

数据库基本 SQL 语句大全

数据库基本----SQL 语句大全

一、基础

1、说明：创建数据库

```
Create DATABASE database-name
```

2、说明：删除数据库

```
drop database dbname
```

3、说明：备份 sql server

```
--- 创建 备份数据的 device
```

```
USE master
```

```
EXEC sp_addumpdevice 'disk', 'testBack', 'c:\mssql7backup\MyNwind_1.d
```

at'

```
--- 开始 备份
```

```
BACKUP DATABASE pubs TO testBack
```

4、说明：创建新表

```
create table tabname(col1 type1 [not null] [primary key],col2 typ  
e2 [not null],...)
```

根据已有的表创建新表：

A: create table tab_new like tab_old (使用旧表创建新表)

B: create table tab_new as select col1,col2... from tab_old defini
tion only

5、说明：删除新表

```
drop table tabname
```

6、说明：增加一个列

```
Alter table tabname add column col type
```

注：列增加后将不能删除。DB2 中列加上后数据类型也不能改变，唯一能改变的是增
加 varchar 类型的长度。

7、说明：添加主键： Alter table tabname add primary key(col)

说明：删除主键： Alter table tabname drop primary key(col)

8、说明：创建索引： create [unique] index idxname on tabname(col...)

删除索引： drop index idxname

注：索引是不可更改的，想更改必须删除重新建。

9、说明：创建视图： create view viewname as select statement

删除视图： drop view viewname

10、说明：几个简单的基本的 sql 语句

选择： select * from table1 where 范围

插入： insert into table1(field1,field2) values(value1,value2)

删除： delete from table1 where 范围

更新： update table1 set field1=value1 where 范围

查找： select * from table1 where field1 like '%value1%' ---like
的语法很精妙，查资料！

排序： select * from table1 order by field1,field2 [desc]

总数: select count as totalcount from table1
 求和: select sum(field1) as sumvalue from table1
 平均: select avg(field1) as avgvalue from table1
 最大: select max(field1) as maxvalue from table1
 最小: select min(field1) as minvalue from table1

11、说明: 几个高级查询运算词

A: UNION 运算符

UNION 运算符通过组合其他两个结果表 (例如 TABLE1 和 TABLE2) 并消去表中任何重复行而派生出一个结果表。当 ALL 随 UNION 一起使用时 (即 UNION ALL), 不消除

重复行。两种情况下, 派生表的每一行不是来自 TABLE1 就是来自 TABLE2。

B: EXCEPT 运算符

EXCEPT 运算符通过包括所有在 TABLE1 中但不在 TABLE2 中的行并消除所有重复行而派生出一个结果表。当 ALL 随 EXCEPT 一起使用时 (EXCEPT ALL), 不消除重复行。

C: INTERSECT 运算符

INTERSECT 运算符通过只包括 TABLE1 和 TABLE2 中都有的行并消除所有重复行而派生出一个结果表。当 ALL 随 INTERSECT 一起使用时 (INTERSECT ALL), 不消除重复行。注: 使用运算词的几个查询结果行必须是一致的。

12、说明: 使用外连接

A、left outer join:

左外连接 (左连接): 结果集既包括连接表的匹配行, 也包括左连接表的所有行。

SQL: select a.a, a.b, a.c, b.c, b.d, b.f from a LEFT OUT JOIN b ON a.a = b.c

B: right outer join:

右外连接 (右连接): 结果集既包括连接表的匹配连接行, 也包括右连接表的所有行。

C: full outer join:

全外连接: 不仅包括符号连接表的匹配行, 还包括两个连接表中的所有记录。

二、提升

1、说明: 复制表 (只复制结构, 源表名: a 新表名: b) (Access 可用)

法一: select * into b from a where 1<>1

法二: select top 0 * into b from a

2、说明: 拷贝表 (拷贝数据, 源表名: a 目标表名: b) (Access 可用)

insert into b(a, b, c) select d,e,f from b;

3、说明: 跨数据库之间表的拷贝 (具体数据使用绝对路径) (Access 可用)

insert into b(a, b, c) select d,e,f from b in '具体数据库' where 条件

例子: ..from b in '&Server.MapPath(".")&"\data.mdb" &"' where..

