数据库课程实验报告-SQL

基本信息
姓名：米成刚
专业：计算机应用技术
学号：201028002829040
年级：一年级
院系：计算与通信工程学院
E-mail：michenggang@gmail.com

1. 实验目的

（1）学会在 SQL Server 2000 中定义表及数据库模式
（2）分析 SQL Server 2000 实现关系模型的吻合度
（3）分析一些特殊的操作，如集合操作、除、嵌套等
（4）学会使用 SQL 语言实现查询

2. 实验准备

　　Microsoft SQL Server 2000

3. 实验内容和步骤

（1）SQL Server 2000 的安装与配置

　　将企业版安装光盘插入光驱后，出现以下提示框。请选择 "安装 SQL Server 2000 组件"，出现下一个页面后，选择 "安装数据库服务器"。

选择 "下一步"，然后选择 "本地计算机" 进行安装。

　　在 "安装选择" 窗口，选择 "创建新的 SQL Server 实例..."。对于初次安装的用户，应选用这一安装模式，不需要使用 "高级选项" 进行安装。"高级选项" 中的内容均可在安装完成后进行调整。

　　在 "用户信息" 窗口，输入用户信息，并接受软件许可证协议。

　　在 "安装定义"窗口，选择 "服务器和客户端工具" 选项进行安装。我们需要将服务器和客户端同时安装，这样在同一台机器上，我们可以完成相关的所有操作，对于我们学习 SQL Server 很有用处。如果你已经在其它机器上安装了 SQL Server，则可以只安装客户端工具，用于对其它机器上 SQL Server 的存取。

　　在 "实例名" 窗口，选择 "默认" 的实例名称。这时本 SQL Server 的名称将和 Windows 2000 服务器的名称相同。例如笔者的 Windows 服务器名称是 "Darkroad"，则 SQL Server 的名字也是 "Darkroad"。SQL Server 2000 可以在同一台服务器上安装多个实例，也就是你可以重复安装几次。这时您就需要选择不同的实例名称了。建议将实例名限制在 10 个字符之内。实例名会出现在各种 SQL Server 和系统工具的用户界面中，因此，名称越短越容易读取。另外，实例名称不能是 "Default" 或 "MSSQLServer" 以及 SQL Server 的保留关键字等。

　　在 "安装类型" 窗口，选择 "典型" 安装选项，并指定 "目的文件夹"。程序和数据文件的默认安装位置都是 "C:\Program Files\Microsoft SQL Server\"。笔者因为 C 盘是系

统区、D 盘是应用区，因此选择了 D 盘。注意，如果您的数据库数据有 10 万条以上的话，请预留至少 1G 的存储空间，以应付需求庞大的日志空间和索引空间。

在"服务账号"窗口，请选择"对每个服务使用统一账户…"的选项。在"服务设置"处，选择"使用本地系统账户"。如果需要"使用域用户账户"的话，请将该用户添加至 Windows Server 的本机管理员组中。

在"身份验证模式"窗口，请选择"混合模式…"选项，并设置管理员"sa"账号的密码。如果您的目的只是为了学习的话，可以将该密码设置为空，以方便登录。如果是真正的应用系统，则千万需要设置和保管好该密码！：）如果需要更高的安全性，则可以选择"Windows 身份验证模式"，这时就只有 Windows Server 的本地用户和域用户才能使用 SQL Server 了。
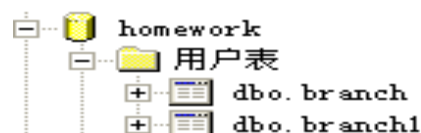
在"选择许可模式"窗口，根据您购买的类型和数量输入（0 表示没有数量限制）。"每客户"表示同一时间最多允许的连接数，"处理器许可证"表示该服务器最多能安装多少个 CPU。
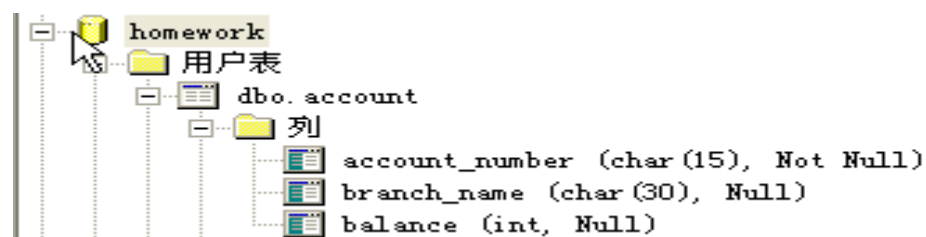
（2）SQL 操作

