

Лабораторная работа №3

Задание 1-3

Ознакомление с PowerShell выполнено.



Выбрана в силу удобства стандартная тема PowerShell.

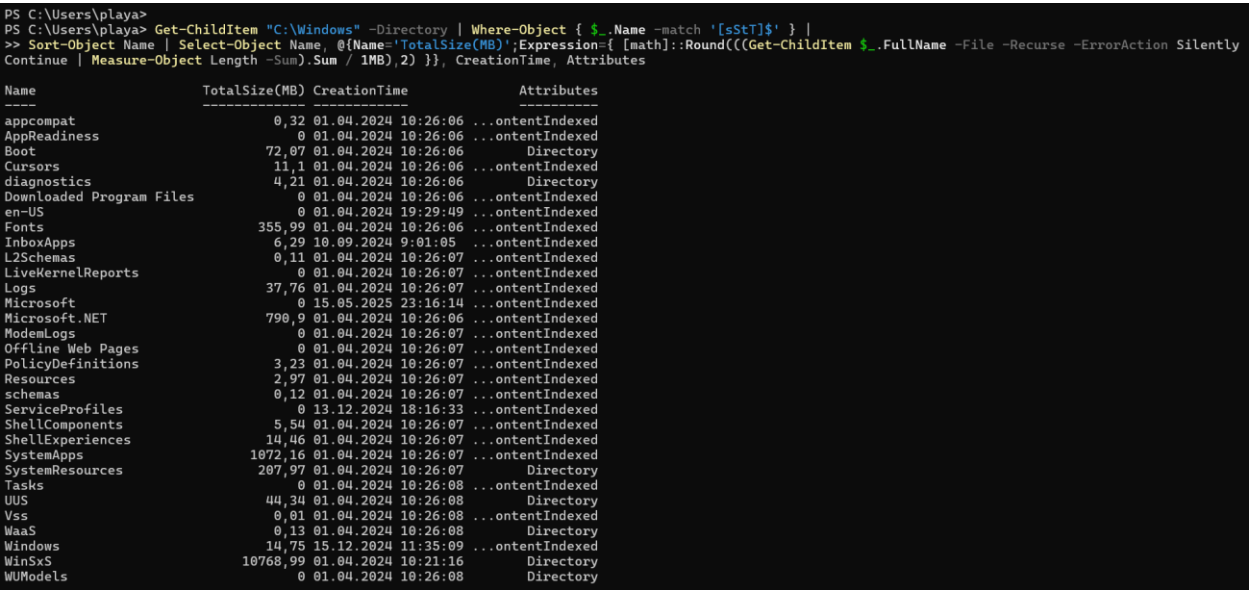
Задание 4

3, 8	Только подкаталоги	Именам	Последняя буква имени S или T
------	--------------------	--------	-------------------------------

По условию варианта необходимо отобразить только подкаталоги каталога C:\Windows, отсортировать их по именам. Дополнительно нужно, чтобы последняя буква имени папки была S или T, а также вывести для каждой папки её имя, размер в мегабайтах, дату создания и атрибуты.

Для вывода на экран использована следующая команда:

```
Get-ChildItem "C:\Windows" -Directory | Where-Object { $_.Name -match '[sStT]$' } |
Sort-Object Name | Select-Object Name, @{Name='TotalSize(MB)';Expression={
[math]::Round(((Get-ChildItem $_.FullName -File -Recurse -ErrorAction
SilentlyContinue | Measure-Object Length -Sum).Sum / 1MB),2) }}, CreationTime,
Attributes
```



Команда для вывода в текстовый файл:

```
Get-ChildItem "C:\Windows" -Directory | Where-Object { $_.Name -match '[sStT]$' } |  
Sort-Object Name | Select-Object Name,  
@{Name='Size(MB)';Expression={[math]::Round(((Get-ChildItem $_.FullName -File -  
Recurse -ErrorAction SilentlyContinue | Measure-Object Length -Sum).Sum /  
1MB),2)}} , CreationTime, Attributes | Out-File "D:\OSST Labs\Lab3\task4.txt" -Encoding  
UTF8
```

