**Sql语句学习**

1. **基本操作**
2. **创建数据库**

**操作代码：**

USE master

IF EXISTS(SELECT NAME FROM SYSDATABASES WHERE name='Demo')

DROP DATABASE 'Demo'

GO

CREATE DATABASE 'Demo'

ON PRIMARY

(

NAME='Demo\_data',

FILENAME='E:\谯谯\课程学习\大二下\数据库\数据库建立\Demo\_data.mdf',

SIZE=10MB,

FILEGROWTH = 5MB

)

LOG ON

(

NAME='Info\_log.ldf',

FILENAME='E:\谯谯\课程学习\大二下\数据库\数据库建立\Info\_log.ldf',

SIZE=5MB,

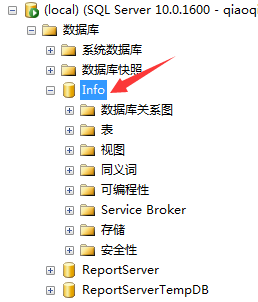
MAXSIZE=1000MB,

FILEGROWTH=10%

)

GO

**截图：**



1. **创建表 (注：由于外键约束必须得按照此顺序建表)**

**1、Department表：**

**操作代码：**

USE Info

IF EXISTS(SELECT NAME FROM SYSOBJECTS WHERE NAME='Department')

DROP TABLE Department

GO

CREATE TABLE Department

(

Dp\_Id INT IDENTITY(1,1) PRIMARY KEY, --加入IDENTITY(1,1)类型后该字段不能自行添加，系统自动添值

Dp\_Name VARCHAR(10) NOT NULL,

--Dp\_Name VARCHAR(10) CONSTRAINT Unique\_Name UNIQUE NONCLUSTERED(Dp\_Name),

--加上唯一约束后会按照该字段排序

)

INSERT INTO Department(Dp\_Name) VALUES ('地科系')

INSERT INTO Department(Dp\_Name) VALUES ('工程系')

INSERT INTO Department(Dp\_Name) VALUES ('仪器系')

INSERT INTO Department(Dp\_Name) VALUES ('经管系')

INSERT INTO Department(Dp\_Name) VALUES ('人文社科系')

INSERT INTO Department(Dp\_Name) VALUES ('外语系')

SELECT \* FROM Department

GO **截图：**



**2、Director表：**

**操作代码：**

USE Info

IF EXISTS(SELECT NAME FROM SYSOBJECTS WHERE NAME='Director')

DROP TABLE Director

GO

CREATE TABLE Director

(

Dr\_Icard NCHAR(18) PRIMARY KEY,

Dr\_Name VARCHAR(6) NOT NULL,

Dr\_Tel NVARCHAR(11) NULL,

Dr\_Sex CHAR(2) CONSTRAINT Dr\_Sex CHECK (Dr\_Sex in ('男','女')),

Dr\_Age INT NULL,

Dp\_Id INT CONSTRAINT FK\_DrDp FOREIGN KEY (Dp\_Id) REFERENCES Department(Dp\_Id)

)

INSERT INTO Director SELECT \* FROM Hwork.DBO.Director

SELECT \* FROM Director

GO

**截图：**



**3、Student表：**

**操作代码：**

USE Info

IF EXISTS(SELECT NAME FROM SYSOBJECTS WHERE NAME='Student')

DROP TABLE Student

GO

CREATE TABLE Student

(

S\_No CHAR(9) PRIMARY KEY,

S\_Name VARCHAR(6) NOT NULL,

S\_Birth DATETIME NULL,

Dp\_Id INT CONSTRAINT FK\_StDp FOREIGN KEY (Dp\_Id) REFERENCES Department (Dp\_Id),

Dr\_Icard NCHAR(18) CONSTRAINT FK\_StDr FOREIGN KEY (Dr\_Icard) REFERENCES Director (Dr\_Icard)

