



哈爾濱工業大學
HARBIN INSTITUTE OF TECHNOLOGY

2020 年秋季学期
计算机学院大三
计算机系网络安全课程

Lab 3 实验报告

数据库用户的权限管理设计与实现

姓名	余涛
学号	1180300829
班号	1803202
电子邮件	1063695334@qq.com
手机号码	15586430583

1. 实验目的

熟练掌握数据库（比如 MySQL）基本权限管理命令、SQL 语言以及学习数据库系统的设计、数据库用户权限管理的实现，包括特定场景下数据表创建和管理、数据库用户的创建和合理的权限分配，权限分配细化到数据库、表、列和行，或者视图。

学生自行设计应用场景（应用场景不要和实验指导书示例相同），为具体的应用需求建立数据库，比如产品销售、人口管理、医院、银行、股票、手机通信信息等的数据库。并为你所面对的应用需求进行用户分类和权限划分，要求权限设计合理，能够实现依据用户权限的分发、权限的收回，并能够成组的批量分发和收回权限。

系统要求：

- （1）系统自建操作界面，能够实现单条、批量权限的查询、分发和收回等操作。（利用数据库系统自带的界面进行演示，视为不合格）
- （2）测试权限设计的合理性。
- （3）管理操作日志。

2. 实验环境搭建

Windows10 操作系统，MySQL 关系数据库管理系统，web 开发中 python 的 Django 框架。

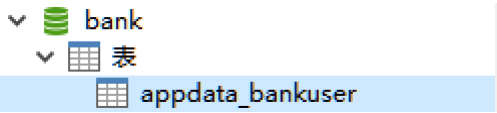
3. 实验步骤

3.1 一个简单的应用场景

一个虚拟银行管理系统，其包含一个表，客户-存款及基本信息表，包括用户名、用户密码、用户存款、用户类型。

3.2 建立数据库

创建 mysql 数据库 bank 如下：其包含一个表 bankuser:



表中包含以下内容：Id 号，用户名，密码，用户存款，用户类型：

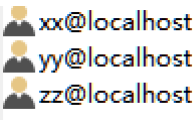
id	username	passwd	currency	isadmin
----	----------	--------	----------	---------

创建了一个管理员账户 manager:

其具有的权限如下，能够完成对所有人的权限的查看、授予以及撤销操作。Manager 自身也能够查看和更改各用户的用户名、id 以及密码，但不能更改用户的存款：

数据库	名	Select	Insert	Update	Reference Delete	Create	Drop	Alter	Index	Trigger	Create View	Show View	Grant	Execute
mysql		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
bank	appdata_bankuser	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
bank	appdata_bankuser.currency	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
bank	appdata_bankuser.id	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
bank	appdata_bankuser.isadmin	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
bank	appdata_bankuser.passwd	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
bank	appdata_bankuser.username	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

然后创建了三个用户 xx、yy、zz，他们的权限可以在实现的 web 客户端中由管理员 manager 进行授予和撤回：



3.3 系统设计实现功能

由于实现的是 web 客户端程序，所有需要登录管理员账号进行操作。

← → ↺

i 127.0.0.1:8002/bank/signin/

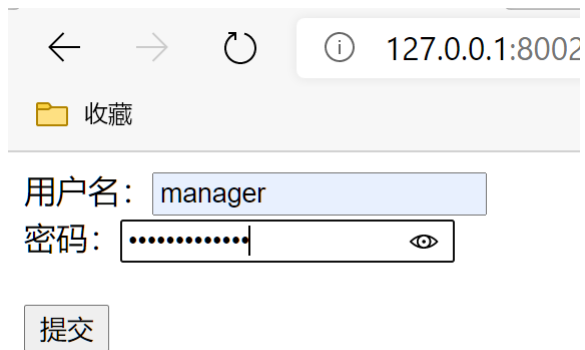
📁 收藏

用户名:

密码:

