

Лекція 3

КОМАНДНІ ФАЙЛИ. ОСНОВИ ЗМІННИХ У CMD

Завдання на повторення матеріалу Лекції 2. Написати фрагмент коду, який забезпечить наступну функціональну модальність:

1. перемістити всі файли з розширенням exe з директорії D:\1 в директорію D:\2;
2. у разі успіху вивести на екран відповідне повідомлення;
3. у разі помилки створити в папці D:\2 файли 1.exe і 2.exe.

До цього моменту ми використовували з Вами інтерактивний режим інтерпретатора CMD, тобто ми вводили деякі команди і отримували відповідну реакцію інтерпретатора на них.

Це не єдиний режим роботи оболонки cmd. Послідовності команд можуть бути записані у відповідні скрипти — командні файли.

Командний файл в Windows — це звичайний текстовий файл з розширенням bat або cmd, в якому записані допустимі команди операційної системи (як зовнішні, так і внутрішні), а також деякі додаткові інструкції та ключові слова, які надають командним файлам деяку схожість з алгоритмічними мовами програмування.

Спробуємо створити найпростіший командний файл і зберегти його в кореневому каталозі диска D:\ (лістинг 3.1.).

```
script.cmd
c:
CD c:\Temp
DEL /P *.tmp
```

Лістинг 3.1. — Текст найпростішого командного файлу

Для того, щоб запустити цей командний файл перейдемо в командний рядок і введемо його ім'я (лістинг 3.2.).

```
D:\>script.cmd  
  
D:\>c:  
  
c:\Temp>CD c:\Temp  
  
c:\Temp>DEL /P *.tmp  
c:\Temp\1.tmp, Delete (Y/N)? y  
c:\Temp\2.tmp, Delete (Y/N)? y
```

Лістинг 3.2. — Приклад запуску командного файлу

Ми бачимо, що виконання нашого командного файлу (лістинг 3.1.) призвело до видалення файлів із розширенням tmp.

В цілому все так, як би ми ввели зазначену послідовність команд в інтерактивному режимі.

За замовчуванням команди пакетного файлу перед виконанням виводяться на екран, що виглядає не дуже естетично (лістинг 3.3.).

```
script.cmd  
ECHO Hello  
D:\>script.cmd  
  
D:\>ECHO Hello  
Hello
```

Лістинг 3.3. — Приклад виведення дублювання команд

За допомогою команди ECHO OFF можна відключити дублювання команд (лістинг 3.4.), що йдуть після неї (сама команда ECHO OFF при цьому все ж дублюється).

```
script.cmd
ECHO OFF
ECHO Hello
D:\>script.cmd
D:\>ECHO OFF
Hello
```

Лістинг 3.4. — Приклад відключення дублювання команд

Крім цього, можна відключити дублювання будь-якого окремого рядка в командному файлі, написавши на початку цього рядка символ `@` (лістинг 3.5.).

Цю можливість зручно використовувати для того, щоб, наприклад, придушити дублювання самої команди `ECHO OFF`.

```
script.cmd
@ECHO OFF
ECHO Hello
D:\>script.cmd
Hello
```

Лістинг 3.5. — Приклад відключення дублювання команди, розташованої в одному рядку

Також часто буває зручно для перегляду повідомлень, що виводяться з пакетного файлу, попередньо повністю очистити екран командою `CLS`.

Відзначимо, що ми цілком можемо змінити колірне оформлення консолі.

Команда `COLOR` встановлює колір тексту і тла вікна виводу командного файлу і має наступний синтаксис:

COLOR bt

де *b* — шістнадцяткова цифра, що задає колір тла, а *t* — шістнадцяткова цифра, що задає колір тексту. Для кольорів визначені значення, зазначені в табл. 3.1.