创建基本表
1> 直接创建表
```
create table branch
(branch_name  char(15) not null,
 branch_city  char(30),
 assets       integer)
```
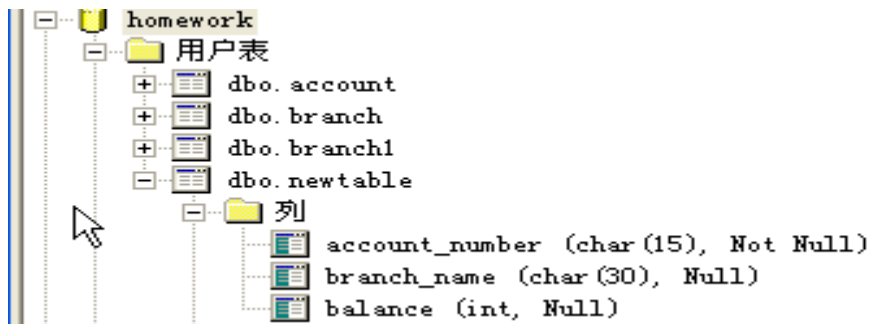

2> 创建包括约束条件的表
```
create table account
(account_number char(15) not null,
 branch_name    char(30),
 balance        integer,
 primary key (account_number),
 check (balance >= 0))
```


3> 根据已有的表创建表
```
select * into newtable
from account
```

删除基本表

1> 删除表中的所有元组

    delete from newtable
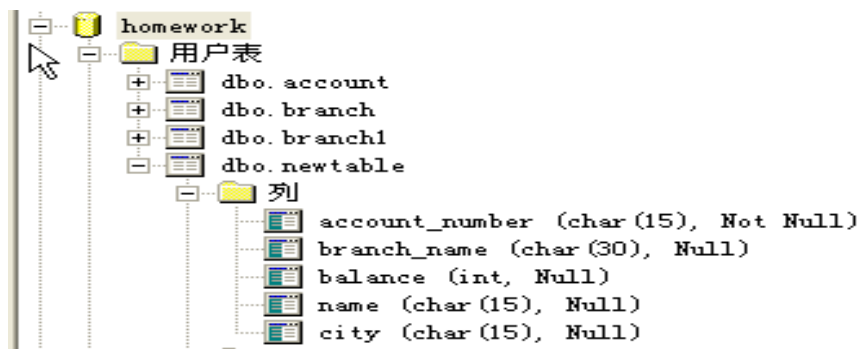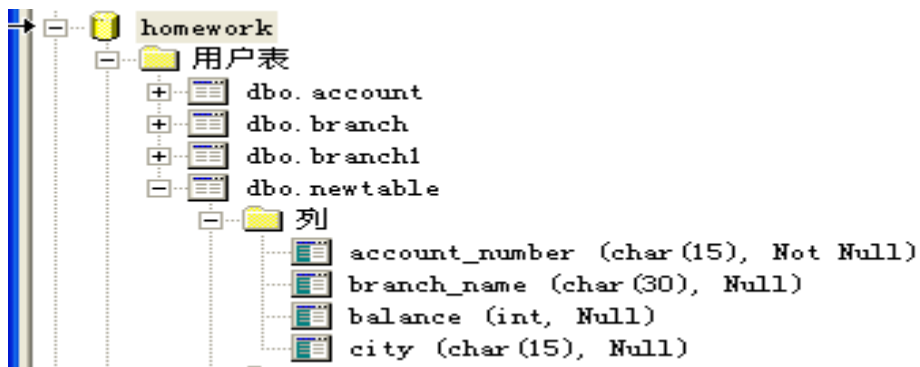
2> 彻底删除表

    drop table newtable

基本表属性修改

1> 修改属性

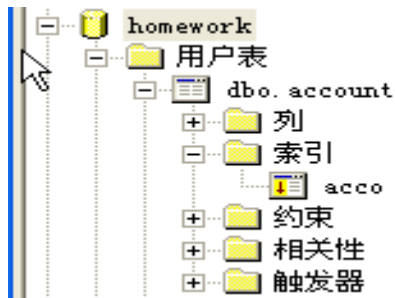alter table newtable add name char(15),city char(15)

