

Task 1

1. 新建数据库 Library

```
create database Library;
```

2. 新建如下表：（键与约束自行合理设定即可，字段可自由增加，比如ID）

账户(用户名, 密码, ...)

书库(ISBN, 书名, 数量, ...)

借阅记录(用户名, ISBN, 书名, 借书时间, 到期时间, 还书时间, ...)

PS：到期时间约定为借书时间+30天。

```
create table 账户 (  
    用户名 char(10) primary key,  
    密码 char(10)  
);  
  
create table 书库 (  
    ISBN char(13) primary key,  
    书名 varchar(50),  
    数量 int,  
    unique index (ISBN, 书名)  
);  
  
create table 借阅记录 (  
    用户名 char(10),  
    ISBN char(13),  
    书名 varchar(50),  
    借书时间 datetime,  
    到期时间 datetime,  
    还书时间 datetime,  
    primary key (用户名, ISBN, 借书时间),  
    foreign key (用户名)  
        references 账户 (用户名),  
    foreign key (ISBN, 书名)  
        references 书库 (ISBN, 书名),  
    check (date(date_add(借书时间, interval 30 day)) = date(到期时间))  
);
```

Task 2

通过存储过程或者自定义函数实现如下用户操作：

1. 注册（注册时密码只包含数字和字母，长度大于等于4、小于等于10；用户名冲突时不能注册）

```

create procedure register (user_name char(10), pass_word char(10), out msg char(20))
begin
    if not pass_word regexp '^[a-zA-Z0-9]{4,10}$' then
        set msg = '密码必须为4-10位字母或数字';
    elseif exists (
        select *
        from 账户
        where 用户名 = user_name
    ) then
        set msg = '用户名已存在';
    else
        insert into 账户 values (user_name, pass_word);
        set msg = '注册成功';
    end if;
end;

```

2. 登录

```

create procedure login (user_name char(10), pass_word char(10), out msg char(20))
begin
    if exists (
        select *
        from 账户
        where 用户名 = user_name and 密码 = pass_word
    ) then set msg = '登录成功';
    else set msg = '用户名或密码错误';
    end if;
end;

```

3. 修改密码（也需要满足注册时的密码要求）

```

create procedure updatePassword (user_name char(10), new_pass_word char(10),
                                out msg char(20))
begin
    if not exists (
        select *
        from 账户
        where 用户名 = user_name
    ) then set msg = '用户名不存在';
    elseif not new_pass_word regexp '^[a-zA-Z0-9]{4,10}$' then
        set msg = '密码必须为4-10位字母或数字';
    else
        update 账户 set 密码 = new_pass_word where 用户名 = user_name;
        set msg = '修改密码成功';
    end if;
end;

```

4. 借书（需要考虑没有足够的书或者用户不存在等特殊情况，规定一个人不能同时借两本同样的书）

```

create procedure borrowBook (user_name char(10), book_no char(13), out msg char(30))
begin
    declare book_name varchar(50);
    if not exists (
        select *
        from 账户
        where 用户名 = user_name
    ) then
        set msg = '用户名不存在';
    elseif (select 数量 from 书库 where ISBN = book_no) < 1 then
        set msg = '没有足够的书';
    elseif exists (
        select *
        from 借阅记录
        where 用户名 = user_name and ISBN = book_no and 还书时间 is null
    ) then
        set msg = '一个人不能同时借两本同样的书';
    else
        select 书名 into book_name from 书库 where ISBN = book_no;
        insert into 借阅记录 values
            (user_name, book_no, book_name, now(), date_add(now(), interval 30 day), null);
        update 书库 set 数量 = 数量 - 1 where ISBN = book_no;
        set msg = '借书成功';
    end if;
end;

```

5. 还书（同样需要考虑参数不合法情况）

```

create procedure returnBook (user_name char(10), book_no char(13), out msg char(20))
begin
    if not exists (
        select *
        from 借阅记录
        where 用户名 = user_name and ISBN = book_no
    ) then
        set msg = '借阅记录不存在';
    else
        update 借阅记录 set 还书时间 = now()
        where 用户名 = user_name and ISBN = book_no;
        update 书库 set 数量 = 数量 + 1 where ISBN = book_no;
        set msg = '还书成功';
    end if;
end;

```

6. 查看当前借阅记录(用户名, 书名, 到期时间)

```
create procedure getBorrowRecords ()
begin
    select 用户名, 书名, 到期时间
    from 借阅记录
    where 还书时间 is null;
end; //
```

7. 查看当前超期借阅记录(用户名, 书名, 超期了多久)

```
create procedure getOverdueRecords ()
begin
    select 用户名, 书名, datediff(now(), 到期时间) as 超期天数
    from 借阅记录
    where 还书时间 is null and now() > 到期时间;
end; //
```