Name	Size(MB)	CreationTime	Attributes
appcompat	0,32	01.04.2024 10:26:06	NotContentIndexed
AppReadiness	0	01.04.2024 10:26:06	NotContentIndexed
Boot	72,07	01.04.2024 10:26:06	Directory
Cursors	11,1	01.04.2024 10:26:06	NotContentIndexed
diagnostics	4,21	01.04.2024 10:26:06	Directory
Downloaded Program Files	0	01.04.2024 10:26:06	NotContentIndexed
en-US	0	01.04.2024 19:29:49	NotContentIndexed
Fonts	355,99	01.04.2024 10:26:06	NotContentIndexed
InboxApps	6,29	10.09.2024 9:01:05	NotContentIndexed
L2Schemas	0,11	01.04.2024 10:26:07	NotContentIndexed
LiveKernelReports	0	01.04.2024 10:26:07	NotContentIndexed
Logs	37,76	01.04.2024 10:26:07	NotContentIndexed
Microsoft	0	15.05.2025 23:16:14	NotContentIndexed
Microsoft.NET	790,9	01.04.2024 10:26:06	NotContentIndexed
ModemLogs	0	01.04.2024 10:26:07	NotContentIndexed
Offline Web Pages	0	01.04.2024 10:26:07	NotContentIndexed
PolicyDefinitions	3,23	01.04.2024 10:26:07	NotContentIndexed
Resources	2,97	01.04.2024 10:26:07	NotContentIndexed
schemas	0,12	01.04.2024 10:26:07	NotContentIndexed
ServiceProfiles	0	13.12.2024 18:16:33	NotContentIndexed
ShellComponents	5,54	01.04.2024 10:26:07	NotContentIndexed
ShellExperiences	14,46	01.04.2024 10:26:07	NotContentIndexed
SystemApps	1072,16	01.04.2024 10:26:07	NotContentIndexed
SystemResources	207,97	01.04.2024 10:26:07	Directory
Tasks	0	01.04.2024 10:26:08	NotContentIndexed
UUS	44,34	01.04.2024 10:26:08	Directory
Vss	0,01	01.04.2024 10:26:08	NotContentIndexed
WaaS	0,13	01.04.2024 10:26:08	Directory
Windows	14,75	15.12.2024 11:35:09	NotContentIndexed
WinSxS	10768,99	01.04.2024 10:21:16	Directory
WUDrivers	0	01.04.2024 10:26:08	Directory

Задание 5

В этом задании необходимо вывести список всех свойств процессов, которые возвращает командлет Get-Process, и подсчитать их общее количество.

Результат сохраняется в текстовый файл task5.txt, а количество свойств выводится на экран.

```
PS C:\Users\playa> $procProps = (Get-Process | Get-Member -MemberType Proper  
ty).Name  
PS C:\Users\playa> $procProps | Out-File "D:\OSST Labs\Lab3\task5.txt" -Enco  
ding UTF8  
PS C:\Users\playa> "Количество свойств процессов: $($procProps.Count)"  
Количество свойств процессов: 52
```

BasePriority
Container
EnableRaisingEvents
ExitCode
ExitTime
Handle
HandleCount
HasExited
Id

Задание 6

3, 5	Имя процесса, Id , TotalProcessorTime PriorityClass, UserprocessorTime, TotalProcessorTime	Id > 100
------	---	----------