Таблиця 3.1. — Допустимі кольори команди COLOR

Код	Колір	Код	Колір
0	Чорний	8	Сірий
1	Синій	9	Світло-синій
2	Зелений	A	Світло-зелений
3	Блакитний	B	Світло-блакитний
4	Червоний	C	Світло-червоний
5	Бузковий	D	Світло-бузковий
6	Жовтий	E	Світло-жовтий
7	Білий	F	Яскраво-білий

Нагадаємо, що команда ECHO призначена для виведення на екран повідомлень. Команда ECHO. (крапка повинна слідувати безпосередньо за словом «ECHO») виводить на екран порожній рядок (лістинг 3.6.).

<pre>script.cmd @ECHO OFF ECHO Hello ECHO. ECHO Bye</pre>
<pre>D:\>script.cmd Hello Bye</pre>

Лістинг 3.6. — Приклад виведення порожнього рядка за допомогою команди ECHO

Використовуючи механізм перенаправлення виводу (символи > і >>), можна направити повідомлення, що виводяться командою ECHO, в певний текстовий файл (лістинг 3.7.).

```
script.cmd
@ECHO OFF
ECHO Hello > hi.txt
ECHO Bye >> hi.txt
```

Лістинг 3.7. — Приклад перенаправлення виводу команди ECHO

За допомогою такого методу можна, скажімо, заповнювати файли-протоколи зі звітом про проведені дії.

Повернемося до нашого прикладу (лістинг 3.1.). Ми з Вами написали простий командний файл, який може видаляти tmp файли в заданій директорії. А якщо ми хочемо видалити tmp файли в іншій директорії?

При запуску командних файлів в командному рядку можна вказувати довільне число параметрів, значення яких можна використовувати всередині файлу. Це дозволяє, наприклад, застосовувати один і той же командний файл для виконання команд з різними параметрами (лістинг 3.8.).

```
D:\>script.cmd c:\Windows c:\Windows\Temp c:\Users
```

Лістинг 3.8. — Приклад введення параметрів командного рядка

Для доступу з командного файлу до параметрів командного рядка застосовуються символи:

*%0,%1, ..., %9 або %**

При цьому замість %0 підставляється ім'я виконуваного пакетного файлу, замість %1, %2,...,%9 — значення перших дев'яти параметрів командного рядка відповідно, а замість %* — всі параметри (лістинг 3.9.).

```
script.cmd
@ECHO OFF
ECHO Batch file name is: %0
ECHO.
ECHO The first argument is: %1
ECHO.
ECHO The second argument is: %2
ECHO.
ECHO The third argument is: %3
```

Лістинг 3.9. — Приклад виведення параметрів командного рядка

Подивимося (лістинг 3.10.), як буде виконуватися командний файл, наведений у лістингу 3.9.

```
D:\>script.cmd c:\ c:\Windows c:\Users
Batch file name is: script1.cmd

The first argument is: c:\

The second argument is: c:\Windows

The third argument is: c:\Users
D:\>script.cmd c:\ c:\Windows
Batch file name is: script1.cmd

The first argument is: c:\

The second argument is: c:\Windows

The third argument is:
```

Лістинг 3.10. — Приклад виконання командного файлу (лістинг 3.9.)

Якщо в командному рядку при виклику пакетного файлу задано менше дев'яти параметрів, то «зайві» змінні з %1...%9 заміщуються порожніми рядками.

Таким чином, стає очевидно (лістинг 3.11.), як створити універсальний командний файл для видалення tmp файлів в будь-якій директорії.

```

script.cmd
@ECHO OFF
ECHO This CMD file is designed to delete tmp files.
CD /D %1
ECHO We moved to the directory %1
ECHO We start removing
DEL /P *.tmp && ECHO Files are deleted successfully

D:\>script.cmd D:\
This CMD file is designed to delete tmp files.
We moved to the directory D:\
We start removing
D:\1.tmp, Delete (Y/N)? y
Files are deleted successfully

```

Лістинг 3.11. — Приклад універсального командного файлу для видалення тимчасових файлів

Завдання. Напишіть командний файл, який буде видаляти файли заданого типу в зазначеній користувачем директорії.

