数据库的查询和视图

一、SELECT语句的基本使用

1. 查询Employees表中所有数据

2. 查询Employees表中指定字段数据

3. 查询Employees表中的部门号和性别，要求使用Distinct消除重复行

4. 使用WHERE子句查询表中指定的数据

查询编号为’000001’的雇员的地址和电话

查询月收入高于2000元的员工号码

查询1970年以后出生的员工的姓名和住址

5. 使用AS子句为表中字段指定别名

查询Employees表中女雇员的地址和电话，并将列标题显示为地址和电话

查询Employees表中男雇员的姓名和出生日期，并将列标题显示为姓名和出生日期

6. 使用使用CASE子句

查询Employees表中员工的姓名和性别，要求Sex值为1时显示“男”，为0时显示“女”

查询Employees表中员工的姓名、住址和收入水平，2000元以下显示为低收入，2000~3000地显示为中等收入，3000元以上显示为高收入。

7. 使用SELECT语句进行简单计算

计算每个雇员的实际收入

8. 使用内置函数

获得员工总数

计算Salary表中员工月收入的平均数

获得Employees表中最大的员工号码

计算Salary表中所有员工的总支出

查询财务部雇员的最高和最低实际收入

9. 模糊查询

找出所有姓王的雇员的部门号

找出所有地址中含有“中山”的雇员的号码及部门号

找出员工号码中倒数第二个数字为0的员工的姓名、地址和学历

10. Between…And…和Or的使用

找出收入在2000~3000元之间的雇员编号

找出部门为“1”或“2”的雇员的编号

11. 使用INTO子句，由源表创建新表

由表Salary创建“SalaryNew”表，要求包括编号和收入，选择收入在1500元以上的雇员

由表Employees创建“EmployeesNew”表，要求包括编号和姓名，选择所有男员工

二、子查询的使用

1. 查找在财务部工作的雇员情况

2. 用子查询的方法查找所有收入在2500以下的雇员的情况

3. 查找财务部年龄不低于研发部雇员年龄的雇员姓名

4. 用子查询的方法查找研发部比所有财务部雇员收入都高的雇员的姓名

5. 查找比所有财务部的雇员收入都高的雇员的姓名

6. 用子查询的方法查找所有年龄比研发部雇员年龄都大的雇员的姓名

三、连接查询的使用

1. 查询每个雇员的情况及薪水的情况

2. 查询每个雇员的情况及其工作部门的情况

3. 使用内连接的方法查询名字为“王林”的雇员所在的部门

4. 使用内连接的方法查找出不在财务部工作的所有雇员信息

5. 使用外连接方法查找出所有员工的月收入

6. 查找财务部收入在2000元以上的雇员姓名及其薪水详情

7. 查询研发部在1976年以前出生的雇员姓名及其薪水详请

四、聚合函数的使用

1. 求财务部雇员的平均收入

2. 查询财务部雇员的最高和最低收入

3. 求财务部雇员的平均实际收入

4. 查询财务部雇员的最高和最低实际收入

5. 求财务部雇员的总人数

6. 统计财务部收入在2500元以上的雇员人数

五、GROUP BY、ORDER BY子句的使用

1. 查找Employees表中男性和女性的人数

2. 按部门列出在该部门工作的员工的人数

3. 按员工的学历分组，排列出本科、大专、硕士的人数

4. 查找员工数超过2的部门名称和雇员数量

5. 按员工的工作年份分组，统计各个工作年份的人数，例如工作1年的多少人，工作2年的多少人

6. 将雇员的情况按收入由低到高排列

7. 将员工信息按出生时间从小到大排列

8. 在ORDER BY 子句中使用子查询，查询员工姓名、性别和工龄信息，要求按实际收入从大到小排列

六、视图的使用

1. 创建视图

（1）在数据库YGGL上创建视图Departments\_View，视图包含Department表的全部列

（2）创建视图Employees\_Departments\_View，视图包含员工号码、姓名、所在部门名称

（3）创建视图Employees\_Salary\_View，视图包含员工号码、姓名和实际收入三列

2. 查询视图

从视图Employees\_Salary\_View中查询出姓名为“王林”的员工的实际收入

3. 更新视图

（1）向视图Departments\_View中添加一条记录（‘6’，‘广告部’，‘广告业务’）

执行完命令后，分别查看Departments\_View和Department表中发生的变化

（2）尝试向Employees\_Departments\_View中添加一条记录，看看会发生什么情况

（3）尝试向Employees\_Salary\_View中添加一条记录，看看会发生什么情况

（4）将视图Departments\_View中，部门号为‘6’的部门名称修改为‘生产车间’

