

数据库系统与应用第一次实验报告

实验目的

熟悉 pl/sql 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：

```
1  CREATE table book(  
2      id char(8) PRIMARY KEY,  
3      name varchar2(20) not null,  
4      author varchar(10),  
5      price number,  
6      status number default 0  
7  );  
8  
9  create table reader(  
10     id char(8) primary key,  
11     name varchar2(10),  
12     age int,  
13     address varchar2(20)  
14 );  
15  
16 create table borrow(  
17     book_id char(8),  
18     reader_id char(8),  
19     borrow_date date,  
20     return_date date,  
21     primary key(book_id,reader_id),  
22     constraint book_idfk foreign key (book_id) references book(id),  
23     constraint reader_idfk foreign key (reader_id) references reader(id)  
24 );
```

插入数据：

```

27  insert all into book(id,
28      name,
29      author,
30      price,
31      status)
32      values('00000001','Oracle PL/SQL','author1','99','0')
33  into book values('00000002','Oracle Java','author2','10','1')
34  into book values('00000003','Oracle Business','Ullman','121','0')
35  into book values('00000004','book4','author4','109','0')
36  into book values('00000005','book5','author5','49','0')
37  into book values('00000006','book5','author5','49','1')
38  into book values('00000007','book7','author7','35','0')
39  into book values('00000008','Expert Oracle','Ullman','79','1')
40  into book values('00000009','book9','Ullman','21','0')
41  into book values('00000010','book9','Ullman','21','0')
42  into book values('00000011','book9','Ullman','21','0')
43  into book values('00000012','book12','author12','51','0')
44  into book values('00000013','book13','author13','69','0')
45  into book values('00000014','book14','author14','78','0')
46  into book values('00000015','book15','author15','28','0')
47  into book values('00000016','book16','author16','5','1')
48  into book values('00000017','book17','author17','19','0')
49      select 1 from dual;

```

```

52  insert into reader(id,
53      name,
54      age,
55      address)
56      select * from (
57      select '10000001','reader1','18','address1' from dual
58      union
59      select '10000002','Rose','28','address2' from dual
60      union
61      select '10000003','mary','48','address3' from dual
62      union
63      select '10000004','reader4','16','address4' from dual
64      union
65      select '10000005','reader5','21','address5' from dual
66      union
67      select '10000006','reader6','32','address6' from dual
68      union
69      select '10000007','reader7','22','address7' from dual
70      union
71      select '10000008','reader8','21','address8' from dual
72      union
73      select '10000009','reader9','20','address9' from dual
74      union
75      select '10000010','reader10','33','address10' from dual
76      union
77      select '10000011','reader11','19','address11' from dual
78      union
79      select '10000012','reader12','15','address12' from dual
80      union
81      select '10000013','reader13','41','address13' from dual
82      );

