Bat脚本语言

1. 批处理命令：

批处理文件是将一系列命令按照一定的顺序集合在一个可执行的文本文件中，例如bat、cmd或shell等批处理文件，其中的命令称为批处理命令。

1. 常见批处理命令：
2. rem 和 ::
3. echo 和 @
4. pause
5. errorlevel
6. title
7. color
8. mode 配置系统设备
9. goto 和 :
10. find
11. start
12. assoc 和 ftype
13. pushd 和 popd
14. call
15. shift
16. if
17. setlocal 与 变量延迟
18. attrib 显示或更改文件属性

提示：在遇到不懂的命令时，可以再dos下输入 command /?可光看该命令的详细介绍。

1. 命令介绍
2. rem 和 ::

rem为注释命令，用于注释相当于C语言中的“//”，表示该内容不被执行，但能回显（回显的意思是，该命令执行时通过终端显示该命令的执行结果）。:: 也可以起到注释作用，使用有以下注意：

任何以 : 开头的字符行在批处理中都被视为标号，其作用与C语言中的标号语句类似，也可以写成 :+ 的形式；与rem不同的是，:: 注释的语句是不会被回显的。

1. echo 和 @

@ 放在命令前，将关闭该命令的显示，也就是终端在执行该命令时，该命令不会被显示。

echo使用如下：

1. echo [on | off]：打开或关闭命令回显功能，也就是批处理文件执行时终端启动或关闭命令执行结果的显示。
2. echo ： 显示当前echo的设置状态；
3. echo 字符串 ： 终端输出该字符串，相当于C语言中的printf的作用。
4. dos下使用echo [off|on]可关闭屏幕显示，仅留下光标，echo on重新显示dos提示符；
5. echo. ：相当于输出一个换行 . 前面无空格，例在dos下输入echo.|time，配合管道转向，表示在time时间打印后输入一个回车；
6. echo 答复语句|文件名：用于答复命令汇总的提问；
7. echo新增文件或追加内容：echo 字符串>文件名（新增文件并输入字符串）；echo 字符串>>文件名（向文件追加字符）；
8. echo 打印机控制码|打印内容>RN : 向打印机输出打印控制码或打印的内容；
9. echo ^g : 在dos窗口中用于启动鸣响；
10. pause

pause 用于命令执行后暂停（有提示语句），pause > nul （暂停无提示语句）；

1. errorlevel

echo %errorlevel%：命令运行结束后用该语句可以在终端显示该命令执行后的返回码，返回0表示命令执行成功，1表示执行出错；

1. title

为cmd窗口设置新标题，title new title

1. color

color [attr] ： 为窗口字体设置颜色(0~F)

1. mode

设置设备属性（串口配置、设备状态、打印重定向等）如：

Mode con cols=133 lines=15 color 9f（依次设置窗口大小：113列，15行，颜色）

1. goto 和 ：

goto 标号，该命令会跳转到 :标号 处执行命令，相当于C语言中的跳转语句；

1. find 字符串查找命令

用于在文件中查找字符串：find [参数] \”字符串\” [[盘符:] [路径]文件名] 参数如下

/v 显示所有未包含指定字符串的行

/c 显示该字符串所在的所有行

/n 显示行号

/I 搜索时忽略带消息

/off[line] 不要跳过具有脱机属性的文件

Find常和type使用：type [盘符:][路径]文件名 | find \”string\” [>tmpfile] #挑选包含string的行

Type结合使用将去除find命令自带的提示语句（文件名提示）

1. start 程序启动命令

用于当前批处理命令调用外部程序命令，该外部程序在新窗口中运行，批处理程序继续往下执行，如果不使用该命令运行外部程序则需要等待外部程序执行完毕后才能继续执行本程序。例如：start explorer d:\\ 调用图形界面打开d盘。

1. assoc 和 type

文件关联，用于设置双击该文件时，告诉操作系统使用哪个软件将其打开；

1. pushd 和 popd

切换当前目录，（不常用）。

1. call

call command ：命令用于在当前批处理执行时调用另外一个批处理，并且等待另外一个批处理执行完毕后在继续往下执行；

例调用其他批处理文件执行（文件必须有.bat 或 .cmd拓展名） call file.bat

1. shift 更改批处理文件中可替换参数的位置

shift %3 ：将参数3 移动到 2的位置

1. if
2. if [not] errorlevel number (cmd) else (cmd)

not：表示后面紧跟着的指定条件为false的情况下，才执行的意思，可以没有。

errorlevel number：表示该位置的程序执行后返回一个大于或等于number，则该条件为true，否则为false;

