

Task 1

```
create table account
(
    a_name varchar(30) not null
        primary key,
    a_pass varchar(10) not null,
    check (a_pass regexp _utf8mb4'[a-zA-Z0-9]{4,10}')
```

```
);

create table book
(
    b_isbn char(13) not null
        primary key,
    b_name varchar(30) not null,
    b_num int default 0 null
);

create table record
(
    r_aname varchar(30) not null,
    r_bisbn char(13) not null,
    r_ltime date not null,
    r_etime date not null,
    r_rtime date null,
    primary key (r_aname, r_bisbn),
    foreign key (r_aname) references account (a_name),
    foreign key (r_bisbn) references book (b_isbn)
);
```

Task 1-1

```
DELIMITER $$

CREATE PROCEDURE update_password(
    in user_name varchar(30),
    in password_old varchar(20),
    in password_new varchar(20),
    in action int
)
BEGIN
    declare user_exists int default 0;
    declare user_cursor cursor for select count(*) from account where a_name =
user_name and a_pass = password_old;
    open user_cursor;
    fetch user_cursor into user_exists;
    close user_cursor;
    if user_exists = 0 then
        select false;
```

```

elseif action = 1 then
    select true;
elseif action = 2 then
    if password_new regexp '^[a-zA-Z0-9]{4,10}$' then
        update account set a_pass = password_new where a_name = user_name;
        select true;
    else
        select false;
    end if;
end if;

END$$
DELIMITER ;

```

Task 1-2

```

create
    definer = root@`%` procedure borrow_book(IN user_name varchar(30), IN isbn
varchar(13))
BEGIN
    declare num int;
    select b_num from book where b_isbn=isbn into num;
    if not exists(select * from book where b_isbn=isbn) or num=0 or
        not exists(select * from account where a_name=user_name) or
        exists(select * from record where r_aname=user_name and r_bisbn=isbn)
    then
        select false;
    else
        insert into record(r_aname, r_bisbn, r_ltime, r_etime, r_rtime)
            values (user_name, isbn, curdate(), date_add(curdate(), interval 30
day), null);
        update book set b_num=num-1 where b_isbn=isbn;
        select true;
    end if;
END;

```

```
1 ✓ call borrow_book( user_name: 'cary', isbn: '1231231231233');  
2 call borrow_book( user_name: 'cary', isbn: '1231231231232');  
3 call borrow_book( user_name: 'cary', isbn: '1231231231231');
```

输出 Result 3

1行

false

1	0
---	---

```
1 call borrow_book( user_name: 'cary', isbn: '1231231231233');  
2 ✓ call borrow_book( user_name: 'cary', isbn: '1231231231232');  
3 call borrow_book( user_name: 'cary', isbn: '1231231231231');
```

输出 Result 5

1行

false

1	0
---	---

```
1 call borrow_book( user_name: 'cary', isbn: '1231231231233');  
2 call borrow_book( user_name: 'cary', isbn: '1231231231232');  
3 ✓ call borrow_book( user_name: 'vioriey', isbn: '1231231231231');
```

输出 Result 12

1行

true

1	1
---	---

Task 1-3

```
DELIMITER $$

CREATE PROCEDURE return_book(
    in user_name varchar(30),
    in isbn varchar(13)
)
BEGIN
    declare num int;
    if not exists(select * from record where r_aname=user_name and r_bisbn=isbn)
then
        select false;
    else
        update record set r_rtime=curdate() where r_aname=user_name and
r_bisbn=isbn;
        select b_num from book where b_isbn=isbn into num;
        update book set b_num=num+1 where b_isbn=isbn;
        select true;
    end if;
END$$
DELIMITER ;
```

1 ✓

call return_book(user_name: 'cary', isbn: '1231231231233');

2

call return_book(user_name: 'cary', isbn: '1231231231232');

3

call return_book(user_name: 'vioriey', isbn: '1231231231231');

输出

Result 13

1行

false

10

1

call return_book(user_name: 'cary', isbn: '1231231231233');

2 ✓

call return_book(user_name: 'cary', isbn: '1231231231232');

3

call return_book(user_name: 'vioriey', isbn: '1231231231231');

输出

Result 16

1行

true

11

1

call return_book(user_name: 'cary', isbn: '1231231231233');

2

call return_book(user_name: 'cary', isbn: '1231231231232');

3 ✓

call return_book(user_name: 'vioriey', isbn: '1231231231231');

输出

Result 17

1行

true

11

Task 1-4

```
DELIMITER $$
```

```
CREATE PROCEDURE check_record(
```

```
    in user_name varchar(30)
```

```
)
```

```
    BEGIN
```

```
        select r_aname, r_bisbn, r_etime from record where r_aname=user_name and  
r_rtime is null;
```

```
    END$$
```

```
DELIMITER ;
```

1 ✓ `call check_record(user_name: 'cary');`
2 `call check_record(user_name: 'carry');`
3 `call check_record(user_name: 'vioriey');`

输出 Result 19 ×

	r_aname	r_bisbn	r_etime
1	cary	1231231231235	2023-05-27
2	carry	1231231231236	2023-05-27

1 `call check_record(user_name: 'cary');`
2 ✓ `call check_record(user_name: 'carry');`
3 `call check_record(user_name: 'vioriey');`

输出 Result 20 ×

	r_aname	r_bisbn	r_etime
1	carry	1231231231234	2023-05-27
2	carry	1231231231236	2023-05-27

1 `call check_record(user_name: 'cary');`
2 `call check_record(user_name: 'carry');`
3 ✓ `call check_record(user_name: 'vioriey');`

输出 Result 21 ×

	r_aname	r_bisbn	r_etime
1	vioriey	1231231231234	2023-05-27
2	vioriey	1231231231235	2023-05-27
3	vioriey	1231231231236	2023-05-27

Task 2

```
create table tableA(  
    id mediumint unsigned primary key auto_increment,  
    sparse mediumint unsigned not null check ( 0 <= sparse <= 5000000 ),  
    dense tinyint not null check ( 0 <= dense <= 9 )  
);
```

```

create procedure insert_tableA(
    in count int
)
BEGIN
    declare i int default 1;
    while i <= count do
        insert into tablea(sparse, dense) values (FLOOR(RAND() * 5000001),
FLOOR(RAND() * 10));
        set i=i+1;
    end while;
end;

```

Database [server] Console [localhost] insert_

3,999,501-4,000,000 /4,000,000

WHERE

	id	sparse	dense
3999501	3999501	1811734	1
3999502	3999502	4382592	8
3999503	3999503	1963636	5
3999504	3999504	2865760	2
3999505	3999505	1776710	1
3999506	3999506	2965654	5
3999507	3999507	357042	6
3999508	3999508	92927	1
3999509	3999509	3552247	0
3999510	3999510	1656333	3
3999511	3999511	4500108	3
3999512	3999512	595046	5
3999513	3999513	804920	2
3999514	3999514	4938070	0
3999515	3999515	1579045	4
3999516	3999516	559227	3
3999517	3999517	1413923	4