```

```

85 insert into borrow(book_id,
86                    reader_id,
87                    borrow_date,
88                    return_date)
89 select * from (
90 select '00000001','10000006',to_date ( '2017-12-20 18:31:34' , 'YYYY-MM-DD HH24:MI:SS' ),to_date ( '2018-1-20 18:31:34' , 'YYYY-MM-DD HH24:MI:SS' ) from dual
91 union
92 select '00000002','10000002',to_date ( '2017-12-21 15:21:14' , 'YYYY-MM-DD HH24:MI:SS' ),null from dual
93 union
94 select '00000001','10000003',to_date ( '2018-2-21 15:21:14' , 'YYYY-MM-DD HH24:MI:SS' ),to_date ( '2018-3-21 15:21:56' , 'YYYY-MM-DD HH24:MI:SS' ) from dual
95 union
96 select '00000003','10000003',to_date ( '2018-2-21 15:21:14' , 'YYYY-MM-DD HH24:MI:SS' ),to_date ( '2018-3-21 15:21:56' , 'YYYY-MM-DD HH24:MI:SS' ) from dual
97 union
98 select '00000003','10000002',to_date ( '2017-12-21 15:21:56' , 'YYYY-MM-DD HH24:MI:SS' ),to_date ( '2018-2-21 15:21:56' , 'YYYY-MM-DD HH24:MI:SS' ) from dual
99 union
100 select '00000005','10000002',to_date ( '2017-12-31 10:24:35' , 'YYYY-MM-DD HH24:MI:SS' ),to_date ( '2018-2-25 18:31:34' , 'YYYY-MM-DD HH24:MI:SS' ) from dual
101 union
102 select '00000006','10000002',to_date ( '2018-3-20 12:13:24' , 'YYYY-MM-DD HH24:MI:SS' ),null from dual
103 union
104 select '00000008','10000003',to_date ( '2017-12-20 18:24:59' , 'YYYY-MM-DD HH24:MI:SS' ),null from dual
105 union
106 select '00000011','10000003',to_date ( '2017-12-25 12:21:14' , 'YYYY-MM-DD HH24:MI:SS' ),to_date ( '2018-1-10 14:31:34' , 'YYYY-MM-DD HH24:MI:SS' ) from dual
107 union
108 select '00000015','10000013',to_date ( '2017-11-2 18:31:12' , 'YYYY-MM-DD HH24:MI:SS' ),null from dual
109 union
110 select '00000011','10000004',to_date ( '2017-1-20 8:11:34' , 'YYYY-MM-DD HH24:MI:SS' ),to_date ( '2018-2-15 10:11:12' , 'YYYY-MM-DD HH24:MI:SS' ) from dual
111 union
112 select '00000012','10000007',to_date ( '2017-1-20 8:11:34' , 'YYYY-MM-DD HH24:MI:SS' ),to_date ( '2018-2-15 10:11:12' , 'YYYY-MM-DD HH24:MI:SS' ) from dual
113 union
114 select '00000007','10000011',to_date ( '2017-1-20 8:11:34' , 'YYYY-MM-DD HH24:MI:SS' ),to_date ( '2018-2-15 10:11:12' , 'YYYY-MM-DD HH24:MI:SS' ) from dual
115 union
116 select '00000005','10000009',to_date ( '2017-1-20 8:11:34' , 'YYYY-MM-DD HH24:MI:SS' ),to_date ( '2018-2-15 10:11:12' , 'YYYY-MM-DD HH24:MI:SS' ) from dual
117 );

```

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	00000008	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

和借期；

```
-- 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；
select book.name, return_date
from reader inner join borrow on reader_id = reader.id inner join book on book_id = book.id
where reader.name = 'Rose';
```

	NAME	RETURN_DATE
1	Oracle Java	...
2	Oracle Business	2018/2/21 15:21:56
3	book5	2018/2/25 18:31:34
4	book5	...

(3) 检索未借阅图书的读者姓名；

```
-- 检索未借阅图书的读者姓名；
select reader.name
from reader
where reader.name not in(select reader.name
                        from reader inner join borrow on reader_id = reader.id);
```

	NAME
1	reader1
2	reader5
3	reader8
4	reader10
5	reader12

(4) 检索 Ullman 所写的书的书名和单价；

```
-- 检索 Ullman 所写的书的书名和单价
select name, price
from book
where author = 'Ullman';
```

	NAME	PRICE
1	Oracle Business	121
2	Expert Oracle)	79
3	book9	21
4	book9	21
5	book9	21

(5) 检索读者“Rose”借阅未还的图书的图书号和书名；

```
--检索读者“Rose”借阅未还的图书的图书号和书名
select book.id,book.name
from reader inner join borrow on reader_id = reader.id inner join book on book_id = book.id
where return_date is null and reader.name = 'Rose';
```

	ID	NAME
1	00000002	Oracle Java ...
2	00000006	book5

(6) 检索借阅图书数目超过 3 本的读者姓名；

```
--检索借阅图书数目超过 3 本的读者姓名
select name
from reader inner join borrow on reader.id = reader_id
group by name
having count(book_id) > 3;
```