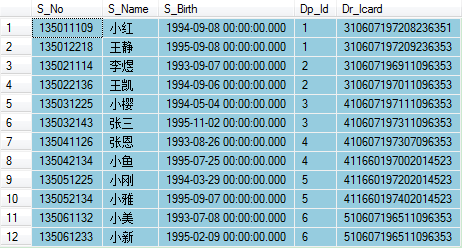
)

INSERT INTO Student SELECT \* FROM Hwork.dbo.Student --复制数据

SELECT \* FROM Student

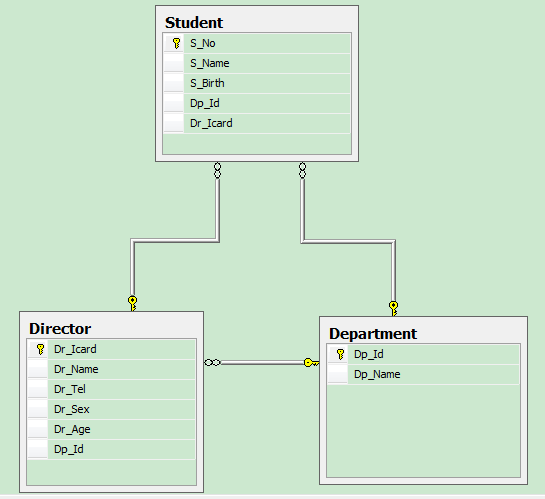
GO

**截图：**



1. **建立数据关系图**

**截图：**

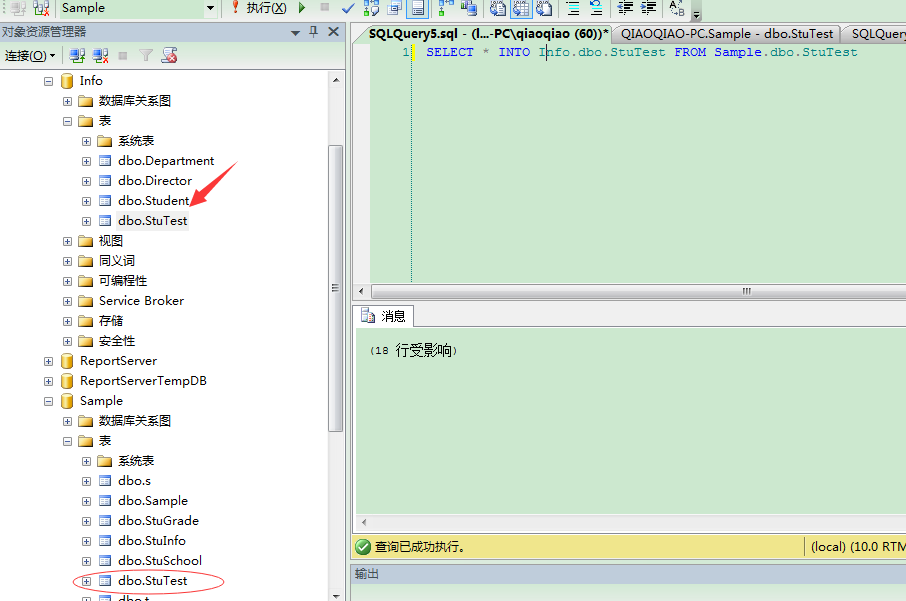


1. **从一个库的表复制到另一个库**

**操作代码：**

SELECT \* INTO Info.dbo.StuTest FROM Sample.dbo.StuTest

**截图：**



1. **在Info库中复制student到Test1并删除Total列**

**操作代码：**

USE Info

GO

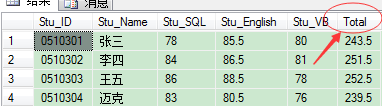
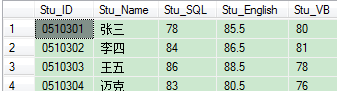
SELECT \* INTO Test1 FROM StuTest

