

# Dávkové soubory ve Windows

## Význam dávkových souborů

Dávkové soubory, nazývané také **dávkové programy** nebo **skripty**, mohou usnadnit provádění rutinních a často se opakujících úloh.

Dávkový soubor je **neformátovaný textový soubor**, který obsahuje jeden nebo více příkazů a jeho název má příponu **BAT** nebo **CMD**.

## Spouštění dávkových souborů

Zadáte-li do příkazového řádku název souboru, příkazový procesor Cmd.exe bude postupně provádět příkazy tak, jak jsou v tomto souboru uvedené.

Např. dávkový soubor **zaloha.bat** uložený v kořenové složce disku D: spustíme takto:

**D:\>zaloha.bat** (potvrdíme klávesou **Enter**)

nebo v kratší formě stačí zadat **D:\>zaloha** (přípony **.bat** i **.cmd** totiž patří k tzv. *spustitelným příponám souborů* – podobně jako **.exe** či **.com** – a nemusí být proto při spouštění zadávány).

## Příkazy v dávkových souborech

V dávkovém souboru lze uvést **libovolný příkaz příkazového řádku** (např. **cls**, **md**, **copy**, **ping** apod.).

Kromě běžných příkazů existují **speciální příkazy určené pro dávkové soubory** - například příkazy **for**, **goto** a **if** umožňují podmíněné zpracování příkazů zahrnutých do dávkového souboru, příkaz **pause** pozastaví běh dávky, jiné příkazy umožňují řídit vstup a výstup (např. **echo**), volat další dávkové soubory (**call**) apod.

*Příklad obsahu dávkového souboru **zaloha.bat**:*

```
1 @echo off
2 cls
3 title DAVKOVY SOUBOR zaloha.bat
4 echo Po stisku klavesy spustite davkovy soubor zaloha.bat,
5 echo ktery provede zalohu souboru ze slozky C:\BP\BIN
6 echo do slozky C:\zaloha
7 echo.
8 echo Chcete-li davku ukoncit, stisknete CTRL + C
9 echo Pro provedeni davky stisknete libovolnou klavesu
10 pause
11 rem Vytvori adresar zaloha
12 c:
13 cd \
14 IF EXIST zaloha rd zaloha /S /Q
15 md zaloha
16 rem Zkopiruje vsechny soubory ze slozky C:\BP\BIN\*.PAS
17 IF EXIST C:\BP\BIN\*.PAS (
18     xcopy C:\BP\BIN\*.PAS C:\zaloha /M
19 ) ELSE (
20     echo Nenasel jsem zadny pozadovany soubor
21     echo Zaloha nebude provedena
22     goto KONEC
23 )
24 rem Zmeni atributy kopirovanych souboru
25 attrib C:\zaloha\*. * +R -A
26 rem Vypise obsah slozky C:\BP\BIN do souboru pascal.txt
27 dir C:\BP\BIN\*. * > C:\zaloha\pascal.txt
28 rem Zobrazí soubor pascal.txt v Poznamkovem bloku
29 notepad.exe C:\zaloha\pascal.txt
30 :KONEC
31 echo Konec davky
```

## Spuštění dávkového souboru s parametry

Za parametry dávkového souboru označujeme údaje, které jsou při spuštění dávky zapsány za samotným názvem souboru a jsou vzájemně odděleny mezerami:

```
D:\>zaloha.bat zari2009 *.PAS
```

V tomto případě spustíme dávkový soubor se dvěma parametry – prvním je údaj **zari2009** a druhým **\*.PAS**.

Parametry ovlivňují chování skriptu – např. údaj **zari2009** můžeme v dávce použít pro označení složky, do které bude provedena záloha; druhý údaj **\*.PAS** např. určí, které soubory chceme zálohovat.

Při každém spuštění dávky mohou být samozřejmě zadány jiné parametry – např. **rijen2009** a **\*.EXE** (záloha bude v tomto případě provedena do složky **rijen2009** a budou zálohovány všechny programové soubory).

