**MCP**

**主讲老师：张胜利**

**（一）运算符的使用**

**比较运算符：**

**= > < >= <= <> 不等于 !  非**

**逻辑运算符：  
 AND：AND连接两个条件，并且仅当两上条件都为真时才返回true   
  OR：OR连接两个条件，但只要其中任一个为真就返回true         
 NOT：当一个语句中使用了多个逻辑运算符时，首先求not的值，然后求and的值，最后再求or的值。**

**（二）关键字：**

1. **通配符：**
2. **\_  表示一个字符**

**Eg:**

**select meal\_code from meal where meal\_code like ‘c\_’**

1. **％  表示任意长度字符串**

**Eg:**

**select meal\_code from meal where meal\_code like ‘%co\_%’/‘co\_%’/‘%co\_’**

1. **[]  表示括号中所指定范围内的一个字符  
   Eg:**

**select \* from flight where aircraft\_code like ‘9w0[1-2]’**

1. **[^] 表示不在括号中所指定范围内的任意一个字符**

**Eg:**

**select \* from flight where aircraft\_code like '9w0[^1-2]'**

1. **AS(重命名列名或者表名)**

**Eg:**

**SELECT COMPANYNAME AS 公司,USERNAME AS 联系人,PHONE AS 联系电话**

**FROM T\_CUSTOMER AS T**

**3) NULL值**

**NULL值要用特殊的关键字IS 或IS NOT，而不能用！=或<>之类的符号**

**4) IN匹配离散值**

**如果要查询离散值，一个方法是很多个OR条件，可以可以用IN或NOT IN关键字。**

**Eg:选出年龄是23,24和34岁的医生**

**SELECT \* FROM T WHERE AGE IN (23,24,34) <===>**

**SELECT \* FROM T WHERE AGE=23 or AGE=24 or AGE=34  
5)BETWEEN匹配连续值**

**Eg:查询出年龄不介于20到50之间的医生数据**

**SELECT \* FROM T WHERE AGE NOT BETWEEN 20 AND 50**

**6)HAVING数据分组后的再筛选**

**HAVING 是对分组以后的数据进行过滤。有了GROUP BY才能使用HAVING.**

**SELECT 列名1,列名2, 聚合函数 FROM <tbname>**

**GROUP BY 列名1，列名2 HAVING 聚合函数的筛选条件**

**7)TOP限制结果集(对查询出的前几条数据的筛选)**

1. **一般是与ORDER BY连用，而且TOP 必须放在\*型号或者列名的前面。不和ORDER BY也可以使用，但是查询结果不确定**

**Eg:选出工资最高的三个人**

**SELECT TOP 3 NAME, SALARY FROM T\_EMPLOYEE**

**ORDER BY SALARY DESC**

1. **对查询出的中间结果进行筛选，使用子查询。**

**Eg:查询出工资排名是4,5，6名**

**SELECT TOP 3 NAME,SALARY FROM T\_EMPLOYEE**

**WHERE ID NOT IN (SELECT TOP 3 ID FROM EMPLOYEE ORDER BY SALARY DESC )**

**ORDER BY SALARY DESC**

**8) DISTINCT去除重复数据**

**1、查询一列中的无重复数据**

**只能查询列的结果集。DISTINCT要放在SELECT之后，和所有列名的前面**

**SELECT DISTINCT 去重列名 FROM <tbname>**

**2、多列组合去重**

**每个列都一样的，才算重复。如果一列不一样，就不算重复**

**SELECT DISTINCT 去重列名1,去重列名2 FROM <tbname>**

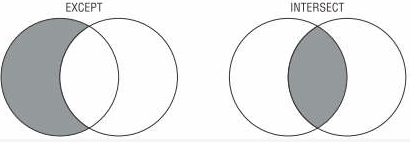