ALTER TABLE Test1

DROP COLUMN Total --若不加入column，则默认删除约束

GO

**截图：**

1. **将表student的Total列更新到Test1的Total中**

**操作代码：**

USE Info

GO

UPDATE Test1

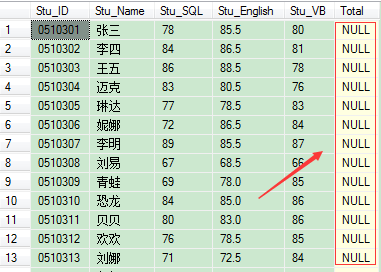
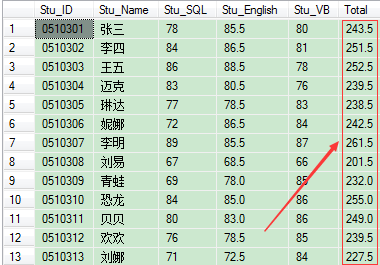
SET Total=StuTest.Total

FROM StuTest

WHERE Test1.Stu\_ID=StuTest.Stu\_ID

GO

**截图：**

1. **修改表Test1的Total字段的数据类型**

**操作代码：**

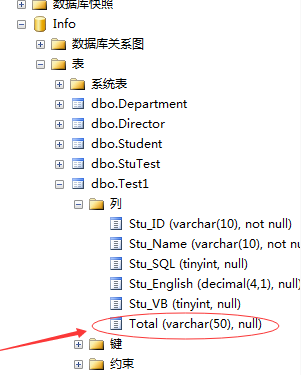
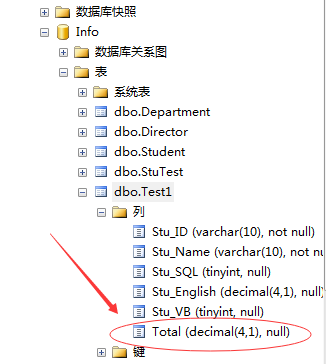
USE Info

GO

ALTER TABLE Test1

ALTER column Total DECIMAL(4,1)

GO

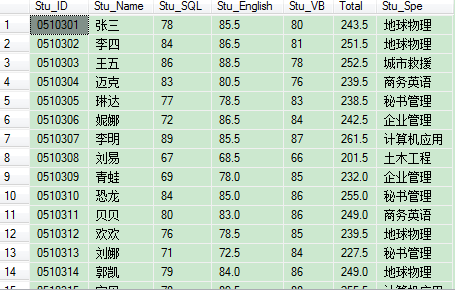
 

1. **常用操作**
2. **SELECT查询**
3. **查询若干列**

**操作代码：**

USE Info

SELECT \* FROM Test1

**截图：**

1. **消除重复的行**

**操作代码：**

USE Info

SELECT DISTINCT Stu\_Spe FROM Test1

**截图：**

1. **查询计算值（出生年）**

**操作代码：**

USE Info

GO

SELECT S\_No,S\_Name,

(CONVERT(varchar(20),YEAR(GETDATE())-YEAR(S\_Birth))+'岁')

AS '年龄' FROM Student

GO

**截图：**



1. **设置生日提醒**

**操作代码：**

USE Info

GO

DECLARE @a INT ,@b INT,@c VARCHAR(10),@d VARCHAR(100)

SET @b=1

SET @d=''

SELECT @a=count(\*) from Student

--print @a

WHILE(@b<=@a)

BEGIN

SET @c=(SELECT S\_Name FROM Student WHERE MONTH(GETDATE())=MONTH(S\_Birth)AND S\_Id=@b )

IF (LEN(@c)<>0)

SET @d=@d+@c+''

IF (@b=@a)

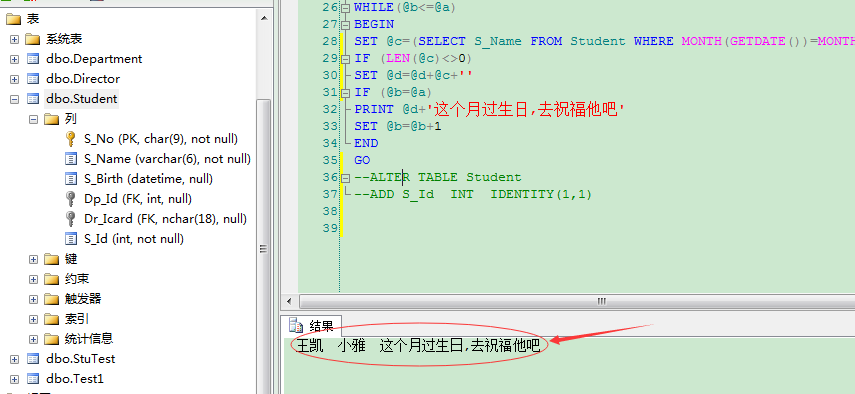
PRINT @d+'这个月过生日,去祝福他吧'

