

cmd.exe (command interpreter MS Windows NT/2000/XP)

O čem jsou následující řádky

- konfigurace prostředí, editování příkazového řádku, doplňování jmen příkazů a souborů, historie příkazů
- proměnné, integer aritmetika, náhodná čísla, zástupné znaky, regulární výrazy
- programování: podmínky, cykly, skoky, procedury, lokalizace platnosti proměnných, substituce příkazů, přesměrování a roury
- podpůrné příkazy: nápověda, porovnání souborů, prohledávání souborů, třídění, plánovač úloh

cmd.exe

Pozor: nezaměňovat s **command.com** (command interpreter MS-DOS a Windows 9x).

Nápověda: CMD **/?**, stejně i u všech příkazů Windows.

Konfigurovatelné vlastnosti:

- rozšíření příkazů (**command extensions**) – default,
- doplňování jmen souborů a adresářů (file and directory **name completion**) – není default,
- pozdržené dosazení za proměnné (delayed environment **variable expansion**) – není default,
- barvy – default COLOR 07, viz COLOR **/?**,
- startovací skript.

Vlastnosti lze konfigurovat:

- **trvale v registru**: REGEDIT, klíče HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Command Processor nebo HKEY_CURRENT_USER\Software\Microsoft\Command Processor, položky EnableExtensions, CompletionChar (obvyklá hodnota 0x6 pro CtrlF), PathCompletionChar (0x4 pro CtrlD, 0x20 pro vypnutí), DelayedExpansion, AutoRun a DefaultColor,
- jednorázově parametrem příkazu CMD (/E:ON, /F:ON, /V:ON) nebo
- v nezávislém skriptu příkazem SETLOCAL ENABLEEXTENSIONS a SETLOCAL ENABLEDELAYEDEXPANSION.

Editování

- doplňování jmen souborů a adresářů: abecední rotace kandidátů opakovaným stiskem **CtrlF/CtrlD**, v protisměru s **Shift**em,
- historie příkazů: šipky Up/Down, PageUp/Down, seznam příkazů F7, opakování příkazu číslem F9.

Formát příkazového řádku

komentářový řádek: **REM** nebo **::**

řetězení: command1 **&&** command2

shlukování: (**command1 && command2**)

pokračovací řádky: (command1
command2)

přesměrování vstupu: command **<file**

výstupu: command **>file** nebo **1>file**, přidání command **>>file** nebo **1>>file**

chybového výstupu: command **2>errfile**, přidání command **2>>errfile**

kombinace: command **>file 2>%1**, command **>>file 2>>errfile** apod.

Proměnné

nastavení: **SET** var=value

zrušení: **SET** var=

použití: **%var%** náhrada hodnotou při vstupu na řádek nebo do víceřádkové struktury

!var! náhrada hodnotou těsně před použitím (delayed variable expansion)

var výjimečně, např. v příkazu SET

př. výpis proměnné var: **SET** var
ECHO %var%

proměnné začínající na v: **SET** v

všechny proměnné: **SET**

pozor: **SET %var%** znamená **SET value** (pokud bylo SET var=value)

interaktivně: **SET /P** var=vyzva

integer počty: **SET /A** [var=] expression

operátory podle priority: **(, */%, +-, <>, &, ^, |, = *= /= %=** atd.

př. **SET /A v1=1+2*3,v2=5/3,v3=5%3 && SET v** » v1=7, v2=1, v3=2
SET /A v1=v1+1 » v1=8 (tj. v1, nikoliv %v1%)
ale i **SET /A v1+=1 && SET /A v1*=2** » v1=9, v1=18 (tj. C-přirazení)
SET /A v4+=1 » v4=1 (tj. inicializace nulou)
pozor: **SET v4+=1** » v4+ = 1 (tj. set bez /a)
16, 8: **SET /A x10 && SET /A 0x10 && SET /A 010 && SET /A 09** » 0, 16, 8, invalid

podřetězce: **%var:~offset,length%** pro překročení offset znaků a vyjmutí následujících length znaků
%var:~offset% pro překročení offset znaků a vyjmutí všech následujících znaků
! místo % pro pozdržené dosazení za var

pozor na potřebu pozdrženého dosazení za proměnné při řetězení příkazů:

př. **SET var=**
SET var=1234567890 && ECHO !var:~2,3! » 345
ale **SET var=1234567890 && ECHO %var:~2,3%** » %var:~2,3%

substitute: **%var:old=new%** pro náhradu old řetězcem new
%var:old=% pro vymazání old
! místo % pro pozdržené dosazení za var

př. **SET znamky=same jednický** » same jednický
SET znamky=%znamky:same=zadne% » zadne jednický
SET znamky=%znamky:zadne = % » jednický

dynamické proměnné: **%CD%** (current dir), **%DATE%**, **%TIME%**, **%RANDOM%** (rozsah 0..2**15-1) aj.