## Parametry jako proměnné v dávkovém souboru

Konzole Cmd.exe podporuje tzv. **rozšiřitelné proměnné parametrů dávky %0 až %9**. Použijeme-li parametry dávky v dávkovém souboru, bude proměnná **%0** nahrazena názvem dávkového souboru a parametry **%1 až %9** odpovídajícími argumenty zadanými v příkazovém řádku.

V našem konkrétním případě by tedy platilo, že v dávkovém souboru bude:

```
%0 = zaloha.bat
```

```
%1 = zari2009
```

```
%2 = *.PAS
```

*Příklad obsahu dávkového souboru **zaloha.bat** s využitím parametrů:*

```
1 @echo off
2 cls
3 title DAVKOVY SOUBOR %0
4 echo Po stisku klavesy spustite davkovy soubor %0,
5 echo ktery provede zalohu souboru %2 ze slozky C:\BP\BIN
6 echo do slozky C:\%1
7 echo.
8 echo Chcete-li davku ukoncit, stisknete CTRL + C
9 echo Pro provedeni davky stisknete libovolnou klavesu
10 pause
11 rem Vytvori adresar %1
12 c:
13 cd \
14 IF EXIST %1 rd %1 /S /Q
15 md %1
16 rem Zkopiruje %2 soubory ze slozky C:\BP\BIN\
17 IF EXIST C:\BP\BIN\%2 (
18     xcopy C:\BP\BIN\%2 C:\%1 /M
19 ) ELSE (
20     echo Nenasel jsem zadny pozadovany soubor
21     echo Zaloha nebude provedena
22     goto KONEC
23 )
24 rem Zmeni atributy kopirovanych souboru
25 attrib C:\%1\*. * +R -A
26 rem Vypise obsah slozky C:\BP\BIN do souboru pascal.txt
27 dir C:\BP\BIN\*. * > C:\%1\pascal.txt
28 rem Zobrazí soubor pascal.txt v Poznamkovem bloku
29 notepad.exe C:\%1\pascal.txt
30 :KONEC
31 echo Konec davky
```

## Příkazy dávkových souborů

### REM

Příkaz umožňuje vkládat komentáře (poznámky) do dávkového souboru nebo do konfiguračních souborů.

*Příklad použití:*

**REM** Tento radek je pouze komentář

### ECHO

Zapne nebo vypne zobrazování příkazů nebo zobrazí zprávu. Při použití bez parametrů zobrazí příkaz echo aktuální nastavení zobrazování příkazů.

*Příklady použití:*

**ECHO ON**      *Zapne zobrazování příkazů dávkového souboru*

**@ECHO OFF**    *Vypne zobrazování příkazů dávkového souboru*

**ECHO** Tento radek se vypíše na obrazovku

### GOTO

V dávkovém programu přesměruje systém Windows XP na řádek označený danou jmenovkou. Je-li tato jmenovka nalezena, začne provádění příkazů na následujícím řádku.

*Příklad použití:*

**format a: /s**

**if not errorlevel 1 GOTO KONEC**

**echo** Při formátování došlo k chybě.

**:KONEC**

**echo** Konec dávkového programu.

*Dávkový program formátuje disk v jednotce A: jako systémový disk. Pokud při operaci nedojde k chybě, přesměruje příkaz GOTO systém Windows XP na jmenovku :KONEC.*

### CALL

Z jednoho dávkového programu volá jiný, aniž by došlo k ukončení nadřazeného dávkového programu.

Cílem volání příkazu call mohou být i návěští.

*Syntaxe:*

**CALL** [[Jednotka:][Cesta] Název\_souboru [Parametry\_dávky]] [:návěští [argumenty]]

*Příklady použití:*

**CALL c:\dávka\kopie.bat \*.pas :ARCHIVNI**

*Voláme dávkový soubor kopie.bat, kterému předáváme první parametr \*.pas. Po spuštění volaného dávkového souboru bude proveden skok na návěští :ARCHIVNI*

*Poznámka:*

Pokud zavoláme jiný dávkový soubor přímo, právě prováděný dávkový soubor se přeruší bez možnosti návratu.