SET @b=@b+1

END

GO

**截图：**



1. **条件查询**

**(1)查询年龄大于20岁的学生学号，姓名，出生年份，年龄**

**操作代码：**

USE Info

GO

SELECT S\_No,S\_Name,YEAR(S\_Birth) AS '出生年份',

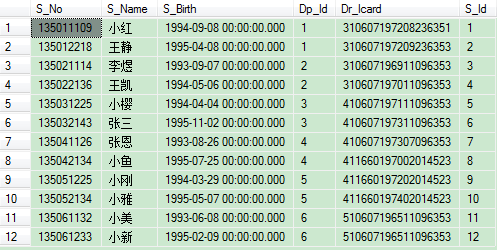
YEAR(GETDATE())-YEAR(S\_Birth) AS '年龄'

FROM Student

WHERE (YEAR(GETDATE())-YEAR(S\_Birth))>20

GO

**截图：**





**(2)查询分数在240到250之间的学生学号及其姓名**

**操作代码：**

USE Info

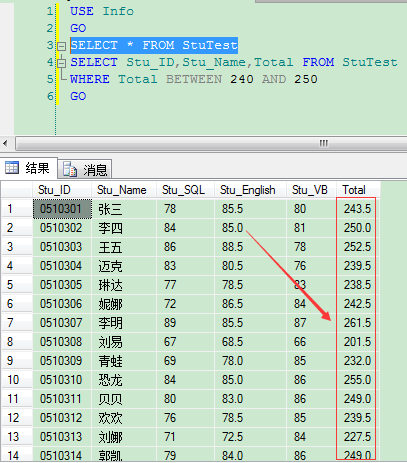
GO

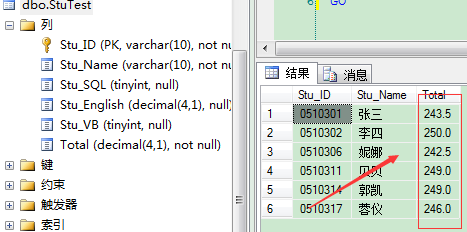
SELECT Stu\_ID,Stu\_Name,Total FROM StuTest

WHERE Total BETWEEN 240 AND 250

GO

**截图：**





**(3)匹配查询（like语句运用）**

**操作代码：**

USE Info

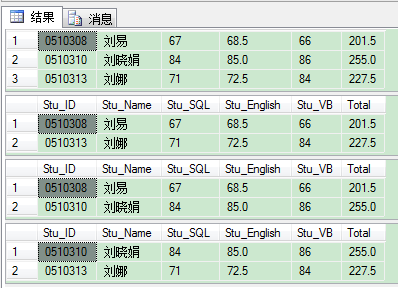
SELECT \* FROM StuTest WHERE Stu\_Name LIKE '刘%'

SELECT \* FROM StuTest WHERE Stu\_Name LIKE '刘\_'

SELECT \* FROM StuTest WHERE Stu\_Name LIKE '刘[易,晓]%'

SELECT \* FROM StuTest WHERE Stu\_Name LIKE '刘[^易]%'

**截图：**



**(4) SELECT嵌套查询**

**操作代码：**

USE Info

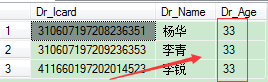
GO

SELECT Dr\_Icard,Dr\_Name,Dr\_Age FROM Director

WHERE Dr\_Age IN(SELECT Dr\_Age FROM Director WHERE Dr\_Name='杨华' )