（5）删除视图Departments\_View中最新增加的的一条记录

4. 删除视图 Employees\_Departments\_View

5. 在界面工具中操作视图

一、SELECT语句的基本使用

1. 查询Employees表中所有数据

SELECT \* FROM Employees;

2. 查询Employees表中指定字段数据

SELECT EmployeeID,Name,DepartmentID FROM Employees;

3. 查询Employees表中的部门号和性别，要求使用Distinct消除重复行

SELECT Distinct DepartmentID,Sex FROM Employees;

4. 使用WHERE子句查询表中指定的数据

查询编号为’000001’的雇员的地址和电话

Select Address,PhoneNumber FROM Employees WHERE EmployeeID='000001';

查询月收入高于2000元的员工号码

SELECT EmployeeID FROM Salary WHERE InCome>2000;

查询1970年以后出生的员工的姓名和住址

SELECT Name,Address FROM Employees WHERE YEAR(Birthday)>'1970';

SELECT Name,Address FROM Employees WHERE Birthday>'1970';

5. 使用AS子句为表中字段指定别名

查询Employees表中女雇员的地址和电话，并将列标题显示为地址和电话

SELECT Address AS 地址,PhoneNumber AS 电话 FROM Employees;

查询Employees表中男雇员的姓名和出生日期，并将列标题显示为姓名和出生日期

SELECT Name AS 姓名,Birthday AS 出生日期 FROM Employees WHERE Sex=1;

6. 使用使用CASE子句

查询Employees表中员工的姓名和性别，要求Sex值为1时显示“男”，为0时显示“女”

SELECT Name AS 姓名,

CASE

WHEN Sex=1 THEN '男'

WHEN Sex=0 THEN '女'

END

AS 性别

FROM Employees;

查询Employees表中员工的姓名、住址和收入水平，2000元以下显示为低收入，2000~3000地显示为中等收入，3000元以上显示为高收入。

SELECT

a.Name AS 姓名,a.Address as 住址,

CASE

WHEN b.InCome<2000 THEN '低收入'

WHEN b.InCome BETWEEN 2000 AND 3000 THEN '中等收入'

WHEN b.InCome >3000 THEN '高收入'

END

AS 收入水平

FROM Employees a,Salary b

WHERE a.EmployeeID=b.EmployeeID;

7. 使用SELECT语句进行简单计算

在Salary表中，计算每个雇员的实际收入，并显示雇员编号和实际收入

SELECT EmployeeID AS 雇员编号,InCome-OutCome AS 实际收入 FROM Salary;

8. 使用聚合函数

获得员工总数

SELECT COUNT(\*) FROM Employees;

计算Salary表中员工月收入的平均数

SELECT AVG(InCome) AS 平均收入 FROM Salary;

获得Employees表中最大的员工号码

SELECT MAX(EmployeeID) AS 雇员编号 FROM Employees;

计算Salary表中所有员工的总支出

SELECT SUM(OutCome) AS 总支出 FROM Salary;

查询财务部雇员的最高和最低实际收入

SELECT MAX(InCome-OutCome) AS 最高收入,MIN(InCome-OutCome) AS 最低收入 FROM Salary;

9. 模糊查询

找出所有姓王的雇员的部门号

SELECT DepartmentID AS 部门号 FROM Employees WHERE Name like '王%';

找出所有地址中含有“中山”的雇员的编号及部门号

SELECT EmployeeID AS 雇员编号,DepartmentID AS 部门号 FROM Employees WHERE Address like '%中山%'

找出雇员编号中倒数第二个数字为0雇员的编号的姓名、地址和学历

SELECT EmployeeID AS 雇员编号,Name AS 姓名,Address AS 地址,Education AS 学历 FROM Employees WHERE SUBSTRING(EmployeeID,5,1)='0';

10. Between…And…和Or的使用

找出收入在2000~3000元之间的雇员编号

SELECT EmployeeID,InCome FROM Salary WHERE InCome BETWEEN 2000 AND 3000;