提交

登录管理员的账号：



← → ↻ ⓘ 127.0.0.1:8002

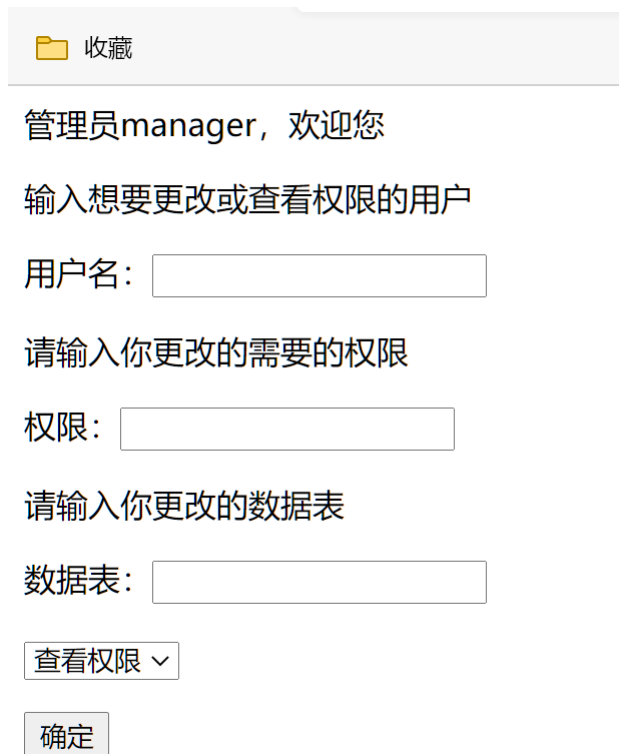
📁 收藏

用户名:

密码:

提交

登录后完成跳转，界面如下所示：



📁 收藏

管理员manager, 欢迎您

输入想要更改或查看权限的用户

用户名:

请输入你更改的需要的权限

权限:

请输入你更改的数据表

数据表:

查看权限 ▾

确定

首先查看管理员的权限，与设计相符（管理员不具有更改用户存款的权限）：

收藏

管理员manager, 欢迎您

输入想要更改或查看权限的用户

用户名:

请输入你更改的需要的权限

权限:

请输入你更改的数据表

数据表:

查看权限 ▾

确定

用户manager:

库(表): *.* 权限: USAGE

库(表): `mysql`.* 权限: SELECT, UPDATE

库(表): `bank`.`appdata_bankuser` 权限: SELECT, SELECT (passwd, username, currency, isadmin, id), UPDATE, UPDATE (passwd, username, isadmin, id), DELETE, CREATE, DROP, ALTER

然后可以单条查看用户的权限:

收藏

管理员manager, 欢迎您

输入想要更改或查看权限的用户

用户名:

请输入你更改的需要的权限

权限:

请输入你更改的数据表

数据表:

查看权限 ▾

确定

用户xx:

库(表): *.* 权限: USAGE

也可以批量查看用户的权限:

收藏

管理员manager，欢迎您

输入想要更改或查看权限的用户

用户名:

请输入你更改的需要的权限

权限:

请输入你更改的数据表

数据表:

▾

用户xx:
库(表): *.* 权限: USAGE
用户yy:
库(表): *.* 权限: USAGE
用户zz:
库(表): *.* 权限: USAGE

可以给单个用户授予单个权限:

管理员manager，欢迎您

输入想要更改或查看权限的用户

用户名:

请输入你更改的需要的权限

权限:

请输入你更改的数据表

数据表:

▾

```
grant select(passwd) on bank.appdata_bankuser to xx@'localhost';
```

此时查看所有用户权限变为:

收藏

管理员manager，欢迎您

输入想要更改或查看权限的用户

用户名:

请输入你更改的需要的权限

权限:

请输入你更改的数据表

数据表:

用户xx:
库(表): *.* 权限: USAGE
库(表): `bank`.`appdata_bankuser` 权限: SELECT (passwd)
用户yy:
库(表): *.* 权限: USAGE
用户zz:
库(表): *.* 权限: USAGE

可以对批量用户授予单个权限：

收藏

管理员manager，欢迎您

输入想要更改或查看权限的用户

用户名:

请输入你更改的需要的权限

权限:

请输入你更改的数据表

数据表:

```
grant select(passwd) on bank.appdata_bankuser to yy@'localhost';  
grant select(passwd) on bank.appdata_bankuser to zz@'localhost';
```

此时查看所有用户权限变为：

收藏

管理员manager, 欢迎您

输入想要更改或查看权限的用户

用户名:

请输入你更改的需要的权限

权限:

请输入你更改的数据表

数据表:

查看权限 ▾

确定

用户xx:
库(表): *.* 权限: USAGE
库(表): `bank`.`appdata_bankuser` 权限: SELECT (passwd)
用户yy:
库(表): *.* 权限: USAGE
库(表): `bank`.`appdata_bankuser` 权限: SELECT (passwd)
用户zz:
库(表): *.* 权限: USAGE
库(表): `bank`.`appdata_bankuser` 权限: SELECT (passwd)

也可以给单个用户撤回单个权限:

收藏

管理员manager, 欢迎您

输入想要更改或查看权限的用户

用户名:

