

中国科学技术大学

计算机科学与技术学院

# 实 验 报 告

学 号 PB16060130

姓 名 顾健鑫。

(实验) 课程名称 数据库系统及应用

教 师 金培权

# 实 验 报 告

学生姓名： 顾健鑫

学号： PB16060130

指导教师： 金培权

实验时间 :2019 年 4 月 19 日

## 一、 实验：SQL & PL/SQL

## 二、 实验环境

本次实验使用的实验环境为 Oracle 12c 发行版以及 PLSQL Developer 13。

操作系统环境为 Windows 7。

## 三、 实验设计

### 1. 本次实验首先需要建立以下三个基本表

- Book (ID: char(8), name: varchar2(10), author: varchar2(10), price: float, status: int)

其中 ID 为主键 , 数名不可为空 , 状态 1 表示借出 , 0 表示在馆 , 默认值为 0。

- Reader (ID: char(8), name: varchar2(10), age: int, address: varchar2(20))  
ID 为主键。

- Borrow (book\_ID: char(8), Reader\_ID: char(8), Borrow\_Date: date, Return\_Date: date)

其中 Return\_Date 为 NULL 表示未还。主键为 ( 图书号 , 读者号 ) , 书号和读者号均为外键。

### 2. 验证实体完整性、参照完整性以及用户自定义完整性

### 3. 利用 SQL 语言完成一系列查询任务

- (1) 检索读者 Rose 的读者号和地址；
- (2) 检索读者 Rose 所借阅读书（包括已还和未还图书）的图书名和借期；
- (3) 检索未借阅图书的读者姓名；
- (4) 检索 Ullman 所写的书的书名和单价；
- (5) 检索读者“李林”借阅未还的图书的图书号和书名；
- (6) 检索借阅图书数目超过 3 本的读者姓名；
- (7) 检索没有借阅读者“李林”所借的任何一本书的读者姓名和读者号；
- (8) 检索书名中包含“Oracle”的图书书名及图书号；
- (9) 创建一个读者借书信息的视图，该视图包含读者号、姓名、所借图书号、图书名和借期；并使用该视图查询最近一年所有读者的读者号以及所借阅的不同图书数；

#### 4. 设计存储过程，完成对 Book 表中 ID 的修改

在实现时，因为 Borrow 表中引用了 Book 表的 ID，因此不可以直接修改，要首先移除引用关系，在修改完后，再把引用关系恢复即可。

#### 5. 设计触发器，当一本书被借出或者归还时，自动修改 status。

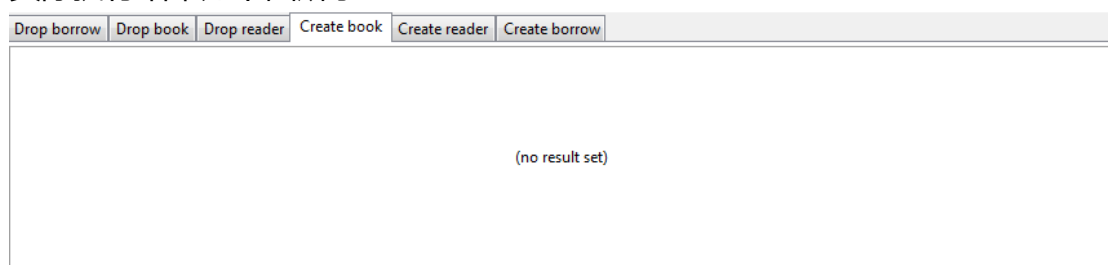
触发器的条件则为当 Insert 或者 Update Borrow 表的时候，都需要触发。当插入新条目时，return\_date 为 NULL，而 Update 实现归还时，return\_date 不为 NULL，因此可以根据 return\_date 是否为 NULL 来判断 status 的取值。