找出部门为“1”或“2”的雇员的编号

SELECT EmployeeID,DepartmentID FROM Employees WHERE DepartmentID='1' OR DepartmentID='2';

11. 使用INTO子句，由源表创建新表

由表Salary创建“SalaryNew”表，要求包括编号和收入，选择收入在1500元以上的雇员

SELECT EmployeeID,InCome

INTO SalaryNew

FROM Salary

WHERE InCome>1500;

由表Employees创建“EmployeesNew”表，要求包括编号和姓名，选择所有男员工

SELECT EmployeeID,Name

INTO EmployeesNew

FROM Employees

WHERE Sex=1;

二、子查询的使用

1. 查找在财务部工作的雇员情况

--分析:雇员表和部门表通过DepartmentID进行关联

--1)在Departments表中查找财务部的编号

--2)在Employees表中查找部门的雇员情况

SELECT \* FROM Employees WHERE DepartmentID =(

SELECT DepartmentID FROM Departments WHERE DepartmentName='财务部');

2. 用子查询的方法查找所有收入在2500以下的雇员的情况

--分析:员工表和收入表通过EmployeeID进行关联

SELECT \* FROM Employees WHERE EmployeeID IN (

SELECT EmployeeID FROM Salary WHERE InCome<2500);

3. 查找财务部年龄不低于研发部雇员年龄的雇员姓名

--分析:雇员表和部门表通过DepartmentID进行关联

--1)先找到研发部的部门编号

--2)再通过部门编号找到研发部的雇员年龄

--3)再找到财务部的部门编号,并通过部门编号查找雇员姓名,条件是年龄不低于所有研发部雇员年龄

SELECT Name FROM Employees WHERE DepartmentID in (

SELECT DepartmentID FROM Departments WHERE DepartmentName='财务部')

AND Birthday !> ALL(

SELECT Birthday FROM Employees WHERE DepartmentID in(

SELECT DepartmentID FROM Departments WHERE DepartmentName='研发部'));

4. 用子查询的方法查找研发部比所有财务部雇员收入都高的雇员的姓名

--1)查找财务部雇员的收入

--2)查找研发部雇员的编号,条件是收入比所有财务部雇员收入都高

--3)通过编号找到雇员姓名

SELECT Name FROM Employees WHERE EmployeeID IN (

SELECT EmployeeID FROM Salary WHERE EmployeeID IN(

SELECT EmployeeID FROM Employees WHERE DepartmentID =(

SELECT DepartmentID FROM Departments WHERE DepartmentName='研发部'))

AND InCome > ALL(

SELECT InCome FROM Salary WHERE EmployeeID IN(

SELECT EmployeeID FROM Employees WHERE DepartmentID =(

SELECT DepartmentID FROM Departments WHERE DepartmentName='财务部'))));

5. 查找比所有财务部的雇员收入都高的雇员的姓名

--分析:1)查找所有财务部的雇员收入

--2)查找其他雇员编号,条件是收入比所有财务部的雇员收入都高

--3)通过编号找到姓名

SELECT Name FROM Employees WHERE EmployeeID IN(

SELECT EmployeeID FROM Salary WHERE InCome>ALL(

SELECT InCome FROM Salary Where EmployeeID IN (

SELECT EmployeeID FROM Employees Where DepartmentID =(

SELECT DepartmentID FROM Departments WHERE DepartmentName='财务部'))));

6. 用子查询的方法查找所有年龄比研发部雇员年龄都大的雇员的姓名

--分析:年龄都大等价于生日都小

--1)找到所有研发部雇员的生日

--2)找到其他部门雇员的姓名,条件是生日比研发部的所有雇员的生日都小

SELECT Name FROM Employees WHERE Birthday < ALL (

SELECT Birthday FROM Employees WHERE DepartmentID IN (

SELECT DepartmentID FROM Departments WHERE DepartmentName = '研发部'));

三、连接查询的使用

1. 查询每个雇员的情况及薪水的情况

SELECT a.\*,b.\*

FROM Employees a,Salary b

WHERE a.EmployeeID=b.EmployeeID;