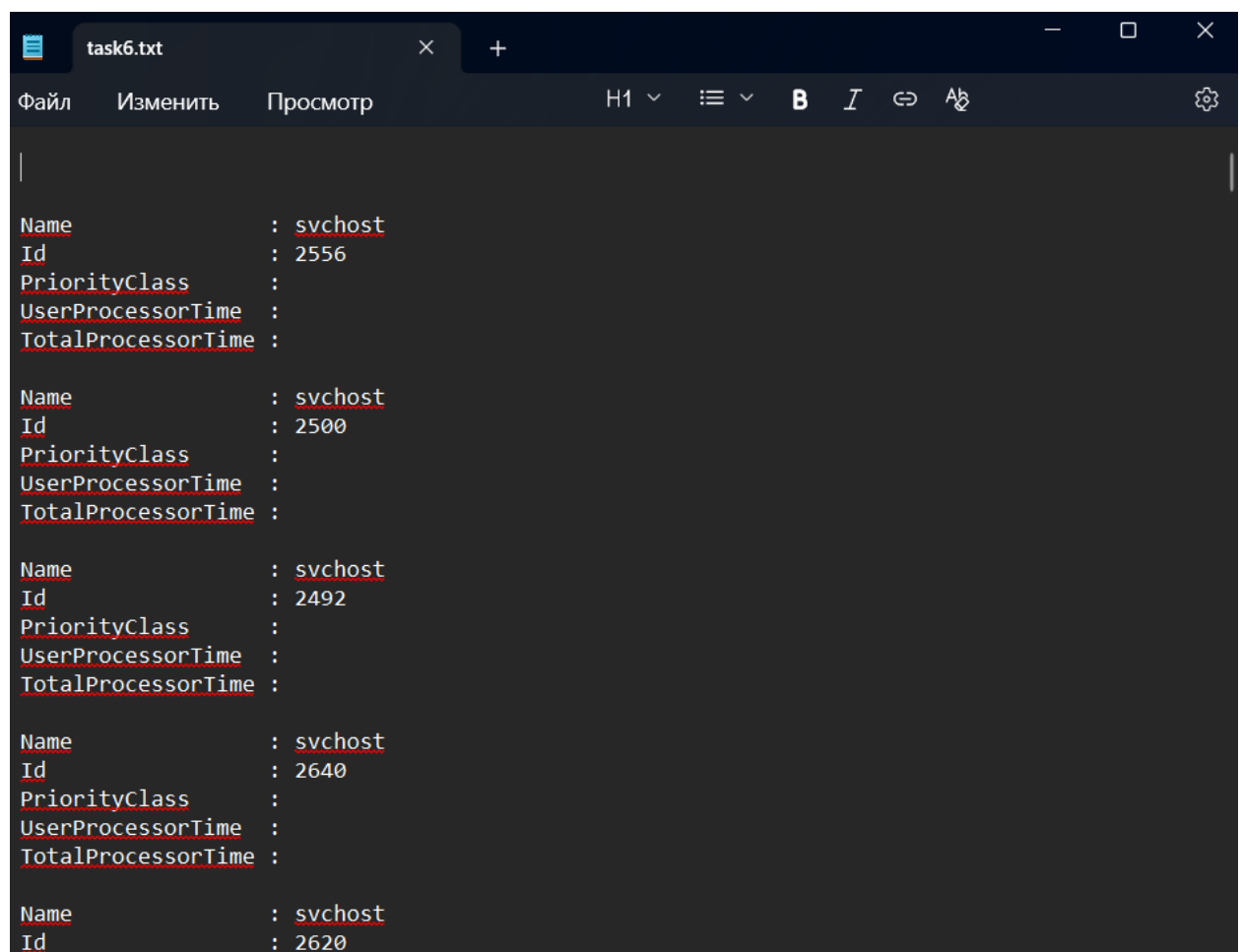
Здесь нужно создать текстовый файл, содержащий список выполняемых процессов, отсортированных по полю TotalProcessorTime, и включить только те, у которых ID > 100. Вывести следует поля: имя процесса, идентификатор, приоритет, пользовательское и общее процессорное время.