Podmíněné příkazy IF a ELSE

příkaz **IF condition commands**

struktura **IF condition (commands) ELSE (commands)**

pozor: ELSE nesmí být na začátku řádku,

tj. nelze: **IF condition (commands) ELSE (commands)** lze: **IF condition (commands)) ELSE (commands)**

condition: **[NOT] EXIST** filename/dirname test (ne)existence souboru nebo adresáře
[NOT] DEFINED var test existence proměnné
[NOT] ERRORLEVEL number test návratové hodnoty (př. po EXIT /B errCode)
[/I] [NOT] string1==string2 test shody řetězců (volitelně velká=malá)
[/I] [NOT] string1 op string2 generický (numerický, jinak řetězcový) test shody
kde op je **EQU, NEQ, LSS, LEQ, GTR, GEQ**
př. **0xF==15** » false (řetězce)
0xF EQU 15 » true (hex vs. dec)
07 EQU 7 » true (okt vs. dec)
08 EQU 8 » false (08 je invalid, ne číslo, tedy řetězec)

Cyklus FOR

%v, %%v řídicí proměnná cyklu: v je jeden znak, %v NEQ %V, %v interaktivně, %%v ve skriptu

výčtový cyklus (default)

FOR %v IN (set) DO commands

vypnutí ozvěny: ... DO @command && @command

nebo ... DO @(commands)

př. **FOR %i IN (1 2 3) DO ECHO 00%i** » 001 // 002 // 003

aritmetický cyklus (/L)

```
FOR /L %v IN (start step end) DO commands
př. SET min=1 && SET max=3 && FOR /L %i IN (!min! 1 !max!) DO ECHO 00%i
                                         » 001 // 002 // 003
pozor: FOR %i IN (%min% 1 %max%) DO ECHO 00%i » 001 // 001 // 003 (chybí /L)
```

výčtový cyklus pro soubory (default) a adresáře (/D)

```
FOR %f IN (file-set) DO commands
FOR /D %d IN (dir-set) DO commands
FOR /R [dir] [/D] %f IN (set) DO commands s rekurzivním průchodem stromu z dir
kde file-set/dir-set mohou obsahovat zástupné znaky *, ?
modifikátory řídicí proměnné %f, je-li jménem souboru/adresáře:
abs. cesta %~ff, jen disk %~df, jen jméno %~nf, jméno s příponou %~nxf,
atributy-datum-velikost %~atzf aj.
př. FOR %f IN (*) DO echo %f » file1 // file2
FOR /R . %d IN (.) DO echo %~nxd » dir1 // dir11 // dir111 // dir12 // dir2 („do hloubky“)
FOR /D /R . %d IN (*) DO echo %~nxd » dir1 // dir2 // dir11 // dir12 // dir111 („do šířky“)
```

cyklus pro analýzu souborů nebo výstupu příkazu (/F)

```
FOR /F ["options"] %a IN (file-set nebo "string" nebo 'command') DO commands
kde options jsou SKIP=n pro vynechání prvních n řádků
EOL=c pro nastavení znaku konce řádku
DELIMS=xxx pro nastavení oddělovačů (default mezera a tab)
TOKENS=x,y,m-n[*] pro výběr položek na řádku, značených pak %a, %b ad.
(default 1)
USEBACKQ pro alternativní zápis (file-set nebo 'string' nebo `command`)
př. vynechání prvního řádku a řádků začínajících na !
FOR /F "SKIP=1 EOL=! TOKENS=*" %a IN (file) DO ECHO %a >> file-without-comments
př. přeskládání sloupců ze souboru
FOR /F "TOKENS=1-3*" %a IN (file1234) DO ECHO %c %b %a %d >> file3214
př. poskládání jmen proměnných z výpisu příkazem SET do proměnné list
SET list= && ( FOR /F "DELIMS==" %a IN ('SET') DO @SET list=!list! %a ) && ECHO !list!
```

