

# 哈工大软件学院数据库实验1

原创

seen\_in\_hw 于 2015-09-10 00:30:31 发布 1361 收藏 3

分类专栏: [databases](#) 文章标签: [数据库](#) [软件](#) [哈工大数据库](#) [sqlite](#) [代码](#)

版权声明: 本文为博主原创文章, 遵循 [CC 4.0 BY-SA](#) 版权协议, 转载请附上原文出处链接和本声明。

本文链接: [https://blog.csdn.net/qq\\_23100787/article/details/48328829](https://blog.csdn.net/qq_23100787/article/details/48328829)

版权



[databases](#) 专栏收录该内容

9 篇文章 0 订阅

订阅专栏

## 前言

这个实验很简单, 我先讲下大致步骤, 然后再解释详细点, 再附上实验解答

## 实验内容

这次实验就是说白了, 把外部的txt文件通过sqlite等软件导入到一个数据库里面去, 这个数据库就是db3后缀的那个文件然后不是导入进去了吗? 导入进去之后就对数据库里面的数据进行一系列的操作, 这就达到了实验的目的

## 实验难点

- .初学者不会使用软件, 不知道该怎么操作
- .不理解相关信息含义, 无法写出有质量的代码

相关难点剖析: (利用sqlite)

创建数据库用 .open seen.db3 //seen.db3是自己改的

创建表用 create table(...这里面输入各种键值, 用来接数据);

.separator “,” 这个里面的逗号是定义txt文件里面的那个分隔符, 利用这个分隔符读取文件

关于导入导出的技巧:

- 1.txt文件千万不能有多余的空格
- 2.导入时txt文件里面的分割符要和导入的相一致
- 3.import语句不要分号
- 4.导入的时候是会重复导入的, 前面导入的后面还会有
- 5.字符串存储相当严格, 连空格都会存储进去

## 实验解答

我看了几年的实验, 似乎都是那50道题, 认真学习的同学会发现这些都在PPT上能够找到, 如果不认真的就上网找吧, 但愿你们能够找到这里来, 给我博客增添点人气~~~

所以我就直接附上那50道题的代码了, 每个都是经过我测试好的。如果其中有问题就找我来问吧, 我肯定会帮助你解决的

```
11,select sID,sName,GPA from Student where GPA>3.6;
```

```
12,select sID,sName from Student where GPA>3.6;
```

```
13,select sName,cName from Student,Apply where Student.sID = Apply.sID;
```

```
14,select distinct sName,cName from Student,Apply where Student.sID = Apply.sID;  
(从上条我们就可以看出, 去除重复元组是有多强)
```

```
15,select sName,GPA,decision from Student,Apply where Student.sID=Apply.sID and  
cName='Stanford' and major='CS' and sizeHS<1000;
```

```
16,select distinct College.cName from College,Apply where  
College.cName=Apply.cName and major='CS' and enrollment>20000;
```

```
17,select sName,College.cName,GPA,state,enrollment from Apply,Student,College  
where Apply.sID = Student.sID and Apply.cName = College.cName order by GPA desc;
```

在满足上述条件下想让次要条件按enrollment降序就可以在后面加一句enrollment desc;

18,参照17附加

```
19,select sID,major from Apply where major like '%bio%';
```

```
20,select * from College,Student;
```

21, ...略

```
22,select s1.sName,s2.sName from Student s1,Student s2 where s1.sID!=s2.sID and  
s1.GPA=s2.GPA; (去除重复元素似乎不管用) -----有问题
```

22, 23 同

```
24, select cName from College union all select sName from Student;(含重复)
```

不含重复版本:

```
select distinct cName from College union all select distinct sName from  
Student;
```

```
25,select distinct cName as name from College union all select distinct sName as  
name from Student order by name;
```

26,25已经完成

```
27, select sID from Apply where major=='CS' intersect select sID from Apply where  
major = 'EE';
```

```
28,select sID from Apply where major=='CS' except select sID from Apply where  
major = 'EE';
```

```
29,select sName,sID from Student where sID in (select sID from Apply where  
major='CS');
```

```

30,select distinct sName from Student,Apply where Student.sID= Apply.sID and
major='CS';

或者: select distinct sName from Student where sID in(select sID from Apply where
major='CS');

31,select distinct sID from Apply where sID in (select sID from Apply where major
= 'CS') and sID not in (select sID from Apply where major = 'EE');

32,select cName from College s1 where not exists (select * from College s2 where
s1.enrollment<s2.enrollment);

33,select distinct sName,GPA from Student s1 where not exists (select * from
Student s2 where s1.GPA<s2.GPA);

33,select distinct sName,GPA from Student s1 where not exists (select * from
Student s2 where s1.GPA<s2.GPA) ;

34,..

35,select sName major from Student,Apply where Student.sID=Apply.sID;

36, select sName,GPA from Student natural join Apply where sizeHS<1000 and
cName='Stanford' and major='CS';

select sName,GPA from Student join Apply using (sID) where sizeHS<1000 and
cName='Stanford' and major='CS';

select sName,GPA from Student join Apply where Student.sID= Apply.sID and
sizeHS<1000 and cName='Stanford' and major='CS';

37,select * from Student natural join Apply natural join College;

38,

39,select avg(GPA) from Student ;

40,select min(GPA) from Student natural join Apply where major='cs'; //聚集查询
// select distinct s1.GPA from Student s1 natural join Apply where not exists
(select * from Student s2 where s1.GPA>s2.GPA ) and major='CS';

41, select avg(GPA) from Student natural join Apply where major='CS';

42,select count(*) from College where enrollment>15000;

43,select count(*) from Apply where cName='Cornell' and sID in (select distinct
sID from Apply); //这个问题,牛逼牛逼

```

```
44, select CS.avgGPA-NonCS.avgGPA from (select avg(GPA) as avgGPA from Student
natural join Apply where major='CS') as CS,(select avg(GPA) as avgGPA from Student where
sID not in(select sID from Apply where major='CS')) as NonCS;

45,select cName,count(*) from Apply group by cName;

46,select state,sum(enrollment) from College group by state;

47,select cName,major,max(GPA),min(GPA) from Student natural join Apply group by
cName,major; //group by 可以循环使用

48,select sID,sName,count(*) from Apply natural join Student group by Apply.sID;

50,select cName from Apply group by cName having count(*)<5;
```