# DB2019

# 第五次作业

# 1712884

# 樊漆亮

## 源码：

#!/usr/bin/python3  
# -\*-:coding:utf-8-\*-  
  
import pymysql  
  
db = pymysql.**connect**("127.0.0.1", "test", "fanqiliang123456", "exercise")  
  
cur = db.**cursor**()  
cur.**execute**("drop function if exists fun;")  
s = '''  
create function *fun*(c\_id varchar(8)) returns integer   
 deterministic reads sql data   
begin   
 declare i integer default 0;   
   
 if *exists*(select \* from prereq t where t.course\_id=c\_id) then   
 insert into result   
 select distinct t0.prereq\_id   
 from prereq t0  
 where t0.course\_id=c\_id ;  
 end if;  
 return i;   
end  
'''.**replace**('\n', ' ')  
  
cur.**execute**("create temporary table if not exists result(id varchar(8) );")  
cur.**execute**(s)  
  
  
def exec\_fun(str\_id):  
 cur.**execute**("select fun(%s);", (str\_id,))  
  
  
def query(str\_id):  
 cur.**execute**("select prereq\_id from prereq where course\_id=%s;", (str\_id,))  
 data = cur.**fetchall**()  
 return data  
  
  
def getresult(str\_id):  
 if str\_id is not None:  
 **exec\_fun**(str\_id)  
 data = **query**(str\_id)  
 for d in data:  
 **getresult**(d[0])  
  
  
s = input()  
**getresult**(s)  
cur.**execute**("select \* from result;")  
data = cur.**fetchall**()  
print(data)  
# for d in data:  
# print(d[0])  
cur.**execute**("drop temporary table if exists result;")  
cur.**execute**("drop function if exists fun;")  
cur.**close**()

## 所获所知：

1. 起初我尝试在sql中直接进行函数递归调用，发现mysql并不支持直接函数递归。
2. 随后我尝试在函数中返回一个表，发现mysql的函数也不支持。
3. 于是我便转向在python中进行函数递归
4. 完成作业后与同学交流时发现，mysql中repeat…end repeat语句和create temporary table语句的使用，可以形成一种自定义栈的模拟函数递归方式，使我有更深刻的印象。
5. python与mysql一同写，mysql client与mysql api函数接口在语法方面的细微区别使我前期写出大量bug，但在解决bug的时候，我对mysql语法、python语法都有更为深入的了解，受益良多。

## 3 感想：

这次的实验有一定难度，但是也令我收获良多，我希望老师以后能布置更难的作业，这样我相信进步会更大！