При необхідності можна використовувати більше дев'яти параметрів командного рядка. Це досягається за допомогою команди SHIFT, яка змінює значення параметрів %0,...,%9, копіюючи кожен параметр в попередній, тобто значення %1 копіюється в %0, значення %2 — в %1 і т.д. (лістинг 3.12.). Значення параметру %9 заміщується значенням наступного за %9 параметра.

```

script.cmd
@ECHO OFF
ECHO Batch file name is: %0
SHIFT
ECHO The first argument is: %0
SHIFT
ECHO The second argument is: %0
SHIFT
ECHO The third argument is: %0

D:\>script.cmd par1 par2 par3
Batch file name is: script1.cmd
The first argument is: par1
The second argument is: par2
The third argument is: par3

```

Лістинг 3.12. — Приклад роботи команди SHIFT

Команда SHIFT підтримує ключ /n, що задає початок зсуву параметрів з номера n, де n може бути числом від 0 до 9.

Наприклад, в наступній команді:

SHIFT /2

параметр %2 замінюється на %3, %3 на %4 і т.д., а параметри %0 і %1 залишаються без змін. Відзначимо, що після виконання команди SHIFT вже не можна відновити параметр (%0), який був першим перед зсувом.

У командних файлах є деякі можливості синтаксичного аналізу параметрів. Для параметра з номером n допустимі синтаксичні конструкції (оператори), представлені в табл. 3.2.

Таблиця 3.2. — Синтаксичні конструкції для аналізу параметрів командного рядка

Оператори	Опис
%~Fn	Змінна %n розширюється до повного імені файлу
%~Dn	Зі змінної %n виділяється тільки ім'я диска
%~Pn	Зі змінної %n виділяється тільки шлях до файлу
%~Nn	Зі змінної %n виділяється тільки ім'я файлу
%~Xn	Зі змінної %n виділяється розширення імені файлу
%~Sn	Значення операторів N та X для змінної %n змінюється так, що вони працюють з коротким ім'ям файлу
%~\$PATH:n	Проводиться пошук по каталогах, заданим в змінній середовища PATH, і змінна %n замінюється на повне ім'я першого знайденого файлу. Якщо змінна PATH не визначена або в результаті пошуку не знайдено жодного файлу, ця конструкція замінюється на порожній рядок

Розглянемо наступний приклад. Нехай ми знаходимося в каталозі C:\TEXT і запускаємо пакетний файл з параметром *Оповідання.doc* (%1 = *Оповідання.doc*). Тоді застосування операторів, описаних в табл. 3.2., до параметру %1 дасть наступні результати:

%~F1=C:\TEXT\ Оповідання.doc

%~D1=C:

%~P1=\TEXT

%~N1= Оповідання

%~X1=.doc

%~DP1=C:\TEXT

%~NX1= Оповідання.doc

Завдання. В першому параметрі командного рядка вказано тільки ім'я файлу (з розширенням), що знаходиться в поточному каталозі. Додати до імені файлу рядок *abc*, а до розширення символ *a*.

Добрим стилем написання командних файлів є використання в них коментарів. Коментарі в командних файлах позначаються за допомогою команди REM (лістинг 3.13.):

REM [текст]

```
script.cmd
@ECHO OFF
REM Displaying the file name
ECHO Batch file name is: %0

REM Shift parameters by 1
SHIFT
REM Displaying the first parameter

ECHO The first argument is: %0
```

Лістинг 3.13. — Приклад використання коментарів при написанні командних файлів

Перейдемо до розгляду змінних у командних файлах. У командних файлах є кілька типів змінних:

- вбудовані змінні — вони доступні всім Windows процесам;
- системні змінні — беруться з реєстру Windows;
- вбудовані змінні (локальні) — встановлюються самим користувачем і дійсні тільки протягом поточного сеансу роботи.