2> 删除属性

alter table newtable drop column name

索引操作

1> 创建索引

create index acco on account (account_number)

2> 索引删除

```
drop index account.acco
```

基本表增加、删除、修改

1> 插入一个元组

```
Insert into account values('A-101','Downtown',400)
```


2> 插入一个查询结果

```
Insert into account
      select loan_number,branch_name,200
      from bank.dbo.loan
      where branch_name='downtown'
```

3> 删除账号为L-14和L-17的元组

```
delete from account
where account_number='L-14' or account_number='L-17'
```

4> 更新

```
Update account
Set balance=balance*1.05
```


基本表查询

/*DISTINCT消除重复的值*/

```
select distinct branch_name
from account
```

/*all显示地表示不要去除重复*/

```
select all branch_name
```

```
from account
```

	branch_name
1	Downtown
2	Train
3	Shop

批查询完成。　OD4EA6F4957A4FC (8.0)　OD4EA6F4957A4FC\Administra　homework　0:00:00　3 行　行 3，列 1

/*符号*表示选择所有属性 */

```
select *
from account
```

	account_number	branch_name	balance
1	A-101	Downtown	420
2	B-123	Train	134
3	B-201	Shop	123

批查询完成。　OD4EA6F4957A4FC (8.0)　OD4EA6F4957A4FC\Administra　homework　0:00:00　3 行　行 3，列 1

/*选择指定属性，并含算术表达式*/

```
select loan_number, branch_name, amount*100 from loan
```

	loan_number	branch_name	（无列名）
1	1	gucas	123400
2	2	cas	2345600

批查询完成。　OD4EA6F4957A4FC (8.0)　OD4EA6F4957A4FC\Administra　homework　0:00:00　2 行　行 1，列 54

/*找出所有在Perryridge支行贷款并且贷款超过1200美元的贷款的贷款号*/

```
select loan_number
from loan
where branch_name ='Perryridge' and amount > 1200
```

	loan_number
1	3
2	4

批查询完成　OD4EA6F4957A4FC (8.0)　OD4EA6F4957A4FC\Administra　homework　0:00:00　2 行　行 4，列 1

/*贷款额在900到1300的贷款号*/

```
select loan_number from loan where amount between 900 and 1300
```

/*找出所有从银行贷款用户的名字、贷款号、贷款额*/

```
select customer_name, T.loan_number, S.amount
from borrower as T, loan as S
where  T.loan_number = S.loan_number
```


/*字符串操作：找出街道地址中含有子串'Main'的所有客户的姓名*/

```
select customer_name
from customer
where customer_street like '%Main%'
```


排序

/*排序：按字母顺序列出在Perryridge支行中有贷款的客户*/

```
select distinct customer_name
from borrower, loan
where borrower.loan_number = loan.loan_number and
      branch_name = 'Perryridge'
order by customer_name
```

批查询完成。　OD4EA6F4957A4FC (8.0)　OD4EA6F4957A4FC\Administra homework　0:00:00　1 行　行 6，列 1

/*排序：升序*/

select *

from account

order by balance ASC

	account_number	branch_name	balance
1	B-201	Shop	123
2	B-123	Train	134
3	A-101	Downtown	420

/*排序：降序*/

select *

from account

order by balance DESC

	account_number	branch_name	balance
1	A-101	Downtown	420
2	B-123	Train	134
3	B-201	Shop	123

集合

集合并运算：找出在银行有帐户、贷款或两者都有的帐户

(select customer_name from depositor)

union

(select customer_name from borrower)

	customer_name
1	lj
2	mcg

批查询完成。　OD4EA6F4957A4FC (8.0)　OD4EA6F4957A4FC\Administra homework　0:00:00　2 行　行 4，列 1

/*集合交运算：找出在银行同时有帐户和贷款的帐户，教材用intersect，T-SQL用select实现
*/

SELECT DISTINCT customer_name

FROM depositor

WHERE EXISTS

7

```
(SELECT *
FROM borrower
WHERE depositor.customer_name = borrower.customer_name)
```

	customer_name
1	lj
2	mcg

网格 消息

批查询完成。　　　　OD4EA6F4957A4FC (8.0)　OD4EA6F4957A4FC\Administra　homework　0:00:00　2 行　行 7, 列 1