2. 查询每个雇员的情况及其工作部门的情况

SELECT a.\*,b.\*

FROM Employees a,Departments b

WHERE a.DepartmentID=b.DepartmentID;

3. 使用内连接的方法查询名字为“王林”的雇员所在的部门

SELECT b.DepartmentName FROM Departments b

INNER JOIN Employees a

ON a.DepartmentID=b.DepartmentID

Where a.Name='王林';

4. 使用内连接的方法查找出不在财务部工作的所有雇员信息

SELECT a.\* FROM Employees a

INNER JOIN Departments b

ON a.DepartmentID=b.DepartmentID

WHERE b.DepartmentName!='财务部';

5. 使用外连接方法查找出所有员工的月收入

SELECT \* FROM Employees a

LEFT OUTER JOIN Salary b ON

a.EmployeeID=b.EmployeeID;

6. 查找财务部收入在2000元以上的雇员姓名及其薪水详情

SELECT a.\*,b.InCome FROM Employees a

INNER JOIN Salary b

ON a.EmployeeID=b.EmployeeID

INNER JOIN Departments c

ON a.DepartmentID=c.DepartmentID

WHERE b.InCome>2000 AND c.DepartmentName='财务部';

7. 查询研发部在1976年以前出生的雇员姓名及其薪水详请

SELECT a.\*,b.InCome FROM Employees a

INNER JOIN Salary b

ON a.EmployeeID=b.EmployeeID

INNER JOIN Departments c

ON a.DepartmentID=c.DepartmentID

WHERE c.DepartmentName='研发部' AND a.Birthday>'1976'

四、聚合函数的使用

1. 求财务部雇员的平均收入

SELECT AVG(a.InCome) FROM Salary a

INNER JOIN Employees b

ON a.EmployeeID=b.EmployeeID

INNER JOIN Departments c

ON c.DepartmentID=b.DepartmentID

WHERE c.DepartmentName='财务部';

2. 查询财务部雇员的最高和最低收入

SELECT MIN(a.InCome),MAX(a.InCome) FROM Salary a

INNER JOIN Employees b

ON a.EmployeeID=b.EmployeeID

INNER JOIN Departments c

ON c.DepartmentID=b.DepartmentID

WHERE c.DepartmentName='财务部';

3. 求财务部雇员的平均实际收入

SELECT AVG(a.InCome-a.OutCome) FROM Salary a

INNER JOIN Employees b

ON a.EmployeeID=b.EmployeeID

INNER JOIN Departments c

ON c.DepartmentID=b.DepartmentID

WHERE c.DepartmentName='财务部';

4. 查询财务部雇员的最高和最低实际收入

SELECT MIN(a.InCome-a.OutCome),MAX(a.InCome-a.OutCome) FROM Salary a

INNER JOIN Employees b

ON a.EmployeeID=b.EmployeeID

INNER JOIN Departments c

ON c.DepartmentID=b.DepartmentID

WHERE c.DepartmentName='财务部';

5. 求财务部雇员的总人数

SELECT COUNT(a.EmployeeID) FROM Employees a

INNER JOIN Departments b

ON a.DepartmentID=b.DepartmentID

WHERE b.DepartmentName='财务部';

6. 统计财务部收入在2500元以上的雇员人数

SELECT COUNT(a.EmployeeID) FROM Employees a

INNER JOIN Departments b

ON a.DepartmentID=b.DepartmentID

INNER JOIN Salary c

ON c.EmployeeID=A.EmployeeID

WHERE c.InCome>2500 AND b.DepartmentName='财务部';

五、GROUP BY、ORDER BY子句的使用

1. 查找Employees表中男性和女性的人数

SELECT Sex,COUNT(Sex) FROM Employees GROUP BY Sex;

2. 按部门列出在该部门工作的员工的人数

--使用内连接的方法

SELECT b.DepartmentName,COUNT(a.EmployeeID) FROM Employees a

INNER JOIN Departments b

ON a.DepartmentID=b.DepartmentID

GROUP BY b.DepartmentName;

--使用多表查询方法,GROUP BY后面的字段,必须出现在SELECT语句要查询的字段中。

SELECT b.DepartmentName,COUNT(a.EmployeeID) FROM Employees a,Departments b

WHERE a.DepartmentID=b.DepartmentID

GROUP BY b.DepartmentName;