З командного рядка Ви можете задати змінну за допомогою команди *SET ім'я_змінної=значення_змінної*

Наприклад,

SET MyVar=Привіт

Зворотною операцією до присвоювання змінної є її підстановка. Змінну можна підставити в будь-якому місці командного файлу уклавши її в символи % (лістинг 3.14.).

<pre>script.cmd @ECHO OFF SET varA=Hello SET varB=World ECHO %varA% ECHO %varB%</pre>
<pre>D:\>script.cmd Hello World</pre>

Лістинг 3.14. — Приклад використання операції підстановки змінної

Зі змінними середовища в командних файлах можна робити деякі маніпуляції. По-перше, над ними можна робити операцію конкатенації (склеювання) (лістинг 3.15.).

```

script.cmd
@ECHO OFF

SET varA=Hello
SET varB=World
SET varC=%varA%%VarB%

ECHO %varC%

```

Лістинг 3.15. — Приклад застосування операції конкатенації до змінних середовища

За допомогою команди SET можливо також переглянути список змінних, які нині зберігаються в пам'яті (лістинг 3.16.).

```

D:\>SET
a=12345
ALLUSERSPROFILE=C:\ProgramData
APPDATA=C:\Users\Sandy-laptop\AppData\Roaming
CommonProgramFiles=C:\Program Files\Common Files
CommonProgramFiles(x86)=C:\Program Files (x86)\Common Files
CommonProgramW6432=C:\Program Files\Common Files
COMPUTERNAME=DESKTOP-1Q6B8MD
ComSpec=C:\WINDOWS\system32\cmd.exe

```

Лістинг 3.16. — Приклад виведення змінних середовища за допомогою команди SET

Насправді список змінних вельми довгий, але серед них є особливі змінні, які можуть бути корисними при написанні командних файлів. Найважливіші змінні середовища наведені в табл. 3.3.

Таблиця 3.3. — Найважливіші змінні середовища

Змінна	Тип	Опис
%CD%	Локальна	Повертає шлях до поточної папки
%COMPUTERNAME%	Системна	Повертає ім'я комп'ютера

Закінчення таблиці 3.3.

%DATE%	Системна	Повертає поточну дату. Використовує той же формат, що і команда DATE /t
%ERRORLEVEL%	Системна	Повертає код помилки останньої використаної команди
%PATH%	Системна	Вказує шлях пошуку для виконуваних файлів
%RANDOM%	Системна	Повертає довільне десяткове число від 0 до 32767
%SYSTEMROOT%	Системна	Повертає розміщення системного каталогу Windows
%TEMP% і %TMP%	Системна і користувача	Повертає шлях до тимчасових папок, які за замовчуванням використовуються додатками, які доступні користувачам, що виконали вхід в систему. Деякі додатки вимагають змінну TEMP, інші — змінну TMP
%TIME%	Системна	Повертає поточний час. Використовує той же формат, що і команда TIME /t
%USERNAME%	Локальна	Повертає ім'я користувача, який виконав вхід в систему

Зі змінної середовища можна витягувати підрядки за допомогою конструкції %ім'я_змінної:~n1,n2%, де число n1 визначає зміщення (кількість символів, що пропускаються) від початку (якщо n1 додатне) або від кінця (якщо n1 від'ємне) відповідної змінної середовища, а число n2 — кількість витягнутих символів (якщо n2 додатне) або кількість останніх символів у змінній, які не увійдуть в підрядок, що виділяється (якщо n2 від'ємне) (лістинг 3.17.). Якщо вказано тільки один від'ємний параметр n, то будуть витягнуті останні n символів.

```
script.cmd
@ECHO OFF
SET DATE=15.04.2019
ECHO %DATE:~0,2%
ECHO %DATE:~0,-8%
ECHO %DATE:~-7,2%
ECHO %DATE:~-4%
```

```
D:\>script.cmd
15
15
04
2019
```