/*集合差运算：找出在银行有帐户但无贷款的帐户，教材用 except，T-SQL 用 select 实现*/
```
select distinct customer_name
from depositor
where not EXISTS
    (select *
    from borrower
    where depositor.customer_name = borrower.customer_name)
```

customer_name

网格 消息

批查询完成。　　　　OD4EA6F4957A4FC (8.0)　OD4EA6F4957A4FC\Administra　homework　0:00:00　0 行　行 7, 列 1

聚集查询

/*聚集函数：找出 Perryridge 支行的帐户余额平均值*/
```
select avg (balance) as avg_bananlce
from account
where branch_name = 'Perryridge'
```

	avg_bananlce
1	111

网格 消息

批查询完成。　　　　OD4EA6F4957A4FC (8.0)　OD4EA6F4957A4FC\Administra　homework　0:00:00　1 行　行 4, 列 1

/*聚集函数：统计 customer 中的元组个数*/
```
select count (*) as customer_count
from customer
```

8

网格	消息

批查询完成。 | OD4EA6F4957A4FC (8.0) | OD4EA6F4957A4FC\Administra | homework | 0:00:00 | 1 行 | 行 3，列 1

/*聚集函数：统计depositor中的元组个数，重复的名字按一个统计*/

select count (distinct customer_name) as customer_distinct_count

from depositor

	customer_distinct_count
1	2

网格	消息

批查询完成。 | OD4EA6F4957A4FC (8.0) | OD4EA6F4957A4FC\Administra | homework | 0:00:00 | 1 行 | 行 3，列 1

/*聚集函数：分组统计每个支行的账户余额平均值*/

select branch_name, avg (balance) as avg_balance

from account

group by branch_name

	branch_name	avg_balance
1	Downtown	420
2	Perryridge	111
3	Shop	123
4	Train	134

网格	消息

批查询完成。 | OD4EA6F4957A4FC (8.0) | OD4EA6F4957A4FC\Administra | homework | 0:00:00 | 4 行 | 行 4，列 1

/*空值测试：检查loan中属性amount是否存在空值，is not null用于检测非空值*/

select loan_number

from loan

where amount is null

	loan_number

网格	消息

批查询完成。 | OD4EA6F4957A4FC (8.0) | OD4EA6F4957A4FC\Administra | homework | 0:00:00 | 0 行 | 行 4，列 1

复杂查询

/*复杂查询：找出那些平均账户余额大于500美元的支行的平均账户余额*/
select branch_name, avg (balance)
        from account
        group by branch_name

	branch_name	(无列名)
	Downtown	420
2	Perryridge	111
3	Shop	123
4	Train	134

批查询完成。 0D4EA6F4957A4FC (8.0) 0D4EA6F4957A4FC\Administra homework 0:00:00 4 行 行 4，列 1

select branch_name, avg_balance
from (select branch_name, avg (balance)
        from account,depositor
        group by branch_name)
as result (branch_name, avg_balance)
where avg_balance > 500

	branch_name	avg_balance
1	Perryridge	678

批查询完成。 0D4EA6F4957A4FC (8.0) 0D4EA6F4957A4FC\Administra homework 0:00:00 1 行 行 7，列 1

/*
    Find all customers who have an account at all branches located in Brooklyn city.
    找出找出所有位于Brooklyn的所有支行都有帐户的客户。已知：关系branch、depositor、
    account
*/
select distinct S.customer_name from depositor as S
where not exists(
  (select branch.branch_name from branch
   where branch.branch_city ='Brooklyn' and branch.branch_name
   not in
     (select R.branch_name from depositor as T, account as R
      Where T.account_number = R.account_number and S.customer_name =
     T.customer_name)))

	customer_name
1	lj
2	mcg

批查询完成。 0D4EA6F4957A4FC (8.0) 0D4EA6F4957A4FC\Administra homework 0:00:00 2 行 行 8，列 1

连接操作

/*基本表连接操作：广义笛卡儿连接*/

SELECT *

FROM loan

	loan_number	branch_name	amount
1	1	gucas	1234
2	2	cas	23456
3	3	Perryridge	1234
4	4	Perryridge	5789
5	5	cas	678

