Python基础 第十六天作业

1. 获取路由器running config并且备份到数据

代码：

import sqlite3  
#  
conn = sqlite3.connect('configuredb.sqlite')  
cursor = conn.cursor()  
cursor.execute('create table config(ip varchar(40),config varchar(99999),md5 config varchar(999))')  
  
from ssh\_router import ssh\_cli  
import hashlib  
import re  
import sqlite3  
  
device\_list = ['1.1.1.200']  
username = 'cisco'  
password = 'cisco'  
  
def get\_config\_md5(ip,username,password):  
 try:  
 device\_config = ssh\_cli(ip, username, password)  
 ret = re.split(r'\nhostname \S+\n', device\_config)  
 run\_conf = re.findall('hostname.\*',str(ret))  
 m = hashlib.md5()  
 m.update(str(run\_conf).encode())  
 md5\_value = m.hexdigest()  
 return run\_conf,md5\_value  
  
 except Exception:  
 return  
  
def write\_config\_md5\_to\_db():  
  
 conn = sqlite3.connect('configuredb.sqlite')  
 cursor = conn.cursor()  
 # cursor.execute('select \* from config')  
 # ret = cursor.fetchall()  
 # for i in ret:  
 # print(i)  
 for device in device\_list:  
 conn = sqlite3.connect('configuredb.sqlite')  
 cursor = conn.cursor()  
 config\_and\_md5 = get\_config\_md5(device,username,password)  
 config\_info = config\_and\_md5[0][0]  
 md5\_info = config\_and\_md5[1]  
 cursor.execute('select \* from config')  
 md5\_result = cursor.fetchall()  
 if not md5\_result:  
 cursor.execute(f'insert into config(ip, config, md5) values("{device}","{config\_info}","{md5\_info}")')  
 cursor.fetchall()  
 conn.commit()  
 else:  
 cursor.execute(f'select md5 from config where ip = "{device}"')  
 md5 = cursor.fetchall()  
 md5 = md5[0][0]  
 if md5 != md5\_info:  
 t = f"update config set config=?,md5=? where ip = ?"  
 cursor.execute(t,(config\_info,md5\_info,device))  
  
 else:  
 continue  
 cursor.execute('select \* from config')  
 all\_result = cursor.fetchall()  
 for x in all\_result:  
 print(x[0],x[2])  
 conn.commit()  
  
  
  
if \_\_name\_\_ == '\_\_main\_\_':  
  
 # ret =get\_config\_md5('1.1.1.200',username,password)  
 write\_config\_md5\_to\_db()

运行结果：