Для этого использована команда:

```
Get-Process | Where-Object { $_.Id -gt 100 } | Sort-Object TotalProcessorTime | Select-Object Name, Id, PriorityClass, UserProcessorTime, TotalProcessorTime | Out-File "D:\OSST Labs\Lab3\task6.txt" -Encoding UTF8
```

Результат сохраняется в файл task6.txt.

```
PS C:\Users\playa> Get-Process -ErrorAction SilentlyContinue |  
>> Where-Object { $_.Id -gt 100 } |  
>> Sort-Object TotalProcessorTime -ErrorAction SilentlyContinue |  
>> Select-Object Name, Id, PriorityClass, UserProcessorTime, TotalProcessorTime |  
>> Out-File "D:\OSST Labs\Lab3\task6.txt" -Encoding UTF8
```



```
task6.txt
Файл  Изменить  Просмотр  H1  B  I  A
Name      : svchost
Id         : 2556
PriorityClass :
UserProcessorTime :
TotalProcessorTime :
Name      : svchost
Id         : 2500
PriorityClass :
UserProcessorTime :
TotalProcessorTime :
Name      : svchost
Id         : 2492
PriorityClass :
UserProcessorTime :
TotalProcessorTime :
Name      : svchost
Id         : 2640
PriorityClass :
UserProcessorTime :
TotalProcessorTime :
Name      : svchost
Id         : 2620
PriorityClass :
UserProcessorTime :
TotalProcessorTime :
```

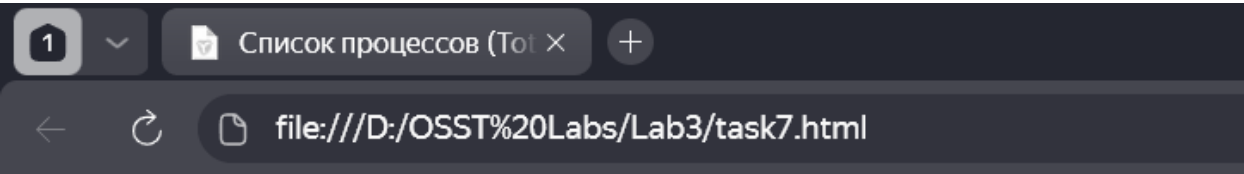
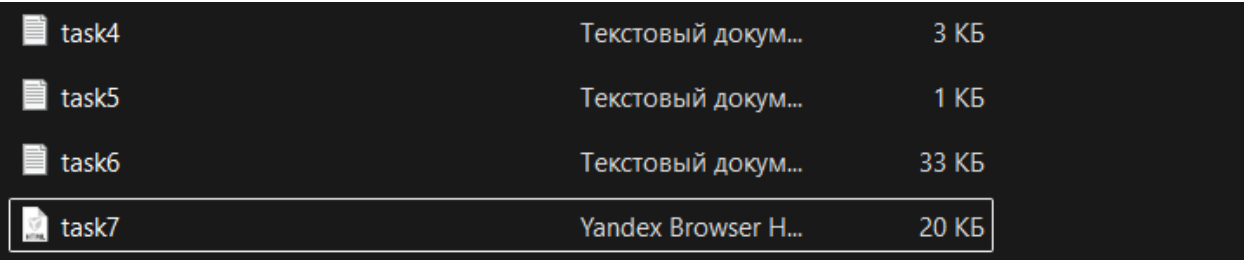
Задание 7

В этом задании нужно создать HTML-файл, в котором будет представлен список процессов, отсортированных по времени процессора (TotalProcessorTime).

Выполнена команда:

```
Get-Process | Sort-Object TotalProcessorTime | Select-Object Name, Id, PriorityClass,
UserProcessorTime, TotalProcessorTime | ConvertTo-Html -Title "Список процессов
(TotalProcessorTime)" | Out-File "D:\OSST Labs\Lab3\task7.html" -Encoding UTF8
Invoke-Item "D:\OSST Labs\Lab3\task7.html"
```

```
PS C:\Users\playa> Invoke-Item "D:\OSST Labs\Lab3\task7.html"
```



Name	Id	PriorityClass	UserProcessorTime	TotalProcessorTime
svchost	2556			
svchost	2500			
svchost	2492			
svchost	2640			
svchost	2620			
svchost	2612			
svchost	2160			
svchost	2124			
svchost	2020			
svchost	2484			
svchost	2440			
svchost	2432			
svchost	2668			
svchost	3188			
svchost	3176			
svchost	3108			
svchost	3564			
svchost	3548			
svchost	3332			

В результате создаётся HTML-файл, который открывается в браузере в виде таблицы со всеми отсортированными процессами.