GO

**截图：**



**(5) ORDER BY查询结果排序（将学生按照总成绩降序）**

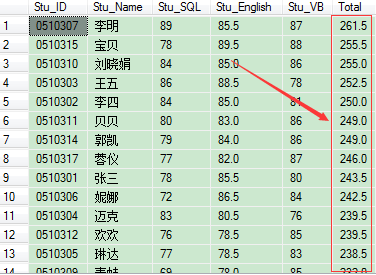
**操作代码：**

USE Info

SELECT \* FROM StuTest

ORDER BY Total DESC --默认ASC排序

**截图：**



**(6)GROUP BY语句的使用（求各个系的平均成绩及其总分）**

**操作代码：**

USE Info

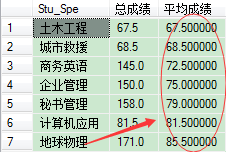
SELECT Stu\_Spe,SUM(Stu\_Test)AS '总成绩',AVG(Stu\_Test)AS '平均成绩'

FROM Test

GROUP BY Stu\_Spe

ORDER BY '平均成绩'

**截图：**



**(7) COMPUTE 与COMPUTE BY语句的使用（compute by必须与order by连用）**

**操作代码：(求Stu\_Test总成绩，Stu\_PS的平均分)**

USE Info

SELECT Stu\_Test,Stu\_PS,Stu\_Spe FROM Test

COMPUTE SUM(Stu\_Test),AVG(Stu\_PS)

**截图：**



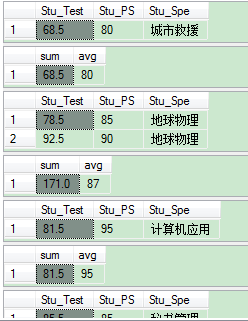
**操作代码：(求Stu\_Test总成绩，Stu\_PS的平均分并按照Stu\_Spe分类)**

USE Info

SELECT Stu\_Test,Stu\_PS,Stu\_Spe FROM Test

ORDER BY Stu\_Spe

COMPUTE SUM(Stu\_Test),AVG(Stu\_PS)BY Stu\_Spe



**(7) 多表查询**

**操作代码：**

USE Info

SELECT S\_No,S\_Name,Dr\_Name,Dp.Dp\_Id,Dp\_Name

FROM Student AS St,Director AS Dr,Department AS Dp

WHERE St.Dr\_Icard=Dr.Dr\_Icard AND Dr.Dp\_Id=Dp.Dp\_Id

ORDER BY Dp\_Id

**截图：**



**(7) UNION子句(一般应用于不同班级间的成绩排名)**

**操作代码：**

USE Info

--SELECT \* INTO T1 FROM Test

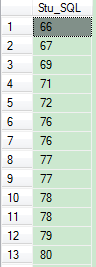
SELECT Stu\_SQL FROM StuTest

UNION ALL

SELECT Stu\_SQL FROM T1

ORDER BY Stu\_SQL

**截图：**



1. **视图**
2. **创建视图（创建查询总成绩大于250的学生信息）**

**操作代码：**

USE Info

GO

CREATE VIEW Stu\_v

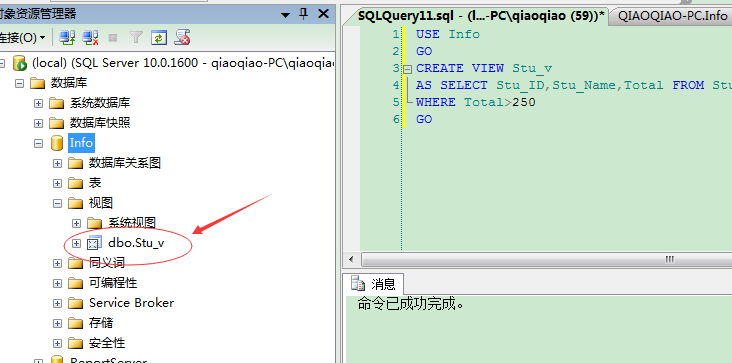
AS SELECT Stu\_ID,Stu\_Name,Total FROM StuTest