请输入你更改的需要的权限

权限:

请输入你更改的数据表

数据表:

收回权限 ▾

确定

revoke select(passwd) on bank.appdata_bankuser from xx@'localhost';

此时查看所有用户权限变为:

收藏

管理员manager, 欢迎您

输入想要更改或查看权限的用户

用户名:

请输入你更改的需要的权限

权限:

请输入你更改的数据表

数据表:

用户xx:
库(表): *.* 权限: USAGE
用户yy:
库(表): *.* 权限: USAGE
库(表): `bank`.`appdata_bankuser` 权限: SELECT (passwd)
用户zz:
库(表): *.* 权限: USAGE
库(表): `bank`.`appdata_bankuser` 权限: SELECT (passwd)

可以对批量用户撤回单个权限:

收藏

管理员manager, 欢迎您

输入想要更改或查看权限的用户

用户名:

请输入你更改的需要的权限

权限:

请输入你更改的数据表

数据表:

```
revoke select(passwd) on bank.appdata_bankuser from yy@'localhost';
revoke select(passwd) on bank.appdata_bankuser from zz@'localhost';
```

此时查看所有用户权限变为:

收藏

管理员manager，欢迎您

输入想要更改或查看权限的用户

用户名:

请输入你更改的需要的权限

权限:

请输入你更改的数据表

数据表:

查看权限 ▾

确定

用户xx:
库(表): *.* 权限: USAGE
用户yy:
库(表): *.* 权限: USAGE
用户zz:
库(表): *.* 权限: USAGE

可以对单个用户授予多个权限:

收藏

管理员manager，欢迎您

输入想要更改或查看权限的用户

用户名:

请输入你更改的需要的权限

权限:

请输入你更改的数据表

数据表:

增加权限 ▾

确定

```
grant select(passwd) on bank.appdata_bankuser to xx@'localhost';  
grant update(passwd) on bank.appdata_bankuser to xx@'localhost';
```

此时查看所有用户权限变为:

收藏

管理员manager, 欢迎您

输入想要更改或查看权限的用户

用户名:

请输入你更改的需要的权限

权限:

请输入你更改的数据表

数据表:

查看权限 ▾

确定

用户xx:

库(表): *.* 权限: USAGE

库(表): `bank`.`appdata_bankuser` 权限: SELECT (passwd), UPDATE (passwd)

用户yy:

库(表): *.* 权限: USAGE

用户zz:

库(表): *.* 权限: USAGE

可以对单个用户撤回多个权限:

收藏

管理员manager, 欢迎您

输入想要更改或查看权限的用户

用户名:

请输入你更改的需要的权限

权限:

请输入你更改的数据表

数据表:

收回权限 ▾

确定

```
revoke select(passwd) on bank.appdata_bankuser from xx@'localhost';
revoke update(passwd) on bank.appdata_bankuser from xx@'localhost';
```

此时查看所有用户权限变为:

收藏

管理员manager, 欢迎您

输入想要更改或查看权限的用户

用户名:

请输入你更改的需要的权限

权限:

请输入你更改的数据表

数据表:

▾

用户xx:
库(表): *.* 权限: USAGE
用户yy:
库(表): *.* 权限: USAGE
用户zz:
库(表): *.* 权限: USAGE

可以对多个用户授予多个权限:

收藏

管理员manager, 欢迎您

输入想要更改或查看权限的用户

用户名:

请输入你更改的需要的权限

权限:

请输入你更改的数据表

数据表:

▾

```
grant select(passwd) on bank.appdata_bankuser to xx@'localhost';
grant update(passwd) on bank.appdata_bankuser to xx@'localhost';
grant select(passwd) on bank.appdata_bankuser to yy@'localhost';
grant update(passwd) on bank.appdata_bankuser to yy@'localhost';
```

此时查看所有用户权限变为:

收藏

管理员manager, 欢迎您

输入想要更改或查看权限的用户

用户名:

请输入你更改的需要的权限

权限:

请输入你更改的数据表

数据表:

查看权限 ▾

确定

用户xx:

库(表): *.* 权限: USAGE

库(表): `bank`.`appdata_bankuser` 权限: SELECT (passwd), UPDATE (passwd)

用户yy:

库(表): *.* 权限: USAGE

库(表): `bank`.`appdata_bankuser` 权限: SELECT (passwd), UPDATE (passwd)

用户zz:

库(表): *.* 权限: USAGE

可以对多个用户撤回多个权限:

收藏

管理员manager, 欢迎您

输入想要更改或查看权限的用户

用户名:

