|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 《数据库系统概论》实验报告 | | | | |
| 实验七 数据安全性实验 | 姓名 | 刘爱兵 | 日期 | 2020.11.19 |
| 实验内容与完成情况：   1. **实验内容：**   1、利用交互式和用T-SQL语句实现存储过程的创建、执行和删除；  2、利用交互式和用T-SQL语句实现触发器的创建、执行和删除。  本实验4个学时。  **二、实验任务**  1、交互式创建存储过程和触发器。  1）创建存储过程prostu,要求通过学号来查询学生姓名、性别和年龄。  CREATE PROCEDURE prostu  @sno\_in char(10)='200215121',@sname\_out char(10) output,@sage\_out int output,@ssex\_out char(10) output  AS  BEGIN  SET NOCOUNT ON;  SELECT @sname\_out=sname,@sage\_out=sage,@ssex\_out=Ssex  from Student where Sno=@sno\_in    END  GO    2)执行存储过程prostu  GO  declare @sno\_in char(10),@sname\_out char(10) ,@sage\_out int ,@sex\_out char(10)  select @sno\_in='200215122'  exec prostu @sno\_in,@sname\_out output, @sage\_out output,@sex\_out output  print @sname\_out  print @sage\_out  print @sex\_out    2、用T-SQL语句创建、执行和删除存储过程。  (1)使用学生选课数据库中的student表、course表;sc表创建一个带参数的存储过程—cjjicx。该存储过程的作用是：当任意输入一个学生的姓名时，将从三个表中返回该学生的学号、选修的课程名称和课程成绩。  create procedure [dbo].[cjjicx] @sname char(10)  as  select s.sno,c.cname,sc.grade  from student as s inner join sc as sc on s.sno=sc.sno inner join course as c on c.cno=sc.cno  where sname=@sname    (2)执行cjjicx存储过程，查询“XXX”（自己选个名字）的学号、选修课程和课程成绩。  执行语句：execute cjjicx 刘晨  结果描述：    (3)使用系统存储过程sp\_helptext查看存储过程cjjicx的文本信息。使用系统存储过程sp\_depends查看存储过程cjjicx中所涉及的对象信息。  执行语句：  execute sp\_helptext cjjicx  execute sp\_depends cjjicx  查看结果描述：      (4)使用学生选课数据库中的student表，为其创建一个存储过程—jmxs。该存储过程的作用是：当执行该存储过程时，将返回计算机系学生的所有信息。  Jmxs创建语句：create procedure jmxs  AS  select s.Sname,s.Sage,s.Sno,s.Ssex,s.Sage,c.Cname,c.Cpno,sc.Grade  from student as s inner join sc as sc on s.sno=sc.sno inner join course as c on c.cno=sc.cno  where Sdept = 'cs'    (5)执行jmxs存储过程，查看计算机系学生的情况。  执行语句：  execute jmxs  结果描述：    (6)删除jmxs存储过程。  删除语句：  drop PROC jmxs  go    (7)编写一个存储过程，给定一个课程名，查询该课程成绩的最高分，最低分和平均分。  create procedure gra  AS  select MAX(sc.Grade) as '最高分'  from student as s inner join sc as sc on s.sno=sc.sno inner join course as c on c.cno=sc.cno  where Cname = '信息系统'  select AVG(sc.Grade) as '平均分'  from student as s inner join sc as sc on s.sno=sc.sno inner join course as c on c.cno=sc.cno  where Cname = '信息系统'  select MIN(sc.Grade) as '最低分'  from student as s inner join sc as sc on s.sno=sc.sno inner join course as c on c.cno=sc.cno  where Cname = '信息系统'  execute gra      (8)创建查询表student的所有内容的存储过程pro\_student ，加密，并执行。  create procedure pro\_student  WITH ENCRYPTION  as  select s.Sname,s.Sage,s.Sno,s.Ssex  from student as s    execute pro\_student    (9) 创建存储过程pro\_student，查询指定姓名的学生学号、姓名、性别、课程名和成绩。  create procedure pro\_student  WITH ENCRYPTION  as  select s.Sname,s.Sno,s.Ssex,sc.Grade,c.Cname  from student as s inner join sc as sc on sc.Sno= s.Sno inner join course as c on c.Cno = sc.Cno  execute pro\_student    (10) 创建存储过程studentinfo，能够在客户端从数据库服务器检索出指定性别学生的信息（学号，姓名，课程名称，成绩），使用游标变量完成。  