

polecenie	opis i składnia	przykład
REM	Zapisuje komentarze (uwagi) w pliku wsadowym lub w pliku CONFIG.SYS REM [komentarz]	REM komentarz
CALL	Polecenie to wywołuje inny plik wsadowy, sterowania do pliku wywołującego CALL [dysk:][ścieżka] [nazwa_pliku] [parametry_wsadowe]	CALL plik.bat
ECHO	Wyświetla komunikaty lub włącza i wyłącza wyświetlanie poleceń ECHO [ON OFF] ECHO [komunikat]	@ECHO OFF ECHO argument1 %1
IF	Określa prawdę (spełnienie warunku), jeśli podane ciągi tekstowe są identyczne IF [NOT] ciąg1==ciąg2 polecenie	IF %1=%2 echo Parametry są równe
IF EXIST	Sprawdza czy istnieje podany plik, następnie wykonuje komendę IF [NOT] EXIST [plik] [komenda]	IF EXIST plik.txt TYPE plik.txt
IF ERRORLEVEL	Określa prawdę (spełnienie warunku), jeśli ostatnio wykonany program zwrócił kod błędu równy lub większy od podanej liczby IF [NOT] ERRORLEVEL liczba polecenie	IF ERRORLEVEL 1 GOTO ABC
GOTO	Służy do ustalenia kolejności wykonywania instrukcji w plikach wsadowych :etykieta - zdefiniowanie etykiety ... - dalsze instrukcje GOTO etykieta ... :etykieta ...	GOTO ABC :ABC ...
FOR	Instrukcja pętli do automatycznego powtarzania komendy systemu MS - DOS for %%Z in (zestaw) do polecenie Dodanie parametru /L iteruje ciąg numeryczny For /L %%Z IN (start, skok , koniec) do polecenie	FOR %%K IN (zadanie zadanie.bat zadanie.bak) DO TYPE %%K FOR /L %%Z IN (1, 1, 10) do ECHO %%Z
SHIFT	Zmienia położenie wymiennych parametrów w pliku wsadowym. Przy włączonych poleceniach rozszerzeń polecenie SHIFT obsługuje przełącznik /n, którego użycie powoduje, że polecenie przesuwa od n-tego argumentu, gdzie n może być z przedziału między zero a osiem. SHIFT [/n]	SHIFT SHIFT /2

Zadanie:

Utworzyć plik wsadowy, który będzie wypisywał na standardowe wyjście wszystkie swoje parametry wywołania

```
@echo off
:petla
if "%1" == "" goto wyjscie
echo %1
shift
goto petla
:wyjscie
```

Zadanie:

Skrypt wyświetla zawartość zbioru podanego jako parametr.

```
@echo off
cls
echo Uruchomiono skrypt %0
if "%1" == "" goto brak
if not exist %1 goto blad
echo Zawartosc pliku %1
echo -----
echo.
type %1 | more
echo.
echo -----
goto koniec

:brak
echo Nie podano parametru np %0 C:\data\plik.txt
goto koniec

:blad
echo Plik nie istnieje.

:koniec
echo Koniec pracy %0
```

Zadanie:

Modyfikacja poprzedniego zadania. Użytkownik wprowadza lokalizację pliku w trakcie działania skryptu. Dodatkowo w przypadku błędu skrypt umożliwia ponowne podanie lokalizacji.

@echo off

:start

cls

echo Uruchomiono skrypt %0

echo Wprowadz lokalizacje pliku

set /p zm=

echo Wprowadziles %zm%

echo.

if "%zm%" == "" goto brak

if not exist %zm% goto blad

echo Zawartosc pliku %zm%

echo -----

echo.

type %zm% | more

echo.

echo -----

goto koniec

:brak

echo Nie podano parametru np C:\data\plik.txt

goto koniec

:blad

echo Plik nie istnieje.

:koniec

set zm=

choice /c TN /M "Czy zakonczyc dzialanie skryptu?" /T 10 /D N

IF ERRORLEVEL 2 goto start

IF ERRORLEVEL 1 goto koniec2

:koniec2

echo Koniec pracy skryptu %0

Zadanie:

Plik kontroluje czy wywołano go z dobrym parametrem.

```
@echo off
cls
echo Rozpoczyna dzialanie plik %0

if "%1"==" " goto BRAK

for %%x in (a b c d) do if "%1"=="%%x" goto OK
echo Zly parametr!!!!
goto END

:BRAK
echo Nie podano parametrow
goto END

:OK
echo Dobry parametr %1

:END
echo Konczy dzialanie plik %0
```

Zadanie:

Poniższy plik wsadowy tworzy "drabinę w dół" podkatalogów podanych jako parametry, a następnie wraca do katalogu z którego został uruchomiony.

```
@echo off
set /A x=0
:loop
if "%1" == " " goto quit
md %1
cd %1
set /a x=x+1
shift
goto loop
:quit
echo Poziom zagniezdzenia: %x%
pause
for /l %%v in (1,1,%x%) do (echo %%v) & (cd ..)
```

Zadanie:

Napisać plik wsadowy, który będzie tworzył drabinę katalogów w dół, o głębokości określonej pierwszym parametrem wywołania.

```
@echo off
set /A x=%1
for /l %%v in (1,1,%x%) do (md %%v) & (cd %%v))
```

Zadanie (tworzenie pseudo-menu)

Przeanalizuj poniższe pliki wsadowe służące do tworzenie prymitywnego menu.