4、说明: 子查询 (表名 1: a 表名 2: b)

select a,b,c from a where a IN (select d from b) 或者: select a,b,c from a where a IN (1,2,3)

5、说明: 显示文章、提交人和最后回复时间

select a.title,a.username,b.adddate from table a,(select max(adddate) adddate from table where table.title=a.title) b

6、说明：外连接查询(表名 1: a 表名 2: b)

```
select a.a, a.b, a.c, b.c, b.d, b.f from a LEFT OUT JOIN b ON a.a = b.c
```

7、说明：在线视图查询(表名 1: a

```
select * from (Select a,b,c FROM a) T where t.a > 1;
```

8、说明：between 的用法, between 限制查询数据范围时包括了边界值, not between 不包括

```
select * from table1 where time between time1 and time2
```

```
select a,b,c, from table1 where a not between 数值1 and 数值2
```

9、说明：in 的使用方法

```
select * from table1 where a [not] in ( '值1' , '值2' , '值4' , '值6' )
```

10、说明：两张关联表，删除主表中已经在副表中没有的信息

```
delete from table1 where not exists ( select * from table2 where table1.field1=table2.field1
```

11、说明：四表联查问题：

```
select * from a left inner join b on a.a=b.b right inner join c on a.a=c.c inner join d on a.a=d.d where ...
```

12、说明：日程安排提前五分钟提醒

```
select * from 日程安排 where datediff('minute',f 开始时间,getdate())>5
```

13、说明：一条 sql 语句搞定数据库分页

```
select top 10 b.* from (select top 20 主键字段, 排序字段 from 表名 order by 排序字段 desc) a, 表名 b where b.主键字段 = a.主键字段 order by a.排序字段
```

14、说明：前 10 条记录

```
select top 10 * form table1 where 范围
```

15、说明：选择在每一组 b 值相同的数据中对应的 a 最大的记录的所有信息(类似这样的用法可以用于论坛每月排行榜, 每月热销产品分析, 按科目成绩排名, 等等.)

```
select a,b,c from tablename ta where a=(select max(a) from tablename tb where tb.b=ta.b)
```

16、说明：包括所有在 TableA 中但不在 TableB 和 TableC 中的行并消除所有重复行而派生出一个结果表

```
(select a from tableA except (select a from tableB) except (select a from tableC)
```

17、说明：随机取出 10 条数据

```
select top 10 * from tablename order by newid()
```

18、说明：随机选择记录

```
select newid()
```

19、说明：删除重复记录

```
Delete from tablename where id not in (select max(id) from tablename group by col1,col2, ...)
```

20、说明：列出数据库里所有的表名

```
select name from sysobjects where type='U'
```

21、说明：列出表里的所有的

```
select name from syscolumns where id=object_id('TableName')
```

22、说明：列示 type、vender、pcs 字段，以 type 字段排列，case 可以方便地实现多重选择，类似 select 中的 case。

```
select type,sum(case vender when 'A' then pcs else 0 end),sum(case  
se vender when 'C' then pcs else 0 end),sum(case vender when 'B' the  
n pcs else 0 end)
```

```
FROM tablename group by type
```

显示结果：

```
type vender pcs
```

```
电脑 A 1
```

```
电脑 A 1
```

```
光盘 B 2
```

```
光盘 A 2
```

```
手机 B 3
```

```
手机 C 3
```

23、说明：初始化表 table1

```
TRUNCATE TABLE table1
```

24、说明：选择从 10 到 15 的记录

```
select top 5 * from (select top 15 * from table order by i  
d asc) table_别名 order by id desc
```

三、技巧

1、1=1, 1=2 的使用，在 SQL 语句组合时用的较多

“where 1=1” 是表示选择全部 “where 1=2” 全部不选，

如：

```
if @strWhere !='
```

```
begin
```

```
set @strSQL = 'select count(*) as Total from [' + @tblName  
ame + '] where ' + @strWhere
```

```
end
```

```
else
```

```
begin
```

```
set @strSQL = 'select count(*) as Total fro  
m [' + @tblName + ']'
```

```
end
```

我们可以直接写成

```
set @strSQL = 'select count(*) as Total from [' + @tblName  
e + '] where 1=1 安定 ' + @strWhere
```

2、收缩数据库

--重建索引

```

DBCC REINDEX
DBCC INDEXDEFRAG
--收缩数据和日志
DBCC SHRINKDB
DBCC SHRINKFILE

```

3、压缩数据库

```
dbcc shrinkdatabase(dbname)
```

4、转移数据库给新用户以已存在用户权限

```

exec sp_change_users_login 'update_one','newname','oldname'
go