create proc studentinfo  @sex char(2),  @student\_cursor cursor varying output  as  set @student\_cursor = cursor  for  select s.sno,sname,cname,grade  from student as s inner join sc as sc on sc.Sno= s.Sno inner join course as c on c.Cno = sc.Cno  open @student\_cursor  declare @stcursor cursor  exec studentinfo 'f', @stcursor output  fetch next from @stcursor  while (@@fetch\_status=1)  begin  fetch next from @stcursor  end  deallocate @stcursor    3、交互式创建、执行、删除触发器。      4、用T-SQL语句创建、执行和删除触发器。  as  begin  declare @sno char(10)  select @sno=inserted.sno from inserted  if not exists(select sno from student where Sno=@sno)  begin  print '该学生的学号不存在'  rollback tran  end  end    insert into Student values('201715031','张三','男',121,'cs')    drop trigger insert\_1     1. 在student表上定义一个触发器，当插入或修改时，年龄如果低于18岁，自动改为18岁。   create trigger insert\_or\_update  on Student  for insert,update  as  update Student  set Sage = 18  From Student t , inserted i  where t.sno= i.sno and i.Sage < 18     1. 执行insert into student values ('1001','aa','男',17,'cs')，请查看student 表中插入记录的情况。      1. 在学生选课数据库中建立一个名为insert\_xh的INSERT触发器，存储在sc表中。该触发器的作用是：当用户向sc表中插入记录时，如果插入了在student表中没有的学生学号sno，则提示用户不能插入记录，否则提示记录插入成功。   create trigger insert\_xh  on sc  for insert  as  declare @sno nchar(10),@number int  select @sno= sno  from inserted  select @number=COUNT(\*)  from Student s  where s.sno=@sno  if(@number=0)  begin  print '不能插入'  rollback transaction  end     1. 为学生选课数据库中的student表创建一个名为dele\_stu的DELETE触发器，该触发器的作用是禁止删除student表中的记录。   create trigger dele\_stu  on student  for delete  as  begin  print '禁止删除'  rollback transaction  end     1. 为学生选课数据库中的sc表创建一个名为update\_grade的UPDATE触发器，该触发器的作用是禁止更新sc表中的grade字段的内容。   create trigger update\_grade  on sc  for update  as  begin  print '禁止更新'  rollback transaction  end     1. 禁用update\_grade触发器。   禁用update\_grade触发器的命令：  disable trigger update\_grade on sc    禁用的后果描述：   1. 删除update\_grade触发器。删除update\_grade触发器的命令：drop trigger update\_grade     (8)定义触发器，当Student学号发生修改时，自动在变化表S\_log登记修改记录。  提示:首先建立变化表S\_log ,然后再建UPDATE和INSERT两个触发器.  create table S\_log(ID int identity(1,1) sno char(10),  Date Datetime,primary key(ID));  create trigger Updata\_S  on student  for update  as  if UPDATE(Sno)  begin  insert into S\_log  select sno,CURRENT\_TIMESTAMP  from deleted  insert into S\_log  select sno,CURRENT\_TIMESTAMP  from inserted  end;  create trigger Insert\_S  on student  for insert  as  insert into S\_log  select sno,CURRENT\_TIMESTAMP  from inserted        （9）为数据库学生选课中的SC表创建一个触发器是Scscoretri,要求当插入一个记录或修改成绩时，确保此记录的成绩在0—100分之间。  create trigger score\_sc\_tri  on sc for update  as  declare @score\_int int  select @score\_int=Grade from sc  if (@score\_int<=100 and @score\_int>=0)  begin  print '操作完成！' return  end  else  begin  print '成绩应在-100之间！' Rollback Transaction  end | | | | |