数据库系统与应用第一次实验报告

实验目的

熟悉 p1/sq1 语法, 熟悉数据库的基本操作。

实验环境

操作系统: Window10 64 位操作系统

数据库: oracle12c 数据库,

开发工具: pl/sql developer 64-bit, sql plus

实验内容

1、 创建三个实验要求的基本表,并插入部分测试数据;

创建基本表:

```
CREATE table book(
       id char(8) PRIMARY KEY,
       name varchar2(20) not null,
       author varchar(10),
       price number,
       status number default 0
    create table reader(
       id char(8) primary key,
       name varchar2(10),
11
12
       age int,
       address varchar2(20)
    create table borrow(
    book_id char(8),
18 reader id char(8),
19 borrow date date,
20 return date date,
    primary key(book_id,reader_id),
constraint book_idfk foreign key (book_id) references book(id),
constraint reader_idfk foreign key (reader_id) references reader(id)
24 );
```

插入数据:

```
insert all into book(id,

name,
author,
price,
status)

into book values('00000001','Oracle PL/SQL','author1','99','0')

into book values('00000003','Oracle Business','Ullman','121','0')

into book values('00000003','Oracle Business','Ullman','121','0')

into book values('00000004','book4','author4','109','0')

into book values('00000005','book5','author5','49','0')

into book values('00000007','book7','author5','49','1')

into book values('00000008','Expert Oracle)','Ullman','79','1')

into book values('00000009','book9','Ullman','21','0')

into book values('00000001','book9','Ullman','21','0')

into book values('00000012','book12','author12','51','0')

into book values('00000013','book13','author13','69','0')

into book values('00000015','book15','author15','28','0')

into book values('00000016','book16','author15','28','0')

into book values('00000017','book17','author17','19','0')

select 1 from dual;
```

```
52
    insert into reader(id,
                        name,
                        age,
                        address)
                              '10000001', 'reader1', '18', 'address1' from dual
                              '10000002', 'Rose', '28', 'address2' from dual
                              '10000003', 'mary', '48', 'address3' from dual
                              '10000004','reader4','16','address4' from dual
                              '10000005', 'reader5', '21', 'address5' from dual
                              '10000006', 'reader6', '32', 'address6' from dual
                              '10000007', 'reader7', '22', 'address7' from dual
                              '10000008','reader8','21','address8' from dual
                              '10000009','reader9','20','address9' from dual
                              '10000010','reader10','33','address10' from dual
                              '10000011','reader11','19','address11' from dual
                              '10000012', 'reader12', '15', 'address12' from dual
                       select '10000013','reader13','41','address13' from dual
```

```
| Insert into borrow(book_id, reader_id, borrow_date, return_date) | Proceedings of the process of the process
```

三个表的数据:

		ID	NAME		AUTHOR	PRICE	STATUS
	1	00000001	Oracle PL/SQL		author1	99	0
	2	00000002	Oracle Java		author2	10	1
	3	00000003	Oracle Business	•••	Ullman	121	0
	4	00000004	book4		author4	109	0
	5	00000005	book5	•••	author5	49	0
	6	00000006	book5		author5	49	1
	7	00000007	book7	•••	author7	35	0
	8	80000000	Expert Oracle)		Ullman	79	1
	9	00000009	book9	•••	Ullman	21	0
	10	00000010	book9		Ullman	21	0
	11	00000011	book9		Ullman	21	0
	12	00000012	book12		author12	51	0
	13	00000013	book13		author13	69	0
	14	00000014	book14		author14	78	0
	15	00000015	book15		author15	28	0
	16	00000016	book16		author16	5	1
	17	00000017	book17		author17	19	0

		ID	NAME	AGE	ADDRESS	
•	1	10000001	reader1	18	address1	
	2	10000002	Rose	28	address2	
	3	10000003	mary	48	address3	
	4	10000004	reader4	16	address4	
	5	10000005	reader5	21	address5	
	6	10000006	reader6	32	address6	
	7	10000007	reader7	22	address7	
	8	10000008	reader8	21	address8	
	9	10000009	reader9	20	address9	
	10	10000010	reader10	33	address10	
	11	10000011	reader11	19	address11	
	12	10000012	reader12	15	address12	
	13	10000013	reader13	41	address13	