WHERE Total>250

GO

SELECT \* FROM Stu\_v

**截图：**





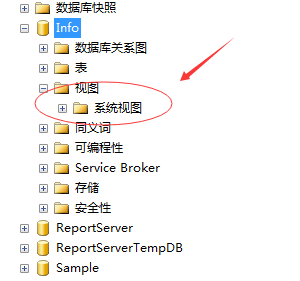
1. **撤销视图**

**操作代码：**

USE Info

DROP VIEW Stu\_v

**截图：**



1. **索引**
2. **创建索引**

**操作代码：**

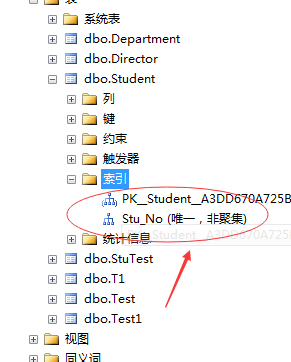
USE Info

GO

CREATE UNIQUE INDEX Stu\_No ON Student(S\_No)

GO

**截图：**



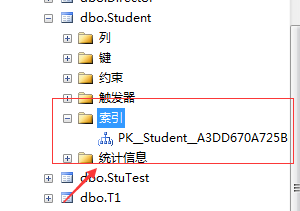
1. **删除索引**

**操作代码：**

USE Info

DROP INDEX Student.Stu\_No

**截图：**



1. **游标**

**（1）创建游标**

**操作代码：**

USE Sample

declare @a varchar(10),@b varchar(10)

declare cursor1 scroll cursor for --定义游标cursor6

select Stu\_ID,Stu\_Name from t

open cursor1

fetch last from cursor1 into @a,@b

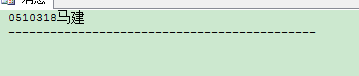
print @a+@b

print'--------------------------------------------'

close cursor1

deallocate cursor1

**截图：**



1. **存储过程**
2. **不带参的存储过程**

**操作代码：**

IF EXISTS(SELECT NAME FROM sysobjects WHERE name='od\_sp' AND TYPE='P')

DROP PROC od\_sp

GO

USE Sample

GO

Create PROC od\_sp --存储过程名

As

Select \* from t

Where Total>260--EXEC od\_sp

**截图：**



1. **带参的存储过程**

**操作代码：**

IF EXISTS (SELECT NAME FROM SYSOBJECTS WHERE name='od1\_sp' AND TYPE='P')

DROP PROC od1\_sp

GO

USE Sample

GO

CREATE PROC od1\_sp @a int

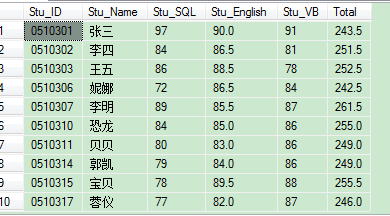
AS

SELECT \* FROM t

WHERE Total>@a

--EXEC od1\_sp 240

**截图：**



1. **带默认参的存储过程**

**操作代码：**

IF EXISTS(SELECT NAME FROM SYSOBJECTS WHERE name='od2\_sp' AND TYPE='P')

DROP PROC od2\_sp

GO

USE Sample

GO

CREATE PROC od2\_sp @a INT = NULL

AS

IF @a IS NULL

BEGIN

SELECT \* FROM t

WHERE Total>250

END

ELSE

BEGIN

SELECT \* FROM t

WHERE Total>@a

END

--EXEC od2\_sp 260

**截图：**



1. **带输出参数的存储过程**

**操作代码：**

if exists(select name from sysobjects where name='qq3\_sp' and type='p')

drop proc qq3\_sp

go

use Sample

go

create proc qq3\_sp @a varchar(10),@sum decimal output

as

select Stu\_Name from t

where Stu\_Name like @a+'%'

select @sum=SUM(total) from t

where Stu\_Name like @a+'%'

--declare @sum decimal

--exec qq3\_sp '李',@sum output

--print @sum

**截图：**