批查询完成。 0D4EA6F4957A4FC (8.0) 0D4EA6F4957A4FC\Administra homework 0:00:00 5 行 行 3，列 1

SELECT *

FROM borrower

	ID	customer_name	loan_number
1	1	mcg	2
2	2	lj	4

批查询完成。 0D4EA6F4957A4FC (8.0) 0D4EA6F4957A4FC\Administra homework 0:00:00 2 行 行 3，列 1

SELECT loan.branch_name,borrower.customer_name,loan.loan_number,loan.amount

FROM loan,borrower/*基本表连接操作：广义笛卡儿连接，与上式等价*/

SELECT a.branch_name,p.customer_name,a.loan_number,a.amount

FROM loan AS a cross JOIN borrower AS p

	branch_name	customer_name	loan_number	amount
1	gucas	mcg	1	1234
2	cas	mcg	2	23456
3	Perryridge	mcg	3	1234
4	Perryridge	mcg	4	5789
5	cas	mcg	5	678
6	gucas	lj	1	1234
7	cas	lj	2	23456

批查询完成。 0D4EA6F4957A4FC (8.0) 0D4EA6F4957A4FC\Administra homework 0:00:00 10 行 行 3，列 1

/*基本表连接操作：自然连接：与教材loan inner join borrower on
　　　loan.loan_number=borrower.loan_number等价*/

SELECT a.branch_name, p.customer_name, a.loan_number, a.amount

FROM loan AS a INNER JOIN borrower AS p

ON a.loan_number = p.loan_number

	branch_name	customer_name	loan_number	amount
1	cas	mcg	2	23456
2	Perryridge	lj	4	5789

批查询完成。　　OD4EA6F4957A4FC (8.0)　OD4EA6F4957A4FC\Administra　homework　0:00:00　2 行　行 4，列 1

/*基本表连接操作：左外连接*/

SELECT loan.branch_name, borrower.customer_name, loan.loan_number, loan.amount

FROM loan LEFT OUTER JOIN borrower

ON borrower.loan_number = loan.loan_number

	branch_name	customer_name	loan_number	amount
1	gucas	NULL	1	1234
2	cas	mcg	2	23456
3	Perryridge	NULL	3	1234
4	Perryridge	lj	4	5789
5	cas	NULL	5	678

批查询完成。　　OD4EA6F4957A4FC (8.0)　OD4EA6F4957A4FC\Administra　homework　0:00:00　5 行　行 4，列 1

/*基本表连接操作：右外连接*/

SELECT loan.branch_name, borrower.customer_name, loan.loan_number, loan.amount

FROM loan RIGHT OUTER JOIN borrower

ON borrower.loan_number = loan.loan_number

	branch_name	customer_name	loan_number	amount
1	cas	mcg	2	23456
2	Perryridge	lj	4	5789

批查询完成。　　OD4EA6F4957A4FC (8.0)　OD4EA6F4957A4FC\Administra　homework　0:00:00　2 行　行 4，列 1

/*基本表连接操作：右外连接*/

SELECT loan.branch_name, loan.loan_number, loan.amount, branch.branch_name

FROM loan RIGHT OUTER JOIN branch

ON branch.branch_name = loan.branch_name

	branch_name	loan_number	amount	branch_name	
1	NULL	NULL	NULL	Downtown	
2	NULL	NULL	NULL	Train	
3	NULL	NULL	NULL	Shop	

网格 消息

批查询完成。　　0D4EA6F4957A4FC (8.0)　0D4EA6F4957A4FC\Administra　homework　0:00:00　3 行　行 4，列 1

SELECT loan.branch_name, loan.loan_number, loan.amount, branch.branch_name
FROM branch RIGHT OUTER JOIN loan
ON branch.branch_name = loan.branch_name

	branch_name	loan_number	amount	branch_name
1	gucas	1	1234	NULL
2	cas	2	23456	NULL
3	Perryridge	3	1234	NULL
4	Perryridge	4	5789	NULL
5	cas	5	678	NULL

网格 消息

批查询完成。　　0D4EA6F4957A4FC (8.0)　0D4EA6F4957A4FC\Administra　homework　0:00:00　5 行　行 4，列 1