Skok GOTO

```
skok na návěští :label
commands
GOTO label
ukončení skriptu GOTO :EOF nebo také EXIT /B [errCode]
```

Volání podprogramů CALL script a CALL :label a lokalizace proměnných pomocí SETLOCAL a ENDLOCAL

```
skok do jiného skriptu bez návratu script parameters
skok do jiného skriptu s návratem CALL script parameters
skok na návěští s návratem (po GOTO :EOF) CALL :label parameters
použití argumentů %1, %2, ..., %9; %0 pro jméno skriptu
posun argumentů (%n+1=>%n, %n+2=>%n+1 ad.) SHIFT [/n]
modifikátory argumentů %~fdnxatz1, viz text u řídicí proměnné cyklu
lokální prostředí pro proměnné SETLOCAL && ... && ENDLOCAL
```

```
př. @ECHO OFF nebo @ECHO OFF
CALL :eval 1+1 CALL eval 1+1
GOTO :EOF GOTO :EOF
-----
:eval soubor eval.bat
SETLOCAL SETLOCAL
SET /A tmp=%1 SET /A tmp=%1
ECHO %tmp% ECHO %tmp%
ENDLOCAL ENDLOCAL
GOTO :EOF
```

Podpůrné příkazy

HELP pro seznam příkazů

START pro spouštění příkazů asociovaných s příponou souborů

START file.pdf

nezapomínat na filename completion

FC a (starý) **COMP** pro porovnání souborů (file compare)

FC file1 file2

COMP file1 file2

pouze pro soubory téže délky

varianty: **/N** uvede čísla řádků, **/W** ignoruje rozdíly v mezerách a tabs (white space)

FINDSTR a (starý a pomalý) **FIND** pro hledání řetězců

FINDSTR "multiple search strings" files

vrací řádky obsahující některý ze strings

FINDSTR **/C**: "exact search string" files

vrací řádky obsahující string přesně

FINDSTR **/R /C**: "regular expression" files

vyhodnocuje regulární výrazy

FINDSTR "strings" < file-in > file-out

command1 | FINDSTR "strings" | command2

varianty: **/V** vrátí řádky neobsahující řetězec, **/I** ignoruje velikost písmen, **/X** exaktní shoda řádku s řetězcem,

/M vypíše jen jména souborů, **/F**:file čte jména souborů z file, **/G**:file čte řetězec z file

regulární výrazy: **.** (lib. 1 znak), ***** (opakovač 0 a více), **[xyz]** (jeden z nich), **[^xyz]** reverze, **[x-y]** rozsah,

^ \$ začátek a konec řádku, **\< \>** začátek a konec slova, **** zastínění metaznaku

př. FINDSTR **/V /I /R** "^REM" file.bat

vypíše pouze nekomentářové řádky

FIND "string" file

vrací řádky obsahující string

FIND "string" < file-in > file-out

command1 | FIND "string" | command2

varianty: **/V** reVerze - vrátí řádky neobsahující řetězec, **/I** ignoruje velikost písmen, **/C** pro count

PAUSE pro pozastavení skriptu

PUSHD a **POPD** pro přepnutí adresáře a návrat

PUSHD \\karel\k

dočasně vytvoří síťový disk Z:, není-li již

POPD

SORT pro třídění

SORT file-in /output file-out

SORT < file-in > file-out

command | SORT > file-out

varianty: **+n** od n-tého sloupce, **/reverse**

AT pro plánovač úloh

AT time **/I**[INTERACTIVE] command

nastaví termín spuštění úlohy

AT

vypíše id a termín nastavených úloh

AT id **/D**[ELETE]

odstraní úlohu z fronty

AT **/?**

poradí

př.: AT 17:00 **/I CMD /C START** caj-o-pate.mp3

spustí něco o páté

anebo do minuty...

@ECHO OFF

SETLOCAL

lokalizace proměnných

SET t=%TIME%

něco jako 16:32:04,80

SET h=%t:~0,2%

znaky 1-2

SET m=%t:~3,2%

znaky 4-5

IF %m%==59 (SET /A h+=1 && SET m=00

přičtení minuty (nespouštět před půlnocí)

) ELSE SET /A m+=1

SET newtime=%h%:%m%

nastavení času spuštění

AT %newtime% **/I CMD /C START** %CD%\znelka.mp3

task scheduler

ENDLOCAL