## 四、实验结果与分析

### 1. 建表操作

本次实验中，建表操作通过如下语句完成。由于在调试时需要重复执行，因此在建表前，加入了 Drop 操作删除之前创建的表。

实际执行结果如下图所示

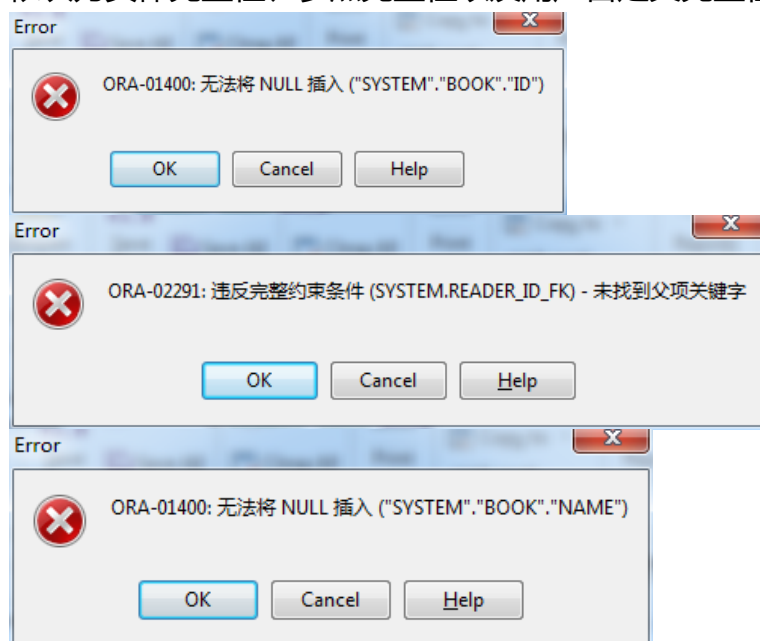


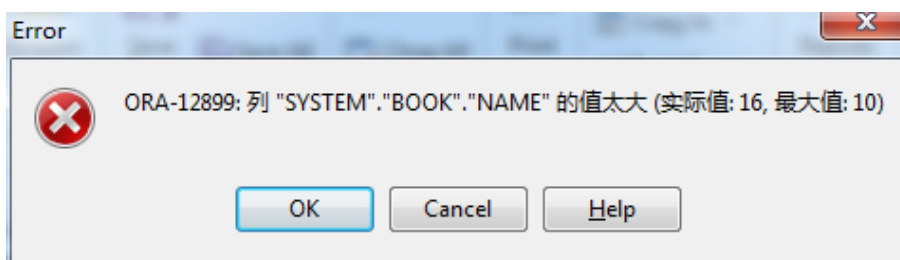
### 2. 验证三条完整性

验证完整性的语句位于 integrity\_check.sql 中

分别验证了三种完整性。

依次为实体完整性、参照完整性以及用户自定义完整性。





### 3. 触发器设计

触发器将根据新的 Borrow 条目的 Return\_Date 是否为 NULL 来判断 status 将取何值。

在执行 insert\_data.sql 中的语句后, 查询语句 search \* from Book 的结果。

	ID	NAME	AUTHOR	PRICE	STATUS
1	00000001	护理	赵本山	20	0
2	00000002	Data Base	C.J.Date	60	0
3	00000003	软工	S.Lawrence	75	0
4	00000004	AI Intro	S.Russell	80	0
5	00000005	算法	T.H.Cormen	100	0
6	00000006	并行	陈国良	75	0
7	00000007	电磁学	张之翔	45.5	0
8	00000008	Oracle_	Ullman	45.5	0
9	00000009	Oracle1	Ullman	45.5	0
10	00000010	Oracle3	Ullman	45.5	0
11	00000011	29Oracle	Ullman	45.5	0

在执行 add\_borrow.sql 中的语句后, 查询语句的结果如下。

	ID	NAME	AUTHOR	PRICE	STATUS
1	00000001	护理	赵本山	20	1
2	00000002	Data Base	C.J.Date	60	1
3	00000003	软工	S.Lawrence	75	0
4	00000004	AI Intro	S.Russell	80	0
5	00000005	算法	T.H.Cormen	100	0
6	00000006	并行	陈国良	75	0
7	00000007	电磁学	张之翔	45.5	0
8	00000008	Oracle_	Ullman	45.5	0
9	00000009	Oracle1	Ullman	45.5	0
10	00000010	Oracle3	Ullman	45.5	0
11	00000011	29Oracle	Ullman	45.5	0

可以看到, 根据 add\_borrow.sql 中的语句, 1 和 2 号书尚未归还, 因此其 status 值为 1, 其余的为 0。

### 4. 查询数据库

根据实验文档给出的查询要求, 具体的 SQL 语句实现如附录代码所示

根据验证, 每条查询语句都可以完成指定的任务。在此不再截图展示。

### 5. 修改 Book 表中的 ID

根据过程的定义, 执行语句 change\_bid('00000001', 'B001');后, search \* from Book 的查询结果为下图所示。可以看到原 ID 为 00000001 的书本 ID 变为 B001, 修改成功。

	ID	NAME	AUTHOR	PRICE	STATUS
▶ 1	B001	护理	赵本山	20	1
2	00000002	Data Base	C.J.Date	60	1
3	00000003	软工	S.Lawrence	75	0
4	00000004	AI Intro	S.Russell	80	0
5	00000005	算法	T.H.Cormen	100	0
6	00000006	并行	陈国良	75	0
7	00000007	电磁学	张之翔	45.5	0
8	00000008	Oracle_	Ullman	45.5	0
9	00000009	Oracle1	Ullman	45.5	0
10	00000010	Oracle3	Ullman	45.5	0
11	00000011	29Oracle	Ullman	45.5	0