/*基本表连接操作：全连接*/

SELECT branch.branch_name, branch.branch_city, branch.assets, account.balance
FROM branch FULL OUTER JOIN account
ON branch.branch_name=account.branch_name

	branch_name	branch_city		assets	balance	
1	Downtown	A		2	420	
2	Train	B		3	134	
3	Shop	C		4	123	
4	NULL	NULL		NULL	678	

网格 消息

批查询完成。　　0D4EA6F4957A4FC (8.0)　0D4EA6F4957A4FC\Administra　homework　0:00:00　4 行　行 4，列 1

SELECT branch.branch_name, branch.branch_city, branch.assets, account.balance
FROM branch left OUTER JOIN account
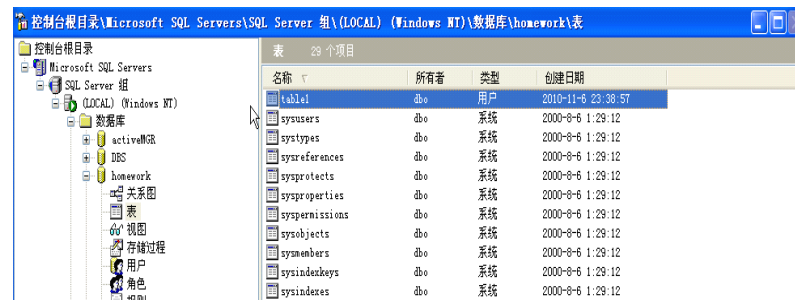ON branch.branch_name=account.branch_name

	branch_name	branch_city		assets	balance	
1	Downtown	A		2	420	
2	Train	B		3	134	
3	Shop	C		4	123	

网格 消息

批查询完成。　　0D4EA6F4957A4FC (8.0)　0D4EA6F4957A4FC\Administra　homework　0:00:00　3 行　行 4，列 1

13

/*基本表连接操作：生成新表*/

```
SELECT a.branch_name,p.customer_name,a.loan_number,a.amount into table1
FROM loan AS a INNER JOIN borrower AS p
ON a.loan_number = p.loan_number
```


视图操作

/*视图创建：根据两个基本表*/

```
create view customer1 as
  (select branch_name, customer_name
   from depositor, account
   where depositor.account_number = account.account_number)
```

/*视图创建：利用查询表创建*/

```
create view all_customer as
  (select branch_name, customer_name
   from depositor, account
   where depositor.account_number = account.account_number)
 union
  (select branch_name, customer_name
   from borrower, loan
   where borrower.loan_number = loan.loan_number)
```

/*创建视图并变更属性*/

```
CREATE VIEW dbo.loan_view(支行名,资产) AS
SELECT branch_name, amount
FROM dbo.loan
```

/*视图查询：与基本表一致*/

```
select * from all_customer
```
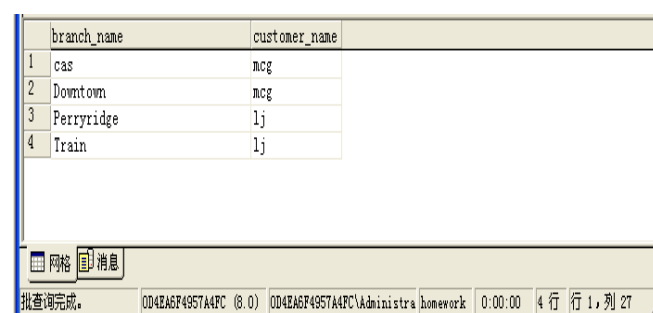
存储过程

```
/*建立存储过程*/
CREATE PROCEDURE [dbo].[getUserId] (@uId int)
AS
select * from stu_user where stu_id=@uId
/*执行存储过程*/
EXEC getUserId 1
```

4. 实验结论

（1）实验结论

A、通过本次实验，对 SQL 语句有了更加深入地理解。通过建立索引、视图、存储过程，也对数据库的结构有了全局的认识。

B、用 SQL 语句进行数据库操作时，对操作对象的逻辑要有全局的认识，这样才能在总体上对数据库有一个把握。

（2）问题分析

建立表之间的关系图时，要先对各个表的属性有较深刻地理解，尤其是主键、外键。另外，存在联系的两个表的属性值的数据类型必须一致，这样，在确定各个表之间关系时才不会出现错误。