### IF

Provádí podmíněné zpracování v dávkových programech.

*Syntaxe:*

**IF** [not] errorlevel číslo příkaz [else výraz]

**IF** [not] řetězec1==řetězec2 příkaz [else výraz]

**IF** [not] exist Název\_souboru příkaz [else výraz]

*Některé parametry podmíněného příkazu:*

**not**

Určuje, že má být příkaz proveden jen v případě, že uvedená podmínka je nepravdivá.

**errorlevel** číslo

Podmínka je vyhodnocena jako pravdivá pouze v případě, že předchozí program spuštěný programem Cmd.exe vrátil ukončovací kód stejný nebo větší než parametr číslo.

**řetězec1==řetězec2**

Podmínka je vyhodnocena jako pravdivá pouze v případě, že řetězec1 a řetězec2 jsou shodné.

**exist** Název\_souboru

Podmínka je vyhodnocena jako pravdivá pouze v případě, že existuje soubor Název\_souboru.

*/i*

Při porovnávání řetězců ignoruje rozdíl mezi velkými a malými písmeny. Přepínač /i lze použít u varianty příkazu if s parametry ve tvaru řetězec1==řetězec2. Tato porovnání jsou generická v tom smyslu, že pokud jsou řetězec1 i řetězec2 složeny pouze z číslic, jsou řetězce převedeny na čísla a je provedeno číselné porovnání.

### Porovnávací\_operátor

Určuje trojpísmenný porovnávací operátor.

Operátor	Popis
EQU	Rovná se
NEQ	Nerovná se
LSS	Menší než
LEQ	Menší než nebo rovná se
GTR	Větší než
GEQ	Větší než nebo rovná se

### Příklady použití:

```
md %1
```

```
IF ERRORLEVEL 1 (  
    echo Chyba %ERRORLEVEL%: Nejde vytvořit adresar  
    goto KONEC  
)
```

*Když nebyl zadán první parametr, dojde k chybě při vytváření složky (příkaz MD). Podmínka vyhodnotí chybu, na obrazovku se vypíše Chyba 1: Nejde vytvořit adresar a poté je proveden skok na návěští KONEC.*

<b>IF "%1"==" " GOTO BEZPAR</b>	<i>Pokud nebyl zadán žádný parametr skoč na návěští BEZPAR.</i>
<b>IF "%1"==" /?" GOTO HELP</b>	<i>Pokud se požaduje nápověda skoč na návěští HELP, jinak pokračuj.</i>
<b>IF %1 EQU 0 GOTO NULA</b>	<i>Pokud bylo v 1. parametru zadáno číslo = 0, vypiš NULA.</i>
<b>IF %1 LSS 0 GOTO ZAPORNE</b>	<i>Pokud bylo v 1. parametru zadáno číslo &lt; 0, vypiš ZAPORNE.</i>
<b>IF %1 GTR 0 ECHO KLDNE</b>	<i>Pokud bylo v 1. parametru zadáno číslo &gt; 0 vypiš KLDNE.</i>

```
IF EXIST %1 GOTO ANO
```

*Existuje soubor zadáný jako první parametr? Tak jdi na návěští ANO, jinak pokračuj.*

```
IF EXIST pokus.txt (  
del pokus.txt  
) ELSE (  
echo pokus.txt nebyl nalezen  
)
```

*Když existuje soubor pokus.txt, smaž ho. V opačném případě vypiš pokus.txt nebyl nalezen.*

### FOR

Opakovaně provede jeden nebo více příkazů.

Syntaxe:

```
for {%proměnná|%%proměnná} in (sada) do příkaz [Parametry_příkazového_řádku]
```

### Příklady použití:

```
cd C:\bp\bin  
for %s in (*.bak *.pas) do (  
    echo %~fs %~zs B  
)
```

*Do proměnné %s postupně vloží odkazy na soubory \*.bak nebo \*.pas a vypíše úplnou cestu k danému souboru (%~fs) a jeho velikost (%~zs).*