请输入你更改的需要的权限

权限:

请输入你更改的数据表

数据表:

收回权限 ▾

确定

```
revoke select(passwd) on bank.appdata_bankuser from xx@'localhost';
revoke update(passwd) on bank.appdata_bankuser from xx@'localhost';
revoke select(passwd) on bank.appdata_bankuser from yy@'localhost';
revoke update(passwd) on bank.appdata_bankuser from yy@'localhost';
```

此时查看所有用户权限变为:

收藏

管理员manager, 欢迎您

输入想要更改或查看权限的用户

用户名:

请输入你更改的需要的权限

权限:

请输入你更改的数据表

数据表:

查看权限 ▼

确定

用户xx:
库(表): *.* 权限: USAGE
用户yy:
库(表): *.* 权限: USAGE
用户zz:
库(表): *.* 权限: USAGE

这样就完成了单条、批量权限的查询、分发和收回。

3.4 权限设计的合理性

要实现权限的细粒度划分，用户应该能够具有对用户名、密码的查看和更新权限，而对存款、用户类型的查看权限，因此最终分配权限如下：

管理员manager, 欢迎您

输入想要更改或查看权限的用户

用户名:

请输入你更改的需要的权限

权限:

请输入你更改的数据表

数据表:

用户xx:

库(表): *.* 权限: USAGE

库(表): `bank`.`appdata_bankuser` 权限: SELECT (passwd, username, isadmin, currency),
UPDATE (passwd, username)

用户yy:

库(表): *.* 权限: USAGE

库(表): `bank`.`appdata_bankuser` 权限: SELECT (passwd, username, isadmin, currency),
UPDATE (passwd, username)

用户zz:

库(表): *.* 权限: USAGE

库(表): `bank`.`appdata_bankuser` 权限: SELECT (passwd, username, isadmin, currency),
UPDATE (passwd, username)

3.5 管理操作日志

将所有对数据库使用的命令操作记录在文件 `journal.txt` 中, 如下所示:

```
show grants for zz@'localhost';          Mon Dec 21 23:40:26 2020
show grants for xx@'localhost';          Mon Dec 21 23:44:27 2020
show grants for yy@'localhost';          Mon Dec 21 23:44:27 2020
show grants for zz@'localhost';          Mon Dec 21 23:44:27 2020
revoke select(passwd) on bank.appdata_bankuser from xx@'localhost';      Mon Dec 21 23:44:54 2020
revoke select(passwd) on bank.appdata_bankuser from yy@'localhost';      Mon Dec 21 23:44:54 2020
revoke select(passwd) on bank.appdata_bankuser from zz@'localhost';      Mon Dec 21 23:44:54 2020
show grants for xx@'localhost';          Mon Dec 21 23:44:57 2020
show grants for yy@'localhost';          Mon Dec 21 23:44:57 2020
show grants for zz@'localhost';          Mon Dec 21 23:44:57 2020
show grants for xx@'localhost';          Tue Dec 22 00:02:31 2020
show grants for yy@'localhost';          Tue Dec 22 00:02:31 2020
show grants for zz@'localhost';          Tue Dec 22 00:02:31 2020
grant update(passwd) on bank.appdata_bankuser to xx@'localhost';          Tue Dec 22 00:03:13 2020
grant update(passwd) on bank.appdata_bankuser to yy@'localhost';          Tue Dec 22 00:03:13 2020
grant update(passwd) on bank.appdata_bankuser to zz@'localhost';          Tue Dec 22 00:03:13 2020
show grants for manager@'localhost';      Tue Dec 22 00:03:20 2020
show grants for xx@'localhost';          Tue Dec 22 00:25:07 2020
show grants for yy@'localhost';          Tue Dec 22 00:25:07 2020
show grants for zz@'localhost';          Tue Dec 22 00:25:07 2020
grant select(passwd) on bank.appdata_bankuser to xx@'localhost';          Tue Dec 22 00:26:20 2020
```