	NAME
1	Rose
2	mary

(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；

```
--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；
select name
from reader
where name not in (select name
from reader inner join borrow on reader_id = reader.id
where book_id in (select book_id
from reader inner join borrow on reader_id = reader.id
where name = 'Rose'));
```

	NAME
1	reader1
2	reader4
3	reader5
4	reader6
5	reader7
6	reader8
7	reader10
8	reader11
9	reader12
10	reader13

(8) 检索书名中包含“Oracle”的图书书名及图书号；

--检索书名中包含“Oracle”的图书书名及图书号

```
select name ,id
from book
where name like '%Oracle%';
```

		NAME	ID
▶	1	Oracle PL/SQL	00000001
	2	Oracle Java	00000002
	3	Oracle Business	00000003
	4	Expert Oracle)	00000008

(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；

--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

```
create or replace view borrow_info
as
select reader.id, reader.name,book.id as borrow_id, book.name as borrow_name,borrow_date
from reader inner join borrow on reader_id = reader.id inner join book on book_id = book.id
with check option;
```

```
select * from borrow info;
```

	ID	NAME	BORROW_ID	BORROW_NAME	BORROW_DATE
▶	1	10000003 mary	00000001	Oracle PL/SQL	2018/2/21 15:21:14
	2	10000006 reader6	00000001	Oracle PL/SQL	2017/12/20 18:31:34
	3	10000002 Rose	00000002	Oracle Java	2017/12/21 15:21:14
	4	10000002 Rose	00000003	Oracle Business	2017/12/21 15:21:56
	5	10000003 mary	00000003	Oracle Business	2018/2/21 15:21:14
	6	10000002 Rose	00000005	book5	2017/12/31 10:24:35
	7	10000009 reader9	00000005	book5	2017/1/20 8:11:34
	8	10000002 Rose	00000006	book5	2018/3/20 12:13:24
	9	10000011 reader11	00000007	book7	2017/1/20 8:11:34
	10	10000003 mary	00000008	Expert Oracle)	2017/12/20 18:24:59
	11	10000003 mary	00000011	book9	2017/12/25 12:21:14
	12	10000004 reader4	00000011	book9	2017/1/20 8:11:34
	13	10000007 reader7	00000012	book12	2017/1/20 8:11:34
	14	10000013 reader13	00000015	book15	2017/11/2 18:31:12

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；

--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

```
select name,count(distinct(borrow_name))
from borrow_info
where borrow_date > to_date ( '2017-4-5 15:21:14' , 'YYYY-MM-DD HH24:MI:SS' )
group by name;
```

	NAME	COUNT (DISTINCT (BORROW_NAME))
▶	1 Rose	3
	2 reader13	1
	3 reader6	1
	4 mary	4

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，所以此处在此 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('00000003','00000031')
```

PL/SQL 过程已成功完成。

```
SQL>
```

	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	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

	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	00000031	10000002	2017/12/21 15:21:56	2018/2/21 15:21:56
5	00000031	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	00000008	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	

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，所以此处在此 sqlplus 上实现

```
SQL> create or replace trigger borrow_status
2  after insert or delete on borrow
3  for each row
4  begin
5  if inserting then
6  update book
7  set book.status = 1
8  where book.id = :new.book_id;
9  else
10 update book
11 set book.status = 0
12 where book.id = :old.book_id;
13 end if;
14 end;
15 /

触发器已创建

SQL> insert into borrow(book_id,
2  reader_id,
3  borrow_date,
4  return_date)
5  values('00000010','10000010',to_date('2017-12-25 12:21:14','YYYY-MM-DD HH24:MI:SS'),null);

已创建 1 行。

SQL> commit
2 /

提交完成。

SQL>
```

	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	book16	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>
```

	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	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