3. 按员工的学历分组，排列出本科、大专、硕士的人数

SELECT a.Education,COUNT(a.EmployeeID) FROM Employees a

GROUP BY a.Education;

4. 查找员工数超过2的部门名称和雇员数量

SELECT b.DepartmentName,COUNT(a.EmployeeID) FROM Employees a,Departments b

WHERE a.DepartmentID=b.DepartmentID

GROUP BY b.DepartmentName

HAVING COUNT(a.EmployeeID)>2;

5. 按员工的工作年份分组，统计各个工作年份的人数，例如工作1年的多少人，工作2年的多少人

SELECT a.WorkYear,COUNT(a.EmployeeID) FROM Employees a

GROUP BY a.WorkYear;

6. 将雇员的情况按收入由低到高排列

SELECT a.\*,b.InCome FROM Employees a,Salary b

WHERE a.EmployeeID=b.EmployeeID

ORDER BY b.InCome ASC;

7. 将员工信息按出生时间从小到大排列

SELECT \* FROM Employees ORDER BY Birthday;

8. 在ORDER BY 子句中使用子查询，查询员工姓名、性别和工龄信息，要求按实际收入从大到小排列

SELECT a.Name,a.Sex,a.WorkYear,b.InCome-b.OutCome FROM Employees a,Salary b

WHERE a.EmployeeID=b.EmployeeID

ORDER BY b.InCome-b.OutCome DESC;

六、视图的使用

1. 创建视图

（1）在数据库YGGL上创建视图Departments\_View，视图包含Department表的全部列

USE YGGL

GO

CREATE VIEW Departments\_View

AS SELECT \* FROM Departments;

（2）创建视图Employees\_Departments\_View，视图包含员工号码、姓名、所在部门名称

USE YGGL

GO

CREATE VIEW Employees\_Departments\_View

AS SELECT a.EmployeeID,a.Name,b.DepartmentName FROM Employees a, Departments b

WHERE a.DepartmentID=b.DepartmentID;

（3）创建视图Employees\_Salary\_View，视图包含员工号码、姓名和实际收入三列

USE YGGL

GO

CREATE VIEW Employees\_Salary\_View(EmployeeID,Name,RealInCome)

AS SELECT a.EmployeeID,a.Name,b.InCome-b.OutCome FROM Employees a, Salary b

WHERE a.EmployeeID=b.EmployeeID;

2. 查询视图

从视图Employees\_Salary\_View中查询出姓名为“王林”的员工的实际收入

SELECT \* FROM Employees\_Salary\_View

WHERE Name='王林';

3. 更新视图

（1）向视图Departments\_View中添加一条记录（‘6’，‘广告部’，‘广告业务’）

执行完命令后，分别查看Departments\_View和Department表中发生的变化

INSERT INTO Departments\_View VALUES('6','广告部','广告业务');

（2）尝试向Employees\_Departments\_View中添加一条记录，看看会发生什么情况

INSERT INTO Employees\_Departments\_View VALUES('777777','小林','综合业务部');

--视图或函数'Employees\_Departments\_View' 不可更新，因为修改会影响多个基表。

--DepartmentID分别为Employees表中的外键和Departments表中的主键,并且均不为空,

--这条INSERT语句相当于给为表Employees和表Departments添加一条DepartmentID为空的记录,显然是非法的.

（3）尝试向Employees\_Salary\_View中添加一条记录，看看会发生什么情况

INSERT INTO Employees\_Salary\_View VALUES('777777','小林',3000);

--对视图或函数'Employees\_Salary\_View' 的更新或插入失败，因其包含派生域或常量域。

--RealInCome就是派生域,它是由InCome-OutCome获得的.

（4）将视图Departments\_View中，部门号为‘6’的部门名称修改为‘生产车间’

UPDATE Departments\_View SET DepartmentName='生产车间' WHERE DepartmentID='6';

（5）删除视图Departments\_View中最新增加的的一条记录

DELETE FROM Departments\_View WHERE DepartmentID='6';

**总结：一般不建议直接对视图进行增加、修改、删除的操作。**

4. 删除视图 Employees\_Departments\_View

DROP VIEW Employees\_Departments\_View;

--删除视图的操作并不会删除基表.

5. 在界面工具中操作视图

演示新建视图和查询视图