（cmd）：要执行的指令，bat脚本中括号表示一条指令，相当于C语言中的if条件执行。

if not define str (

echo undefined str

) else (

echo defined str

)

rem表明如果没有定义str 执行语句命令体1，否则执行命令体2，if和else必须在一条命令上。

1. if [not] string1 == string2 command

if not \”string1\” == \”string2\” (

echo string1 != string2

) else (

echo string1 == string2

)

rem 以上语句判断字符串是否相等

1. if [not] exist filename command

rem 用于判断该文件或目录是否存在

1. if增强语句

if /I string compare-pot string2 rem参数/I表示不区分大小写对比

if CMDEXTVERSION number command rem和errorlevel一样

if DEFINED var command rem判断变量是否定义

if %var% 比较宏 number

比较宏如下：

EQU ：等于

NEQ ：不等

LSS ：小于

LEQ ：小于等于

GTR ：大于

GEQ ：大于等于

1. setlocal 与 变量延迟

变量延迟：当在一行命令中，包含变量赋值时，在该行语句中并不会立即对该便令赋值，而是在执行吓一跳语句时对该变量赋值；

setlocal enabledelayedexpansion：开启变量延迟；

1. set 变量定义

set i=7

rem 定义一个变量并赋值，和C语言变量定义类似

1. attrib 显示或更改文件属性

attrib [+r/-r] [+a/-a] [+s/-s] [+h/-h] ] [[drive:][path][filename]] [/s[/d]]

r：只读文件

a：存档文件

s：系统文件属性

h：隐藏文件属性

文件路径：[[drive:][path][filename]]

/s：表示处理该文件夹时递归处理其子类文件

/d：递归处理该文件夹下的文件夹

1. 常用符号

（1）@：命令回显屏蔽；

关闭该行命令的回显。

（2）%：批处理变量引用符号，%i% 对变量i的引用

用法1：变量的引用 %var%

用法2：命令行参数传递使用，%0~%9 表示参数0(批处理文件本身.bat或cmd)至参数9，%\*表示从参数1开始的所有参数

（3）>：重定向符号

一般用于创建文件（文件存在则截短），并将字符重定向。

（4）>>：重定向符号，一般向一个文件中追加字符。

（5）<、>、<&：重定向符号

<：输入重定向,表示从文件中读入命令

>：输出重定向，除上面提到的用法外，还可以句柄(0、1、2，分别表示标准输入、输出、标准错误)的使用；

1>nul：表示禁止输出正确信息；

2>nul：表示禁止输出错误信息；

（6）|：管道符号；

格式： command1 | comand2 [|command3…]，将第一条命令执行的结果作为第二条命令的参数；

（7）^：转义字符

其用法同C语言中\用法类似。

（8）组合命令：command1 command2 command3… 由于脚本命令只认行不认命名个数，使用组合命令意义在于将多个命令组合起来，当做一个命令来执行，当一个命令失败了也不影响后面的命令执行。

