**13、CALL**  
CALL命令可以在批处理执行过程中调用另一个批处理，当另一个批处理执行完后，再继续执行原来的批处理  
CALL command  
调用一条批处理命令，和直接执行命令效果一样，特殊情况下很有用，比如变量的多级嵌套，见教程后面。在批处理编程中，可以根据一定条件生成命令字符串，用call可以执行该字符串，见例子。  
CALL [drive:][path]filename [batch-parameters]  
调用的其它批处理程序。filename 参数必须具有 .bat 或 .cmd 扩展名。  
CALL :label arguments  
调用本文件内命令段，相当于子程序。被调用的命令段以标签:label开头  
以命令goto :eof结尾。  
另外，批脚本文本参数参照(%0、%1、等等)已如下改变:  
     批脚本里的 %\* 指出所有的参数(如 %1 %2 %3 %4 %5 ...)  
     批参数(%n)的替代已被增强。您可以使用以下语法:（看不明白的直接运行后面的例子）  
         %~1         - 删除引号(")，扩充 %1  
         %~f1        - 将 %1 扩充到一个完全合格的路径名  
         %~d1        - 仅将 %1 扩充到一个驱动器号  
         %~p1        - 仅将 %1 扩充到一个路径  
         %~n1        - 仅将 %1 扩充到一个文件名  
         %~x1        - 仅将 %1 扩充到一个文件扩展名  
         %~s1        - 扩充的路径指含有短名  
         %~a1        - 将 %1 扩充到文件属性  
         %~t1        - 将 %1 扩充到文件的日期/时间  
         %~z1        - 将 %1 扩充到文件的大小  
         %~$PATH : 1 - 查找列在 PATH 环境变量的目录，并将 %1  
                       扩充到找到的第一个完全合格的名称。如果环境  
                       变量名未被定义，或者没有找到文件，此组合键会  
                       扩充到空字符串  
    可以组合修定符来取得多重结果:  
        %~dp1       - 只将 %1 扩展到驱动器号和路径  
        %~nx1       - 只将 %1 扩展到文件名和扩展名  
        %~dp$PATH:1 - 在列在 PATH 环境变量中的目录里查找 %1，  
                       并扩展到找到的第一个文件的驱动器号和路径。  
        %~ftza1     - 将 %1 扩展到类似 DIR 的输出行。  
    在上面的例子中，%1 和 PATH 可以被其他有效数值替换。  
%~ 语法被一个有效参数号码终止。%~ 修定符不能跟 %\*使用  
注意：参数扩充时不理会参数所代表的文件是否真实存在，均以当前目录进行扩展  
要理解上面的知识，下面的例子很关键。  
例：  
@echo off  
Echo 产生一个临时文件 > tmp.txt  
Rem 下行先保存当前目录，再将c:\windows设为当前目录  
pushd c:\windows  
Call :sub tmp.txt  
Rem 下行恢复前次的当前目录  
Popd  
Call :sub tmp.txt  
pause  
Del tmp.txt  
exit  
:sub  
Echo 删除引号： %~1  
Echo 扩充到路径： %~f1  
Echo 扩充到一个驱动器号： %~d1  
Echo 扩充到一个路径： %~p1   
Echo 扩充到一个文件名： %~n1  
Echo 扩充到一个文件扩展名： %~x1  
Echo 扩充的路径指含有短名： %~s1   
Echo 扩充到文件属性： %~a1   
Echo 扩充到文件的日期/时间： %~t1   
Echo 扩充到文件的大小： %~z1   
Echo 扩展到驱动器号和路径：%~dp1  
Echo 扩展到文件名和扩展名：%~nx1  
Echo 扩展到类似 DIR 的输出行：%~ftza1  
Echo.  
Goto :eof  
例：  
set aa=123456  
set cmdstr=echo %aa%  
call %cmdstr%  
pause  
本例中如果不用call，而直接运行%cmdstr%，将显示结果%aa%，而不是123456

**14、shift**  
更改批处理文件中可替换参数的位置。  
SHIFT [/n]  
如果命令扩展名被启用，SHIFT 命令支持/n 命令行开关；该命令行开关告诉  
命令从第 n 个参数开始移位；n 介于零和八之间。例如:  
    SHIFT /2  
会将 %3 移位到 %2，将 %4 移位到 %3，等等；并且不影响 %0 和 %1。

**15、setlocal 与 变量延迟**

@echo off  
set a=4  
set a=5 & echo %a%  
pause  
结果：4

例2:  
@echo off  
setlocal enabledelayedexpansion  
set a=4  
set a=5 & echo !a!  
pause   
结果：5  
解说：启动了变量延迟，得到了正确答案。变量延迟的启动语句是“setlocal enabledelayedexpansion”，并且变量要用一对叹号“!!”括起来（注意要用英文的叹号），否则就没有变量延迟的效果。  
分析一下例2，首先“setlocal enabledelayedexpansion”开启变量延迟，然后“set a=4”先给变量a赋值为  
4，“set a=5 & echo !a!”这句是给变量a赋值为5并输出（由于启动了变量延迟，所以批处理能够感知到动态变化，即不是先给该行变量赋值，而是在运行过程中给变量赋值，因此此时a的值就是5了）。  
再举一个例子巩固一下。

**17、ATTRIB  显示或更改文件属性**  
ATTRIB [+R|-R] [+A|-A] [+S|-S] [+H|-H] [[drive:] [path] filename] [/S [/D]]  
  +   设置属性。  
  -    清除属性。  
  R   只读文件属性。  
  A   存档文件属性。  
  S   系统文件属性。  
  H   隐藏文件属性。  
  [drive:][path][filename]  
      指定要处理的文件属性。  
  /S  处理当前文件夹及其子文件夹中的匹配文件。  
  /D  也处理文件夹。