## 五、 评价与收获

通过本次实验，我对 Oracle DBMS 以及 SQL 语句的编写有了更深入的了解。在配置安装环境以及 debug 时，对于数据库的运行及其特性有了深刻的认知，如不同 SQL 文件在执行后需要 commit 才可实现对数据库的更改。

## 六、 附录代码：

代码名称 1: change\_bid.prc

```
create or replace procedure change_bid(old_id in char, new_id in char)
is
begin
    execute immediate 'alter table Borrow drop constraint book_ID_FK';
    update book set id = new_id where id = old_id;
    update borrow set book_id = new_id where book_id = old_id;
    execute immediate 'alter table Borrow add constraint book_ID_FK Foreign key(book_ID)
references Book(ID)';
end change_bid;
```

代码名称 2: set\_status.trg

```
create or replace trigger SetStatus
After Insert OR Update
on Borrow
for each row
begin
    If :new.return_date IS NULL Then
        Update Book Set status = 1 where ID = :new.book_ID;
    Else
        Update Book Set status = 0 where ID = :new.book_ID;
    end if;
end;
```

代码名称 3: lab01.sql

```
drop table borrow;
drop table book;
drop table reader;
```

```

create table book(
    ID      char(8) constraint book_PK Primary Key,
    name    varchar2(10) not null,
    author  varchar2(10),
    price   float,
    status  int default 0
);

create table reader(
    ID      char(8) constraint reader_PK Primary Key,
    name    varchar2(10),
    age     int,
    address varchar2(20)
);

create table borrow(
    book_ID      char(8),
    reader_ID     char(8),
    borrow_date   date,
    return_date   date,
    constraint book_ID_FK foreign key(book_ID) references book(ID) on delete
cascade,
    constraint reader_ID_FK foreign key(reader_ID) references reader(ID) on
delete cascade,
    constraint borrow_PK Primary Key(book_ID, reader_ID)
);

```

代码名称 4: integrity\_check.sql

```

insert into book
values(NULL, 'COD', 'USTC', 60, 0);

insert into borrow
values('909022', '1928379', to_date('2019-1-1', 'yyyy-mm-dd'), NULL);

insert into book
values('0000111', NULL, 'Billy Van', 90, 0);

insert into book
values('0000110', 'Operating system', NULL, 100, 0);

```

代码名称 5: insert\_data.sql

```

Insert Into book Values('00000001', '护理', '赵本山', 20, 0);
Insert Into book Values('00000002', 'Data Base', 'C.J.Date', 60, 0);
Insert Into book Values('00000003', '软工', 'S.Lawrence', 75, 0);

```

```

Insert Into book Values('00000004', 'AI Intro', 'S.Russell',80, 0);
Insert Into book Values('00000005', '算法', 'T.H.Cormen',100, 0);
Insert Into book Values('00000006', '并行', '陈国良',75, 0);
Insert Into book Values('00000007', '电磁学', '张之翔', 45.5, 0);
Insert Into book Values('00000008', 'Oracle_', 'Ullman', 45.5, 0);
Insert Into book Values('00000009', 'Oracle1', 'Ullman', 45.5, 0);
Insert Into book Values('00000010', 'Oracle3', 'Ullman', 45.5, 0);
Insert Into book Values('00000011', '290racle', 'Ullman', 45.5, 0);

Insert Into reader Values('0001', 'Rose', 20, 'Hefei');
Insert Into reader Values('0002', '李林', 60, 'New York');
Insert Into reader Values('0005', 'Billy', 50, 'Long Island ');
Insert Into reader Values('0006', 'Van', 45, 'Shinnipori');
Insert Into reader Values('0007', 'Obama', 48, 'Washington D.C. ');
Insert Into reader Values('0008', 'Trump', 50, 'Brazil');

```

代码名称 6: add\_borrow.sql

```

insert into borrow values('00000001', '0002', to_date('2019-3-20',
'YYYY-MM-DD'),NULL);
update borrow set return_date = to_date('2019-4-17', 'YYYY-MM-DD')
    where book_id = '00000001' and reader_id = '0002' and borrow_date =
to_date('2019-3-20', 'YYYY-MM-DD');

insert into borrow values('00000001', '0001', to_date('2019-4-19',
'YYYY-MM-DD'),NULL);

insert into borrow values('00000002', '0001', to_date('2019-1-10',
'YYYY-MM-DD'),NULL);
update borrow set return_date = to_date('2019-3-8', 'YYYY-MM-DD')
    where book_id = '00000002' and reader_id = '0002' and borrow_date =
to_date('2019-1-10', 'YYYY-MM-DD');

insert into borrow values('00000002', '0002', to_date('2019-3-19',
'YYYY-MM-DD'),NULL);

insert into borrow values('00000003', '0001', to_date('2017-12-20',
'YYYY-MM-DD'),NULL);
update borrow set return_date = to_date('2018-2-17', 'YYYY-MM-DD')
    where book_id = '00000003' and reader_id = '0001' and borrow_date =
to_date('2017-12-20', 'YYYY-MM-DD');

insert into borrow values('00000004', '0001', to_date('2017-3-20',
'YYYY-MM-DD'),NULL);
update borrow set return_date = to_date('2017-4-14', 'YYYY-MM-DD')

