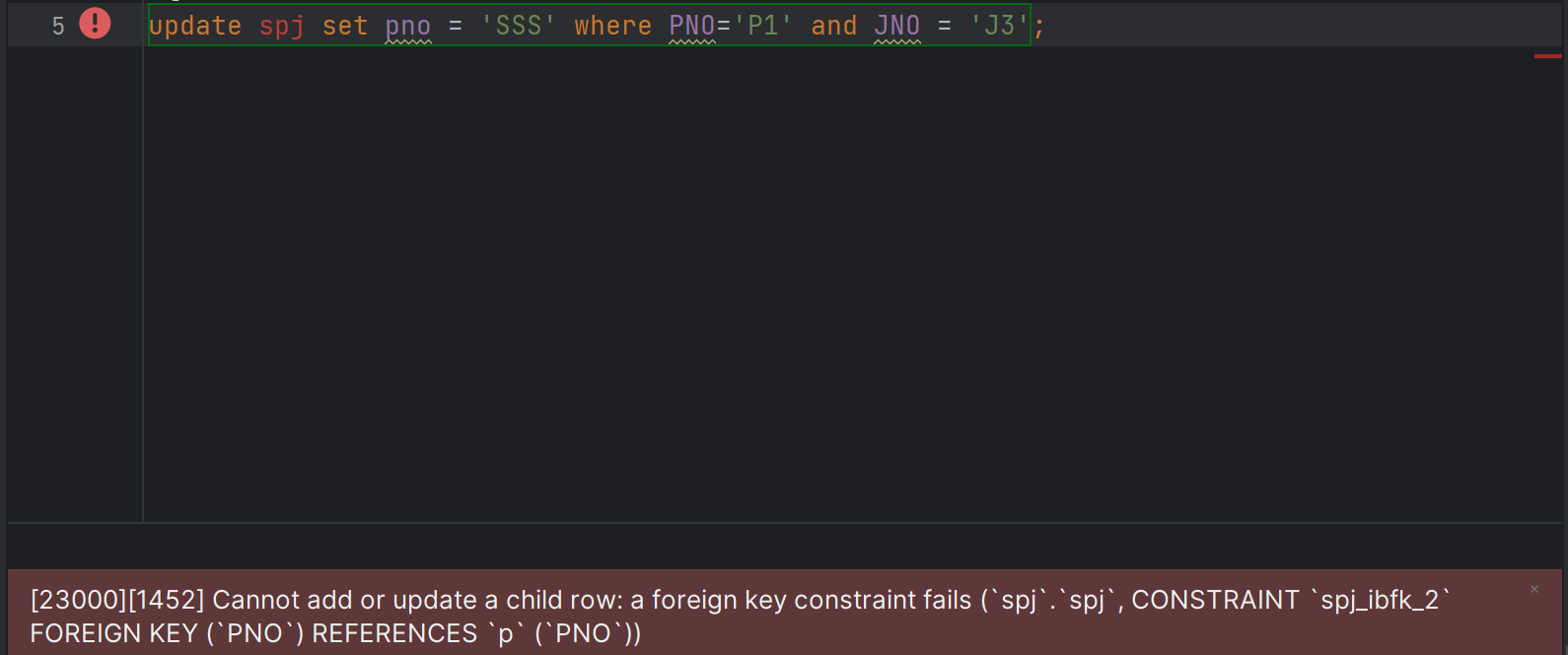
create user 'manager' identified by 'qweqwe';  
  
grant select,update on spj to 'manager';  
  
create user 'role' identified by 'qwe123';  
  
grant select on spj to role;

role 角色没有update 权限 无法update



此时使用manager用户

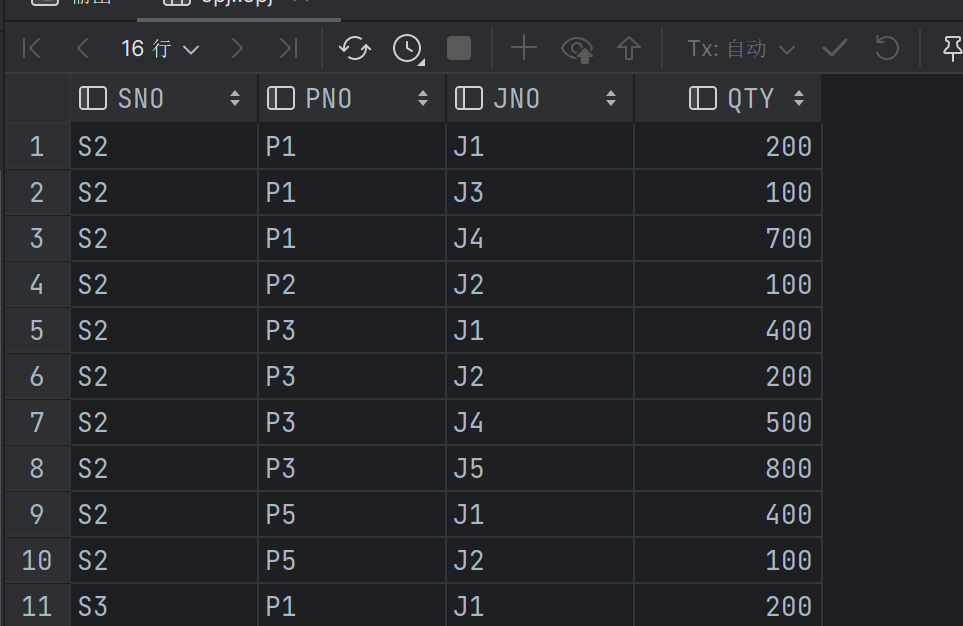
第一个没有授权权限的话 不能对外码中的数据进行修改



grant select,update on j to 'manager';

授权权限后

update spj set sno = 'S2' where SNO = 'S1';



此时S1 都更新为S2 此时权限可以正常进行

1. 定义参照完整性，定义参照完整性的违约处理，删除参照完整性。写出两种方式定义参照完整性的SQL语句：创建表时定义参照完整性、创建表后定义参照完整性。

创建表格时创建

create table mm(  
 MNo varchar(4),  
 MName varchar(10),  
 phone bigint,  
 address varchar(30),  
 constraint NAME\_RE primary key (MNo)  
);  
  
create table sss(  
 MNo varchar(4),  
 address varchar(20),  
 sssName varchar(20),  
 foreign key (MNo) references mm(MNo),  
 primary key (sssName)  
);

在创建表格后再添加参照完整性

create table sss\_test  
(  
 MNo varchar(4),  
 address varchar(20),  
 sssName varchar(20)  
);  
  
alter table sss\_test add constraint ss\_re foreign key (MNo) references mm(MNo);

1. 针对具体应用语义，选择NULL/NOT NULL、DEFAULT，UNIQUE、CHECK等，定义属性上的约束条件。

create table nnn(  
 id integer primary key not null ,  
 name char(10) unique ,  
 sal float default 0, // 工资没有时默认为0  
 birthyear int,  
 endyear int,  
 check ( endyear>nnn.birthyear ) // 此处定义结束年份大于诞生年份  
)

**3、实验仪器设备**

1. 学生每个一台PC机
2. 已安装SQL Server环境