（9）&：command1 & command2 [&…]与命令组合，当组合命令执行时若出现命名执行错误的时候，则终止命令的继续执行；

（10）||：或组合命令，当一条命令执行失败后才执行后面的命令

（11）\”\”：字符串界定符

（12）,：逗号，在某些情况下当做空使用；

（13）；：分号，当命令相同时，可以用来隔离不同的目标，但执行效果不变，如果执行过程中出现错误，会返回错误报告，程序任然执行。

（14）()：括号，被括号包含的命令，被视为一条命令行；

（15）!：叹号

（16）其他符号

:：放在字符前表示标签，类似于C语言中的goto跳转的标签

CR(0D)：命令结束符

escape(1B)：ansi转义字符引导符

space(20)：常用参数鉴定符；

tab(09)：= 不常用参数鉴定符

+copy：命令文件连接符；

\*/?：文件通配符；

/：参数开关引导符；

1. bat变量
2. 系统变量：

%ALLUSERSPROFILE%：本地返回所有用户配置文件的位置。

%APPDATA%：本地返回默认情况下引用程序存储数据的位置。

% CD %：本地返回当前目录字符串。

%CMDCMDLINE%：返回用来启动当前的cmd.exe的准确命令行。

%CMDEXTVERSION%:返回当前的命令处理程序拓展的版本号。

%COMPUTERNAME%：返回计算机名

%COMSPEC%：返回命令行解释器可执行程序的准确路径。

%DATE%：返回当前日期。

%ERRORLEVEL%：返回上一条命令的错误代码。通常用非零值表示错误。

%OS%：返回操作系统名称。

%PATH%：指定可执行文件的收索路径。

%RANDOM%：返回随机数

%TIME%：返回当前系统时间。

1. 命令行参数：%0~%9
2. 自定义变量

set var=值 rem 和C语言中的自定义变量类似，但该变量不用指定数据类型，但是值中不能包含=。

%var% ：变量的引用。

set /p var= rem手动输入便令值，等号后面可以跟提示语句用于人机交互（同scanf用法类似），

set /A 运算表达式：该命令可以用于算术运算，其支持的算术运算符优先级依次如下：

! ~ - ：一元运算符；

/ % \* ：算术与运算符，取余在bat中须写成%%；

+ - ：算术运算符；

<< >> ：二进制移位运算符；

& ：二进制与运算；

^ ：按位异或运算；

| ：按位或；

= \*= /= %= += -=：赋值；

^= |= <<= >>=：二进制赋值；

, ：表达式分割符；

1. set变量字符串的替换

格式：set %字符串:str1=str2% rem该语句表示，将字符串（可以使一个字符串变量也可以是PATH环境变量）的字符串str1替换成字符串str2；

1. 变量截取：%a:~m,n%

将字符串a从m（缺省为0，负数为倒数）开始，截取n（缺省为全部，负数为截取到倒数n）个字符；例set b=%a:~1,2% rem 将a从字符1到字符2截取。

1. for循环命令

（1）基本格式：for 参数 %%variable in (set) DO command [command-paramenters]

%%variable：指定一个单一字母表示可替换的变量；

(set)：指定一个或一组文件，可使用通配符。

command：指定对每个文件执行的命令。

command-parameters：为特定命令指定参数或命令行开关。

for每次循环时在set中提取一个元素，把提取的值赋予形式变量I，然后将变量待到command中参与command命令的运行，直至set中的元素被提取完毕退出循环。

（2）参数 /d

匹配目录名：for /d %%i in (set) do command [command-parameters]

for /d %% in (d:\\) do (echo %%i) rem 打印d盘下所有的目录

（3）参数 /r

递归参数：for /r [[drive:]path] %%variable in (set) do command [command-parmeters]

检查[drive:]path 为根的目录树，指向每个目录中的for语句，默认为当前目录；

（4）参数 /L

以迭代的形式执行该循环体（指定循环次数）：for /l %%i in (start, step, end) do command [command-parameters]

start：起始值；

step：增量值，可以为负数；

end：结束值；

for /l %%i (1, 1, 5) do (echo %%i) rem 执行打印1~5

（5）参数 /F

迭代以及文件解析：使用文件解析类处理命令输出、字符串以及文件内容。

for /f [\”options\”] %%i in (file-set/\”string\”/#39;command) do command [command-parmeters]

\”options\”：文件解析方式

eol=：文件以 ; 号开头的为注释

tokens=n,m\* 将文件中每行的第n段和第m段分别赋给对应变量，\*表示其余的赋值给剩下的变量；

delims=,- ：（减号后含空格）以逗号和减号以及空格作为分隔符；

（6）for循环中的变量

1）~I：去除引号

@echo off

echo ^"1111>temp.txt rem 字符头含一各双引号

echo "2222">>temp.txt rem 字符被双引号包引

echo 3333^">>temp.txt rem 字符尾部含一个双引号

echo ^"4444^"44>>temp.txt rem 字符头和字符中间含双引号

echo ^"55^"55^"55>>temp.txt

FOR /F %%i in (temp.txt) DO (

echo %%~i

)

pause

del temp.txt

rem 无头不删除,有头连尾删，字符中不删

2）%~fI 将 %I 将文件名拓展到一个完全合格的路径名，也就是拓展该文件的绝对路径名。

3）%~dI 仅将%I拓展为一个驱动器号，也就是该文件在哪个盘符下

4）%~pI 仅将%I拓展为一个路径名，不拓展文件名，相当于Linux 下pwd的效果。

5）%~nI 仅将I拓展到一个文件名，也就是只打印文件名。

6）%~xI 仅将%I拓展为一个拓展名。

7）%~sI 拓展的路径致函有短名。

8）%~aI 拓展到文件属性。

9）%~tI 拓展到日期/时间

10）%~zI 拓展到文件的大小。

11）%~$PATH:I 查找列在路径环境变量中的目录，并将%I拓展到找到的第一个完全合格的名称，未找到则返回空字符串。

1. 延时

for循环延时、call：delay time 、调用vbs脚本延时。

1. 随机数%random%

random：系统变量，返回0-2767之间的任意数，有cmd.exe生成。

注意：

管道命令的优先级高于重定向命令优先级；