Задание 8

Необходимо вычислить суммарный объем всех графических файлов форматов .bmp и .jpg, находящихся в каталоге C:\Windows и его подкаталогах.

Для выполнения командлет Measure-Object, который суммирует их размеры.

```
PS C:\Users\playa> $images = Get-ChildItem "C:\Windows" -Include *.bmp, *.jpg -File -Recurse -ErrorAction SilentlyContinue
PS C:\Users\playa> $totalSize = ($images | Measure-Object Length -Sum).Sum
PS C:\Users\playa> "Общий размер графических файлов: {0:N2} МБ" -f ($totalSize / 1MB)
Общий размер графических файлов: 88,27 МБ
```

Задание 9

В этом задании нужно вывести характеристики процессора компьютера

```
PS C:\Users\playa> Get-CimInstance Win32_Processor |
>> Select-Object Name, Manufacturer, NumberOfCores, NumberOfLogicalProcessors,
MaxClockSpeed

Name                : Intel(R) Core(TM) i5-7300HQ CPU @ 2.50GHz
Manufacturer       : GenuineIntel
NumberOfCores       : 4
NumberOfLogicalProcessors : 4
MaxClockSpeed       : 2496
```

Задание 10

Здесь нужно измерить время выполнения двух командлетов — dir и ps. Для каждой команды измерения выполняются 5 раз, после чего рассчитываются минимальное, максимальное и среднее значения.

Для dir:

```
PS C:\Users\playa> $dirTimes = 1..5 | ForEach-Object { (Measure-Command { Get-ChildItem C:\Windows }).TotalMilliseconds }
PS C:\Users\playa> "Результаты для Get-ChildItem (ms):"
Результаты для Get-ChildItem (ms):
PS C:\Users\playa> $dirTimes
4,1826
10,2862
3,8248
3,3237
3,2593
PS C:\Users\playa> "Минимальное: {0:N2}, Максимальное: {1:N2}, Среднее: {2:N2}" -f ($dirTimes | Measure-Object -Minimum).Minimum, ($dirTimes | Measure-Object -Maximum).Maximum, ($dirTimes | Measure-Object -Average).Average
Минимальное: 3,26, Максимальное: 10,29, Среднее: 4,98
```

Для ps:

```

PS C:\Users\playa> $psTimes = 1..5 | ForEach-Object { (Measure-Command { Get-Pro
cess }).TotalMilliseconds }
PS C:\Users\playa> "Результаты для Get-Process (ms):"
Результаты для Get-Process (ms):
PS C:\Users\playa> $psTimes
3,2975
2,4172
9,5396
2,4451
2,2956
PS C:\Users\playa> "Минимальное: {0:N2}, Максимальное: {1:N2}, Среднее: {2:N2}"
-f ($psTimes | Measure-Object -Minimum).Minimum, ($psTimes | Measure-Object -Max
imum).Maximum, ($psTimes | Measure-Object -Average).Average
Минимальное: 2,30, Максимальное: 9,54, Среднее: 4,00

```

Задание 11

Необходимо написать две функции для работы с файлом nn.txt, в котором находятся числовые данные.

Первая функция подсчитывает количество положительных чисел, а вторая — количество чётных чисел.

Вывод кол-ва положительных чисел:

```

PS C:\Users\playa> $numbers = Get-Content "D:\OSST Labs\Lab3\nn.txt" | ForEach-Ob
ject { [int]$_ }
PS C:\Users\playa> $positiveCount = ($numbers | Where-Object { $_ -gt 0 }).Count
PS C:\Users\playa> Write-Host "Положительных чисел: $positiveCount"
Положительных чисел: 5

```

Вывод кол-ва четных чисел:

```

PS C:\Users\playa> $evenCount = ($numbers | Where-Object { $_ % 2 -eq 0 }).Count
PS C:\Users\playa> Write-Host "Чётных чисел: $evenCount"
Чётных чисел: 4

```