```

```

        where book_id = '00000004' and reader_id = '0001' and borrow_date =
to_date('2017-3-20', 'YYYY-MM-DD');

insert into borrow values('00000005', '0002', to_date('2018-2-26',
'YYYY-MM-DD'),NULL);
update borrow set return_date = to_date('2018-7-17', 'YYYY-MM-DD')
        where book_id = '00000005' and reader_id = '0002' and borrow_date =
to_date('2018-2-26', 'YYYY-MM-DD');

insert into borrow values('00000005', '0005', to_date('2019-4-1',
'YYYY-MM-DD'),NULL);
update borrow set return_date = to_date('2019-4-19', 'YYYY-MM-DD')
        where book_id = '00000005' and reader_id = '0005' and borrow_date =
to_date('2019-4-1', 'YYYY-MM-DD');

insert into borrow values('00000005', '0006', to_date('2019-2-12',
'YYYY-MM-DD'),NULL);
update borrow set return_date = to_date('2019-3-15', 'YYYY-MM-DD')
        where book_id = '00000005' and reader_id = '0006' and borrow_date =
to_date('2019-2-12', 'YYYY-MM-DD');

insert into borrow values('00000006', '0005', to_date('2016-3-20',
'YYYY-MM-DD'),NULL);
update borrow set return_date = to_date('2016-6-18', 'YYYY-MM-DD')
        where book_id = '00000006' and reader_id = '0005' and borrow_date =
to_date('2016-3-20', 'YYYY-MM-DD');

insert into borrow values('00000007', '0008', to_date('2017-12-25',
'YYYY-MM-DD'),NULL);
update borrow set return_date = to_date('2018-1-31', 'YYYY-MM-DD')
        where book_id = '00000007' and reader_id = '0008' and borrow_date =
to_date('2017-12-25', 'YYYY-MM-DD');

insert into borrow values('00000009', '0006', to_date('2017-12-31',
'YYYY-MM-DD'),NULL);
update borrow set return_date = to_date('2018-2-1', 'YYYY-MM-DD')
        where book_id = '00000009' and reader_id = '0006' and borrow_date =
to_date('2017-12-31', 'YYYY-MM-DD');

insert into borrow values('00000010', '0005', to_date('2018-9-10',
'YYYY-MM-DD'),NULL);
update borrow set return_date = to_date('2018-11-13', 'YYYY-MM-DD')
        where book_id = '00000010' and reader_id = '0005' and borrow_date =
to_date('2018-9-10', 'YYYY-MM-DD');

```



代码名称 7: query.sql

```
/*1*/
select ID, address from reader
      where name = 'Rose';

/*2*/
select b.name, c.borrow_date from reader a, book b, borrow c
      where a.id = c.reader_id and b.id = c.book_id and a.name = 'Rose';

/*3*/
select name from(
      select name, id from reader where
      id not in(select distinct reader_ID from borrow)
);

/*4*/
select name, price from book where author = 'Ullman';

/*5*/
select b.id, b.name from reader a, book b, borrow c
      where a.id = c.reader_id and b.id = c.book_id
      and c.return_date IS NULL and a.name = '李林';

/*6*/
select a.name from reader a, borrow c
      where a.id = c.reader_id
      group by a.id, a.name having COUNT(*)>3;

/*7*/
select a.name, a.id from reader a
      where NOT EXISTS(
      select * from reader a1, borrow c1
      where a1.id = c1.reader_id and a1.name = '李林'
      and EXISTS(
      select * from reader a2, borrow c2 where
      a2.id = c2.reader_id and c1.book_id = c2.book_id and a.id = a2.id
      )
      );

/*8*/
select name, id from book where name like '%Oracle%';

/*9*/
```

```
Drop View borrow_info;
```

```
create view borrow_info(rid, rname, bid, bname, borrow_date)
as select a.id, a.name, b.id, b.name, c.borrow_date
from reader a, book b, borrow c
where a.id = c.reader_id and b.id = c.book_id;
```

```
select rname, borrow_num
from(
    select rname, rid, count(*) as borrow_num
    from borrow_info
    where borrow_date > to_date('2018-1-1', 'YYYY-MM-DD')
    group by rname, rid
);
```