		BOOK_ID	READER_ID	BORROW_DATE		RETURN_DATE	
•	1	00000001	10000003	2018/2/21 15:21:14	•	2018/3/21 15:21:56	•
	2	00000001	10000006	2017/12/20 18:31:34	•	2018/1/20 18:31:34	•
	3	00000002	10000002	2017/12/21 15:21:14	•		•
	4	00000003	10000002	2017/12/21 15:21:56	•	2018/2/21 15:21:56	•
	5	00000003	10000003	2018/2/21 15:21:14	•	2018/3/21 15:21:56	•
	6	00000005	10000002	2017/12/31 10:24:35	•	2018/2/25 18:31:34	•
	7	00000005	10000009	2017/1/20 8:11:34	•	2018/2/15 10:11:12	•
	8	00000006	10000002	2018/3/20 12:13:24	•		•
	9	00000007	10000011	2017/1/20 8:11:34	•	2018/2/15 10:11:12	•
	10	80000000	10000003	2017/12/20 18:24:59	•		•
	11	00000011	10000003	2017/12/25 12:21:14	•	2018/1/10 14:31:34	•
	12	00000011	10000004	2017/1/20 8:11:34	•	2018/2/15 10:11:12	•
	13	00000012	10000007	2017/1/20 8:11:34	•	2018/2/15 10:11:12	•
	14	00000015	10000013	2017/11/2 18:31:12	•		•

2、 设计例子,验证实体完整性、参照完整性、用户自定义完整性;

以上插入均不能完成。

- 3、 用 SQL 语言完成下面小题, 并测试运行结果:
 - (1) 检索读者 Rose 的读者号和地址;

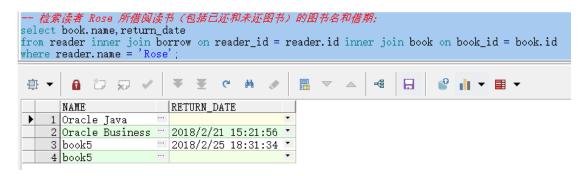
```
--检察读者 Rose 的读者号和地址:
select id, address
from reader
where name = 'Rose';

ID ADDRESS

▶ 1 100000002 address2 ...
```

(2) 检索读者 Rose 所借阅读书(包括已还和未还图书)的图书名

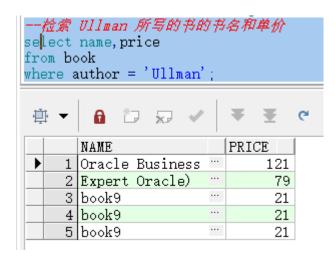
和借期:



(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;



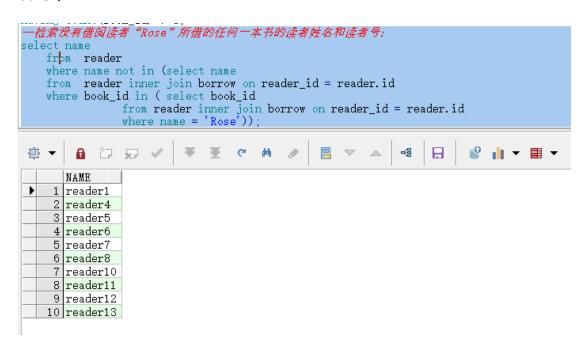
(5) 检索读者 "Rose"借阅未还的图书的图书号和书名;



(6) 检索借阅图书数目超过 3 本的读者姓名;



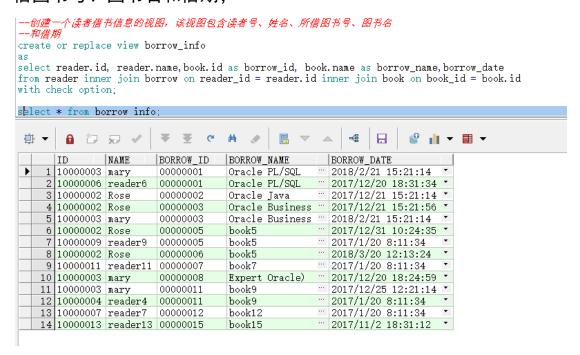
(7) 检索没有借阅读者 "Rose" 所借的任何一本书的读者姓名和读者号:



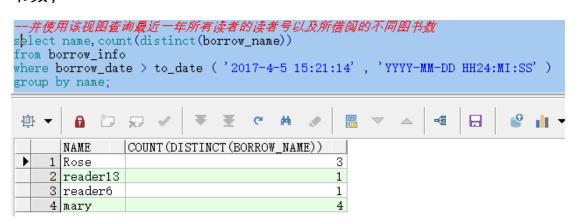
(8) 检索书名中包含 "Oracle" 的图书书名及图书号;