MENU.CMD

@echo off

cls

type menu.txt

echo on

MENU.TXT

***** > MENU <*****

Dla wybrania : Napisz

Program1 1

Program2 2

Program3 3

1.CMD

cd P1

program1

cd ..

2.CMD

cd P2

program2

cd ..

3.CMD

cd P3

program3

cd ..

Wsk.

1) w bieżącym katalogu powinny znajdować się podkatalogi P1, P2, P2

2) w każdym z tych podkatalogów powinien znajdować się plik uruchamiający odpowiedni program

3) dla celów ćwiczenia możemy utworzyć takie "programy"

progra1.cmd

@echo off

echo Odpaliłeś program numer 1

Zadanie:

Pseudo-menu – 2 wersja. Skrypt wykorzystujący proste menu umożliwiające tworzenie i kasowanie katalogu. Dodatkowo zawiera zabezpieczenia przed:

- **skasowaniem nieistniejącego katalogu**
- **utworzeniem nowego – już istniejącego**
- **niewprowadzeniu przez użytkownika nazwy katalogu**

@echo off

:start

cls

echo MENU:

echo 1. Utworzenie katalogu

echo 2. Skasowanie katalogu

echo 3. Zakonczenie dzialania skryptu

choice /c 123 /M "Co chcesz zrobic?"

IF ERRORLEVEL 3 (echo Wybrano 3
goto koniec)

IF ERRORLEVEL 2 (echo Wybrano 2
goto kasuj)

IF ERRORLEVEL 1 (echo Wybrano 1
goto tworz)

rem *****

:tworz

echo Wprowadz katalog do utworzenia

set /p zm=

if "%zm%" == "" goto brakparam

if exist %zm% goto bladkat2

md %zm%

echo Utworzono %zm%

set zm=

pause

goto start

:bladkat2

echo Podany katalog juz istnieje

set zm=

pause

goto start

rem *****

:kasuj

echo Wprowadz katalog do skasowania

set /p zm=

if "%zm%" == "" goto brakparam

if not exist %zm% goto bladkat

rd /S %zm%

echo Usunieto %zm%

set zm=

```
pause  
goto start
```

```
:bladkat  
echo Podany katalog nie istnieje  
set zm=  
pause  
goto start
```

```
rem *****
```

```
:brakparam  
echo Nie podano nazwy katalogu  
pause  
goto start
```

```
:koniec  
echo Koniec pracy.
```

Zadanie 1

Napisz plik wsadowy, który zapisze zawartość katalogu (pierwszy parametr skryptu) do pliku (drugi parametr skryptu). Następnie skrypt powinien wyszukać w tym pliku linie zawierające określony ciąg znaków (trzeci parametr skryptu) i wypisać je wraz z ich numerami na ekran. W przypadku gdy podana zostanie zbyt mała liczba argumentów skrypt powinien wypisać komunikat o błędzie.

Zadanie 2

Napisz plik wsadowy, który będzie kopiował pliki według założeń:

- plik działa tylko w przypadku gdy został uruchomiony z dwoma argumentami,
- pierwszym argumentem jest nazwa rozszerzenia plików, które będą kopiowane do katalogu podanego jako drugi argument
- jeśli katalog (podany w drugim parametrze) nie istnieje, to ma zostać stworzony

Zadanie 3

Napisz plik wsadowy – kalkulator – dodający/odejmujący/mnożący/dzielący dwie liczby

- ma być zabezpieczony przez zbyt małą i zbyt dużą liczbą argumentów
- dane mają być wprowadzane postaci
nazwa_skryptu 1_liczba działanie 2_liczba np. kalkulator 12 + 4
- wynik ma być przedstawiony w postaci 1_liczba działanie 2_liczba = wynik_działania np.
 $12 + 4 = 16$

Zadanie 4

Napisz plik wsadowy – kalkulator – dodający/odejmujący/mnożący dwie liczby:

- obie liczby mają być wprowadzane w trakcie wykonywania skryptu.
- skrypt ma być zabezpieczony w sytuacji gdyby użytkownik nie wprowadził jakiegokolwiek informacji podczas podawania liczb
- Ma istnieć możliwość ponownego wykonania skryptu (polecenie choice)
- menu ma wyglądać następująco:

MENU:

1. Dodaj
2. Odejmij
3. Pomnoz
4. Zakoncz prace

Co chcesz zrobic? [1,2,3,4]

Zadanie 5

Napisz skrypt kontrolujący z jakim parametrem został wywołany.

- akceptowane są parametry postaci: -a -b -c
- parametr wprowadzamy jako pierwszy argument
- skrypt ma być zabezpieczony w przypadku podania za dużej ilości argumentów lub ich braku