```

5、检查备份集

```
RESTORE VERIFYONLY from disk='E:\dvbbs.bak'
```

6、修复数据库

```

Alter DATABASE [dvbbs] SET SINGLE_USER
GO
DBCC CHECKDB('dvbbs',repair_allow_data_loss) WITH TABLOCK
GO
Alter DATABASE [dvbbs] SET MULTI_USER
GO

```

7、日志清除

```

SET NOCOUNT ON
DECLARE @LogicalFileName sysname,
        @MaxMinutes INT,
        @NewSize INT

```

```

USE          tablename          -- 要操作的数据库名
Select       @LogicalFileName = 'tablename_log',      -- 日志文件名
@MaxMinutes = 10,                                -- Limit on time all
owed to wrap log.
@NewSize = 1
-- 你想设定的日志文件的大小(M)

```

```
-- Setup / initialize
```

```
DECLARE @OriginalSize int
```

```
Select @OriginalSize = size
```

```
FROM sysfiles
```

```
Where name = @LogicalFileName
```

```
Select 'Original Size of ' + db_name() + ' LOG is ' +
```

```

        CONVERT(VARCHAR(30),@OriginalSize) + ' 8K pages o
r ' +
        CONVERT(VARCHAR(30),(@OriginalSize*8/1024)) + 'MB'
FROM sysfiles
Where name = @LogicalFileName
Create TABLE DummyTrans
(DummyColumn char (8000) not null)

DECLARE @Counter INT,
        @StartTime DATETIME,
        @TruncLog VARCHAR(255)
Select @StartTime = GETDATE(),
        @TruncLog = 'BACKUP LOG ' + db_name() + ' WIT
H TRUNCATE_ONLY'

DBCC SHRINKFILE (@LogicalFileName, @NewSize)
EXEC (@TruncLog)
-- Wrap the log if necessary.
WHILE @MaxMinutes > DATEDIFF (mi, @StartTime, GETDATE()) -
- time has not expired
        AND @OriginalSize = (Select size FROM sysfiles Where
e name = @LogicalFileName)
        AND (@OriginalSize * 8 /1024) > @NewSize
BEGIN -- Outer loop.
Select @Counter = 0
WHILE ((@Counter < @OriginalSize / 16) AND (@Counter
r < 50000))
        BEGIN -- update
                Insert DummyTrans VALUES ('Fill Log')
                Delete DummyTrans
                Select @Counter = @Counter + 1
        END
EXEC (@TruncLog)
END
Select 'Final Size of ' + db_name() + ' LOG is ' +
        CONVERT(VARCHAR(30),size) + ' 8K pages or ' +
        CONVERT(VARCHAR(30), (size*8/1024)) + 'MB'
FROM sysfiles
Where name = @LogicalFileName
Drop TABLE DummyTrans
SET NOCOUNT OFF

```

8、说明：更改某个表

```
exec sp_changeobjectowner 'tablename','dbo'
```

9、存储更改全部表

```
Create PROCEDURE dbo.User_ChangeObjectOwnerBatch
@OldOwner as NVARCHAR(128),
@NewOwner as NVARCHAR(128)
AS

DECLARE @Name as NVARCHAR(128)
DECLARE @Owner as NVARCHAR(128)
DECLARE @OwnerName as NVARCHAR(128)

DECLARE curObject CURSOR FOR
select 'Name' = name,
      'Owner' = user_name(uid)
from sysobjects
where user_name(uid)=@OldOwner
order by name

OPEN curObject
FETCH NEXT FROM curObject INTO @Name, @Owner
WHILE (@@FETCH_STATUS=0)
BEGIN
if @Owner=@OldOwner
begin
set @OwnerName = @OldOwner + '.' + rtrim(@Name)
exec sp_changeobjectowner @OwnerName, @NewOwner
end
-- select @name,@NewOwner,@OldOwner

FETCH NEXT FROM curObject INTO @Name, @Owner
END

close curObject
deallocate curObject
GO
```

10、SQL SERVER 中直接循环写入数据

```
declare @i int
set @i=1
while @i<30
begin
```

```
insert into test (userid) values(@i)
set @i=@i+1
end
```