Лістинг 3.17. — Приклад витягування підрядків зі змінної середовища

Можна виконувати процедуру заміни підстрок за допомогою конструкції `%ім'я_змінної:s1=s2%` (в результаті буде повернутий рядок, в якому кожне входження підрядка `s1` у відповідній змінній середовища замінено на `s2`) (лістинг 3.18.).

```
script.cmd
@ECHO OFF
SET a=123456
ECHO %a:23=99%
SET a=123456
ECHO %a:23=%
```

```
D:\>script.cmd
199456
1456
```

Лістинг 3.18. — Приклад заміни підрядків у змінній середовища

Завдання. Написати командний файл, який поміняє місцями ім'я файлу і його розширення. Вважати, що в параметрі вказано тільки ім'я файлу (з розширенням), що знаходиться в поточному каталозі.

Завдання. Написати командний файл, який буде виводити поточну дату в такому форматі: «Сьогодні xx день xx місяць xx рік».

Особливо варто відзначити змінну ERRORLEVEL. У командного процесора cmd.exe є така річ — рівень помилки (error level). Це код виходу (exit code) програми, яку Ви запускали останньою.

Багато програм повертають більшу степінь помилки все більшими й більшими кодами виходу (лістинг 3.19.).

```
script.cmd
@ECHO OFF

CD /d c:\Windows
ECHO %errorlevel%

MORE 8.txt
ECHO %errorlevel%

REM Comparing the name of a variable with its value
"errorlevel"=="0"
ECHO %errorlevel%

D:\>script.cmd
0
Cannot access file C:\Windows\8.txt
1
'"errorlevel"' is not recognized as an internal or external command,
operable program or batch file.
9009
```

Лістинг 3.19. — Значення змінної ERRORLEVEL при деяких помилках

Всі задані в командному файлі за допомогою команди SET змінні середовища зберігаються і після завершення роботи цього файлу, але діють тільки всередині поточного командного вікна.

Також є можливість локалізувати зміни змінних середовища всередині пакетного файлу, тобто автоматично відновлювати значення всіх змінних в тому вигляді, в якому вони були до початку запуску цього файлу.

Для цього використовуються дві команди: SETLOCAL і ENDLOCAL. Команда SETLOCAL визначає початок області локальних установок змінних

середовища. Команда `ENDLOCAL` відновлює колишні значення змінних середовища.

Є можливість розглядати значення змінних середовища як числа і проводити з ними арифметичні обчислення. Для цього використовується команда `SET` з ключем `/A`.

Наведемо приклад пакетного файлу, який складає два числа, що задані як параметри командного рядка, і виводить отриману суму на екран (лістинг 3.20.).

```
script.cmd
@ECHO OFF
REM The variable M will store the sum
SET /A M=%1+%2
ECHO The sum of %1 and %2 is equal to %M%
REM Let us to remove the variable
SET M=

D:\>script.cmd 5 5
The sum of 5 and 5 is equal to 10
D:\>script.cmd 32 32
The sum of 32 and 32 is equal to 64
D:\>script.cmd 32000000 32135791
The sum of 32000000 and 32135791 is equal to 64135791
```

Лістинг 3.20. — Командний файл для підсумовування перших двох параметрів командного рядка

При цьому в командних файлах доступні наступні арифметичні оператори, наведені в табл. 3.4.

Таблиця 3.4. — Допустимі оператори для виконання дій з арифметичними змінними

Оператор	Дія
+	Сума
—	Віднімання
*	Множення

Закінчення таблиці 3.4.

/	Ділення
%	Знаходження залишку від ділення

Домашнє завдання до Лекції 3.

Привести текст командного файлу, який буде виконувати наступне завдання: виводити в файл, що вказаний в якості першого параметра командного рядка, і на екран, поточну дату і час у форматі «xx день xx місяць, xx година xx хвилина».