**9)exists、except、intersect**

**1.not exists（true/false）不等于 not in**

**Eg: SELECT \* FROM EMPLOYEE WHERE Exists (SELECT \* FROM EMPLOYEE where id=2) 红字为true 查询出所有,为false 查不出**

**2. intersect是指在两个集合中都存在的数据—数学中的交集运算。**

**3.except是指在第一个集合中存在，但是不存在于第二个集合中的数据（A结果集-(A结果集 intersect B结果集)）。**

****

**Eg:**

**Select ProductID from Production.Product EXCEPT**

**Select ProductID from Sales.SalesOrderDetail Order by ProductID**

**10)PERSISTED**

**指定 SQL Server 数据库引擎将在表中物理存储计算值，而且，当计算列依赖的任何其他列发生更新时对这些计算值进行更新。将计算列标记为 PERSISTED，可允许您对具有确定性、但不精确的计算列创建索引。如果Persisted属性被关掉了，那么计算列只是虚拟列。该列将没有数据存储到磁盘上，并且这些值每次在一个脚本中参照时都会被计算。如果这个属性北设置成激活的，那么计算列的数据将会存储在磁盘中。如果它是Persisted，那么参照列的任何更新将会在计算中自动同步。**

**Eg:**

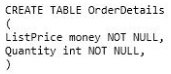
**你需要在新服务器上创建一个名为OrderDetails表， OrderDetails必须满足以下条件：**

**您需要创建一个计算列，用于返回每行的ItemslnStore值和ItemslnWarehous值的总和。**

**每行总和的计算不能每次查询的表的都行都要计算一次。不要使用任务分隔符。**

**该方案必须确保lineitemtotal存储在表的最后一列。**

**正确的T-SQL语句的一部分已经在答题区域中，请补全代码。**

****

**CREATE TABLE OrderDetails(**

**ListPrice money NOT NULL,Quantty int NOT NULL,**

**LineItemTotal AS (ListPrice \* Quantty) PERSISTED 不使用聚合函数)**

**（1）UNION合并查询结果**

1. **合并查询结果集，所以每个结果集必须有相同的列数，每个结果集的列必须类型相容，位置也必须一致，可以合并2个以上的查询结果。但是默认会合并重复数据**
2. **如果列数不一样，必须再相应的位置补齐。默认查询结果是对第一列排序**
3. **如果列名不一致，则表头默认是采用第一个查询的列名**
4. **外键是把多个列组合在一起，而UNION是把多个行组合在一起**

**SELECT 列1,列2, 补齐列 FROM <tbname1>**

**UNION**

**SELECT 列1,列2, 列3 FROM <tbname2>**

**（2）ALL合并查询出的所有结果，不去重**

**除非程序需要，一般要加ALL，因为判断去重，会让数据库消耗掉大量的资源。**

**SELECT 列1,列2, 补齐列 FROM <tbname1>**

**UNION ALL**

**SELECT 列1,列2, 列3 FROM <tbname2>**

**（3）利用UNION多加一行，做一些函数操作**

**Eg：**

**SELECT F\_NAME,F\_SALARY FROM T\_EMPLOYEE UNION ALL**

**SELECT ‘工资合计’,SUM(F\_SALARY) FROM T\_EMPLOYEE**

**UNION ALL**

**SELECT ‘平均工资’,AVG(F\_SALARY) FROM T\_EMPLOYEE**

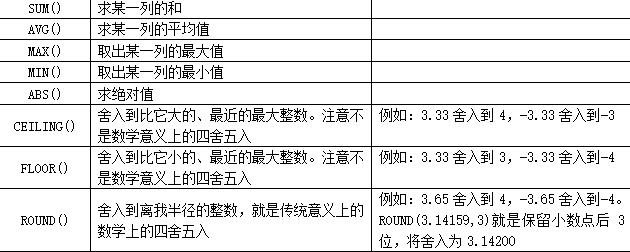
**（4）JOIN联合查询**

**1)JOIN联合查询不光是可以查有主/外键关联的,也可以查没有关联的几张数据表,但是得有一列数据相等.而且可以列出几张表中所有的列**

**SELECT 表1.列1，表2.列3，表2.列4**

**FROM 表1 JOIN 表2 ON 表1.列1=表2.列2**

**2)Left [outer] join(左联接) 返回包括左表中的所有记录和右表中联结字段相等的记录   
3)right [outer] join(右联接) 返回包括右表中的所有记录和左表中联结字段相等的记录  
4)inner join(等值连接) 只返回两个表中联结字段相等的行  
(5)数学函数：**



**(6)字符串函数：**