(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,实现对 Book 表的 ID 的修改。

由于在 pl/sql developer 上有难以解决的问题, 所以此处在 sqlplus 上实现

```
SQL> create or replace procedure Changebookid(id_old in char, id_new in char)
2 as
3 begin
4 execute immediate 'alter table borrow disable constraint book_idfk';
5 update book set book.id = id_new where book.id = id_old;
6 update borrow set book_id = id_new where book_id = id_old;
7 execute immediate 'alter table borrow enable constraint book_idfk';
8 end;
9 /

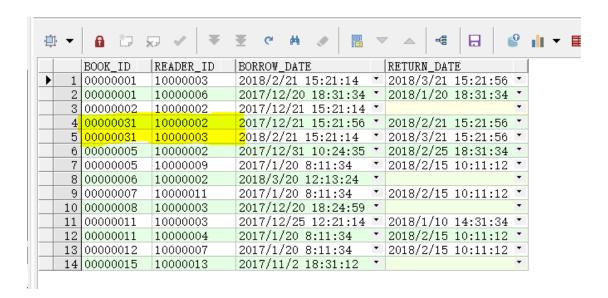
过程已创建。

SQL> exec Changebookid('000000003','000000031')

PL/SQL 过程已成功完成。

SQL>
```

Ţ	. ▼	a 🗀	₽ ✓ ₹ ₹		C # %		_ △ ≪
		ID	NAME		AUTHOR	PRICE	STATUS
Þ	1	00000001	Oracle PL/SQL	•••	author1	99	0
	2	00000002	Oracle Java		author2	10	1
	3	00000031	Oracle Business		Ullman	121	0
	4	00000004	book4		author4	109	0
	5	00000005	book5	•••	author5	49	0
	6	00000006	book5		author5	49	1
	7	00000007	book7	•••	author7	35	0
	8	80000000	Expert Oracle)		Ullman	79	1
	9	00000009	book9	•••	Ullman	21	0
	10	00000010	book9		Ullman	21	0
	11	00000011	book9	•••	Ullman	21	0
	12	00000012	book12		author12	51	0
	13	00000013	book13	•••	author13	69	0
	14	00000014	book14		author14	78	0
	15	00000015	book15	•••	author15	28	0
	16	00000016	book16		author16	5	1
	17	00000017	book17	•••	author17	19	0



5、 设计触发器,实现: 当一本书被借出时,自动将 Book 表中相应图书的 status 修改为1;当某本书被归还时,自动将 status 改为0。

由于在 pl/sql developer 上有难以解决的问题, 所以此处在 sqlplus 上实现

		ID	NAME		AUTHOR	PRICE	STATUS
)	1	00000001	Oracle PL/SQL		author1	99	0
	2	00000002	Oracle Java		author2	10	1
	3	00000031	Oracle Business	•••	Ullman	121	0
	4	00000004	book4		author4	109	0
	5	00000005	book5	•••	author5	49	0
	6	00000006	book5		author5	49	1
	7	00000007	book7	•••	author7	35	0
	8	00000008	Expert Oracle)		Ullman	79	1
	9	00000009	book9	•••	Ullman	21	0
	10	00000010	book9		Ullman	21	1
	11	00000011	book9	•••	Ullman	21	0
	12	00000012	book12		author12	51	0
	13	00000013	book13	•••	author13	69	0
	14	00000014	book14		author14	78	0
	15	00000015	book15	•••	author15	28	0
	16	00000016			author16	5	1
	17	00000017	book17	•••	author17	19	0

```
SQL> delete from borrow
2 where reader_id = '10000010';
已删除 1 行。

SQL> commit;/
2 /
commit;/
*
第 1 行出现错误:
ORA-02185: COMMIT 后面跟的标记不是 WORK

SQL> commit;
提交完成。

SQL>
```

		I	I			1	I
		ID	NAME		AUTHOR	PRICE	STATUS
•	1	00000001	Oracle PL/SQL	•••	author1	99	0
	2	00000002	Oracle Java	•••	author2	10	1
	3	00000031	Oracle Business	•••	Ullman	121	0
	4	00000004	book4	•••	author4	109	0
	5	00000005	book5	•••	author5	49	0
	6	00000006	book5		author5	49	1
	7	00000007	book7	•••	author7	35	0
	8	80000000	Expert Oracle)		Ullman	79	1
	9	00000009	book9	•••	Ullman	21	0
	10	00000010	book9		Ullman	21	0
	11	00000011	book9	•••	Ullman	21	0
	12	00000012	book12		author12	51	0
	13	00000013	book13	•••	author13	69	0
	14	00000014	book14		author14	78	0
		00000015		•••	author15	28	0
	16	00000016	book16		author16	5	1
	17	00000017	book17	•••	author17	19	0