3.6 部分源代码

查看权限源代码:

```
1. if operate == "check":
2.     conn = pymysql.connect(
3.         host='localhost',
4.         port=3306,
5.         user="manager",
6.         password="yutao19981119",
7.         db='bank',
8.         charset='utf8'
9.     )
10.    whom = []
11.    if (username != ""): # 分隔开读取的用户字符串
12.        if (',' in username):
13.            whom = username.split(',')
14.        else:
15.            whom.append(username)
16.    temp1 = ""
17.    for w in whom:
18.        cursor = conn.cursor() # 得到数据库的一个游标对象
19.        sql = "show grants for " + w + "@'localhost'" + ';'
20.        print(sql, "\n") # 打印信息
21.        cursor.execute(sql) # 数据库执行该 sql 语句
22.        # print(cursor)
23.        priv = []
24.        for i in cursor:
25.            priv.append(tuple(re.split(r' TO ', str(*i))[0].split(r' ON ')))
26.
27.        print(''.center(80, '~'))
28.        # print("用户", username)
29.        temp = "用户" + w + ":\n"
30.        for j in priv:
31.            privs = j[0].replace('GRANT', '')
32.            privs_info = j[1]
33.            temp = temp + '{0} {1:<20} {2} {3}'.format('库
(表):', privs_info, '权限:', privs) + "\n"
34.            print('{0} {1:<20} {2} {3}'.format('库(表):', privs_info, '权
限:', privs))
35.        print(''.center(80, '~'))
36.        print('\n')
```



```

37.         localtime = time.asctime(time.localtime(time.time()))
38.         journal = open('journal.txt', 'a') # 写日志
39.         journal.write(sql + "          " + localtime + "\n") # 写入日志
40.         conn.commit() # 提交
41.         cursor.close()
42.         temp1 = temp1 + temp
43.         # request.session['info'] = temp1
44.         # return redirect("../admin")
45.         return render(request, "appdata/admin.html", {'username': "manager", 'data': temp1})

```

授予权限源代码:

```

1. if operate == "give":
2.     conn = pymysql.connect(
3.         host='localhost',
4.         port=3306,
5.         user="manager",
6.         password="yutao19981119",
7.         db='bank',
8.         charset='utf8'
9.     )
10.    whom = []
11.    allright = []
12.    if (username != ""): # 分隔开读取的用户字符串
13.        if (',' in username):
14.            whom = username.split(',')
15.        else:
16.            whom.append(username)
17.    if (right!= ""): # 分隔开读取的用户字符串
18.        if (',' in right):
19.            allright = right.split(',')
20.        else:
21.            allright.append(right)
22.    temp = ""
23.    for w in whom:
24.        for r in allright:
25.            cursor = conn.cursor() # 得到数据库的一个游标对象
26.            sql = 'grant ' + r + ' on bank.' + table + ' to ' + w + "
                @'localhost'" + ';' # 为每个用户赋予权限
27.            print(sql, "\n") # 打印信息
28.            cursor.execute(sql) # 数据库执行该 sql 语句
29.            journal = open('journal.txt', 'a') # 写日志
30.            localtime = time.asctime(time.localtime(time.time()))

```

```

31.             journal.write(sql + "          " + localtime + "\n") # 写入
    日志
32.             conn.commit() # 提交
33.             cursor.close()
34.             temp = temp + sql + "\n"
35.             # request.session['info'] = temp
36.             # return redirect("../admin")
37.             return render(request, "appdata/admin.html", {'username': "manage
r", 'data': temp})

```

收回权限源代码:

```

1. if operate == "withdraw":
2.     conn = pymysql.connect(
3.         host='localhost',
4.         port=3306,
5.         user="manager",
6.         password="yutao19981119",
7.         db='bank',
8.         charset='utf8'
9.     )
10.    whom = []
11.    allright = []
12.    if (username != ""): # 分隔开读取的用户字符串
13.        if (',' in username):
14.            whom = username.split(',')
15.        else:
16.            whom.append(username)
17.    if (right != ""): # 分隔开读取的用户字符串
18.        if (',' in right):
19.            allright = right.split(',')
20.        else:
21.            allright.append(right)
22.    temp = ""
23.    for w in whom:
24.        for r in allright:
25.            cursor = conn.cursor() # 得到数据库的一个游标对象
26.            sql = 'revoke ' + r + ' on bank.' + table + ' from ' + w + "@'lo
calhost'" + ';'
27.            print(sql, "\n") # 打印信息
28.            cursor.execute(sql) # 数据库执行该 sql 语句
29.            journal = open('journal.txt', 'a') # 写日志
30.            localtime = time.asctime(time.localtime(time.time()))
31.            journal.write(sql + "          " + localtime + "\n") # 写入日志

```

```
32.         conn.commit() # 提交
33.         cursor.close()
34.         temp = temp + sql + "\n"
35.     # request.session['info'] = temp
36.     # return redirect("../admin")
37.     return render(request, "appdata/admin.html", {'username': "manager", 'data': temp})
38. rn render(request, "appdata/admin.html", {'username': "manager", 'data': ""}
    )
```