测试

```
mysql> call register('hzy', 'abc123@?!', @msg);
Query OK, 0 rows affected (0.02 sec)
```

```
mysql> select @msg;
+-----+
| @msg          |
+-----+
| 密码必须为4-10位字母或数字 |
+-----+
1 row in set (0.00 sec)
```

```
mysql> call register('hzy', 'abc123', @msg);
Query OK, 1 row affected (0.01 sec)
```

```
mysql> select @msg;
+-----+
| @msg          |
+-----+
| 注册成功      |
+-----+
1 row in set (0.00 sec)
```

```
mysql> call login('hzy', 'abc1234', @msg);
Query OK, 0 rows affected (0.00 sec)
```

```
mysql> select @msg;
+-----+
| @msg          |
+-----+
```

```

| 用户名或密码错误 |
+-----+
1 row in set (0.00 sec)

mysql> call login('hzy', 'abc123', @msg);
Query OK, 0 rows affected (0.00 sec)

mysql> select @msg;
+-----+
| @msg |
+-----+
| 登录成功 |
+-----+
1 row in set (0.00 sec)

mysql> call updatePassword('testuser', '123', @msg);
Query OK, 0 rows affected (0.00 sec)

mysql> select @msg;
+-----+
| @msg |
+-----+
| 用户名不存在 |
+-----+
1 row in set (0.00 sec)

mysql> call updatePassword('hzy', '123abc', @msg);
Query OK, 1 row affected (0.01 sec)

mysql> select @msg;
+-----+
| @msg |
+-----+
| 修改密码成功 |
+-----+
1 row in set (0.00 sec)

mysql> insert into 书库 values
    -> ('9787544784207', '零κ', 10),
    -> ('9787544785600', '抑郁症', 5),
    -> ('9787544784382', '局外人 西绪福斯神话', 3),
    -> ('9787540790356', '海东五百年', 1);
Query OK, 4 rows affected (0.00 sec)
Records: 4 Duplicates: 0 Warnings: 0

mysql> call borrowBook('testuser', '9787544784207', @msg);
Query OK, 0 rows affected (0.00 sec)

mysql> select @msg;

```

```

+-----+
| @msg  |
+-----+
| 用户名不存在 |
+-----+
1 row in set (0.00 sec)

mysql> call borrowBook('hzy', '9787540790356', @msg);
Query OK, 0 rows affected (0.00 sec)

mysql> select @msg;
+-----+
| @msg  |
+-----+
| 没有足够的书 |
+-----+
1 row in set (0.00 sec)

mysql> call borrowBook('hzy', '9787544784382', @msg);
Query OK, 1 row affected (0.00 sec)

mysql> select @msg;
+-----+
| @msg  |
+-----+
| 借书成功 |
+-----+
1 row in set (0.00 sec)

mysql> call borrowBook('hzy', '9787544784382', @msg);
Query OK, 0 rows affected (0.00 sec)

mysql> select @msg;
+-----+
| @msg  |
+-----+
| 一个人不能同时借两本同样的书 |
+-----+
1 row in set (0.00 sec)

mysql> call returnBook('hzy', '9787544784207', @msg);
Query OK, 0 rows affected (0.00 sec)

mysql> select @msg;
+-----+
| @msg  |
+-----+
| 借阅记录不存在 |
+-----+

```

```
1 row in set (0.00 sec)
```

```
mysql> call returnBook('hzy', '9787544784382', @msg);
```

```
Query OK, 1 row affected (0.00 sec)
```

```
mysql> select @msg;
```

```
+-----+
```

```
| @msg |
```

```
+-----+
```

```
| 还书成功 |
```

```
+-----+
```

```
1 row in set (0.00 sec)
```

```
mysql> call borrowBook('hzy', '9787544784207', @msg);
```

```
Query OK, 1 row affected (0.00 sec)
```

```
mysql> call borrowBook('hzy', '9787544785600', @msg);
```

```
Query OK, 1 row affected (0.00 sec)
```

```
mysql> call borrowBook('hzy', '9787544784382', @msg);
```

```
Query OK, 1 row affected (0.01 sec)
```

```
mysql> call getBorrowRecords();
```

```
+-----+-----+-----+
```

```
| 用户名 | 书名 | 到期时间 |
```

```
+-----+-----+-----+
```

```
| hzy | 零K | 2021-06-15 16:22:57 |
```

```
| hzy | 局外人 西绪福斯神话 | 2021-06-15 16:25:33 |
```

```
| hzy | 抑郁症 | 2021-06-15 16:23:22 |
```

```
+-----+-----+-----+
```

```
3 rows in set (0.00 sec)
```

```
Query OK, 0 rows affected (0.00 sec)
```

```
mysql> insert into 借阅记录 values ('hzy', '9787540790356', '海东五百年', '2020-1-1',  
date_add(str_to_date('2020-1-1', '%Y-%m-%d'), interval 30 day), null);
```

```
Query OK, 1 row affected (0.00 sec)
```

```
mysql> call getOverdueRecords();
```

```
+-----+-----+-----+
```

```
| 用户名 | 书名 | 超期天数 |
```

```
+-----+-----+-----+
```

```
| hzy | 海东五百年 | 471 |
```

```
+-----+-----+-----+
```

```
1 row in set (0.01 sec)
```

```
Query OK, 0 rows affected (0.01 sec)
```