SQL分类：

　　DDL—数据定义语言(CREATE，ALTER，DROP，DECLARE)

　　DML—数据操纵语言(SELECT，DELETE，UPDATE，INSERT)

　　DCL—数据控制语言(GRANT，REVOKE，COMMIT，ROLLBACK)

　　首先,简要介绍基础语句：

　　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:mssql7backupMyNwind\_1.dat’

　　--- 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 type2 [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 definition 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 (#add 并集)

　　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 (#add 差集)

　　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 (#add 交集)

　　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）：结果集几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。

　　其次，大家来看一些不错的sql语句

　　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

SQL: 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 vender when ’C’ then pcs else 0 end),sum(case vender when ’B’ then 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 id asc) table\_别名 order by id desc

　　随机选择数据库记录的方法（使用Randomize函数，通过SQL语句实现）

　　对存储在数据库中的数据来说，随机数特性能给出上面的效果，但它们可能太慢了些。你不能要求ASP“找个随机数”然后打印出来。实际上常见的解决方案是建立如下所示的循环：

Randomize

RNumber = Int(Rnd\*499) +1

While Not objRec.EOF

If objRec("ID") = RNumber THEN

... 这里是执行脚本 ...

end if

objRec.MoveNext

Wend

　　这很容易理解。首先，你取出1到500范围之内的一个随机数（假设500就是数据库内记录的总数）。然后，你遍历每一记录来测试ID 的值、检查其是否匹配RNumber。满足条件的话就执行由THEN 关键字开始的那一块代码。假如你的RNumber 等于495，那么要循环一遍数据库花的时间可就长了。虽然500这个数字看起来大了些，但相比更为稳固的企业解决方案这还是个小型数据库了，后者通常在一个数据库内就包含了成千上万条记录。这时候不就死定了？

　　采用SQL，你就可以很快地找出准确的记录并且打开一个只包含该记录的recordset，如下所示：

Randomize

RNumber = Int(Rnd\*499) + 1

SQL = "SELECT \* FROM Customers WHERE ID = " & RNumber

set objRec = ObjConn.Execute(SQL)

Response.WriteRNumber & " = " & objRec("ID") & " " & objRec("c\_email")

　　不必写出RNumber 和ID，你只需要检查匹配情况即可。只要你对以上代码的工作满意，你自可按需操作“随机”记录。Recordset没有包含其他内容，因此你很快就能找到你需要的记录这样就大大降低了处理时间。

        再谈随机数

　　现在你下定决心要榨干Random 函数的最后一滴油，那么你可能会一次取出多条随机记录或者想采用一定随机范围内的记录。把上面的标准Random 示例扩展一下就可以用SQL应对上面两种情况了。

　　为了取出几条随机选择的记录并存放在同一recordset内，你可以存储三个随机数，然后查询数据库获得匹配这些数字的记录：

　　SQL = "SELECT \* FROM Customers WHERE ID = " & RNumber & " OR ID = " & RNumber2 & " OR ID = " & RNumber3

　　假如你想选出10条记录（也许是每次页面装载时的10条链接的列表），你可以用BETWEEN 或者数学等式选出第一条记录和适当数量的递增记录。这一操作可以通过好几种方式来完成，但是 SELECT 语句只显示一种可能（这里的ID 是自动生成的号码）：

SQL = "SELECT \* FROM Customers WHERE ID BETWEEN " & RNumber & " AND " & RNumber & "+ 9"

　　注意：以上代码的执行目的不是检查数据库内是否有9条并发记录。

　　随机读取若干条记录，测试过

Access语法：SELECT top 10 \* From 表名 ORDER BY Rnd(id)

Sql server:select top n \* from 表名 order by newid()

mysql select \* From 表名 Order By rand() Limit n

　　Access左连接语法(最近开发要用左连接,Access帮助什么都没有,网上没有Access的SQL说明,只有自己测试, 现在记下以备后查)

　　语法 select table1.fd1,table1,fd2,table2.fd2 From table1 left join table2 on table1.fd1,table2.fd1 where ...

　　使用SQL语句 用...代替过长的字符串显示

　　语法：

　　SQL数据库：select case when len(field)>10 then left(field,10)+’...’ else field end as news\_name,news\_id from tablename

　　Access数据库：SELECT iif(len(field)>2,left(field,2)+’...’,field) FROM tablename;

　　Conn.Execute说明

　　Execute方法

　　该方法用于执行SQL语句。根据SQL语句执行后是否返回记录集，该方法的使用格式分为以下两种：

　　1．执行SQL查询语句时，将返回查询得到的记录集。用法为：

　　Set 对象变量名=连接对象.Execute("SQL 查询语言")

　　Execute方法调用后，会自动创建记录集对象，并将查询结果存储在该记录对象中，通过Set方法，将记录集赋给指定的对象保存，以后对象变量就代表了该记录集对象。

　　2．执行SQL的操作性语言时，没有记录集的返回。此时用法为：

　　连接对象.Execute "SQL 操作性语句" [, RecordAffected][, Option]

　　·RecordAffected 为可选项，此出可放置一个变量，SQL语句执行后，所生效的记录数会自动保存到该变量中。通过访问该变量，就可知道SQL语句队多少条记录进行了操作。

　　·Option 可选项，该参数的取值通常为adCMDText，它用于告诉ADO，应该将Execute方法之后的第一个字符解释为命令文本。通过指定该参数，可使执行更高效。

　　·BeginTrans、RollbackTrans、CommitTrans方法

　　这三个方法是连接对象提供的用于事务处理的方法。BeginTrans用于开始一个事物；RollbackTrans用于回滚事务；CommitTrans用于提交所有的事务处理结果，即确认事务的处理。

　　事务处理可以将一组操作视为一个整体，只有全部语句都成功执行后，事务处理才算成功；若其中有一个语句执行失败，则整个处理就算失败，并恢复到处里前的状态。

　　BeginTrans和CommitTrans用于标记事务的开始和结束，在这两个之间的语句，就是作为事务处理的语句。判断事务处理是否成功，可通过连接对象的Error集合来实现，若Error集合的成员个数不为0，则说明有错误发生，事务处理失败。Error集合中的每一个Error对象，代表一个错误信息。