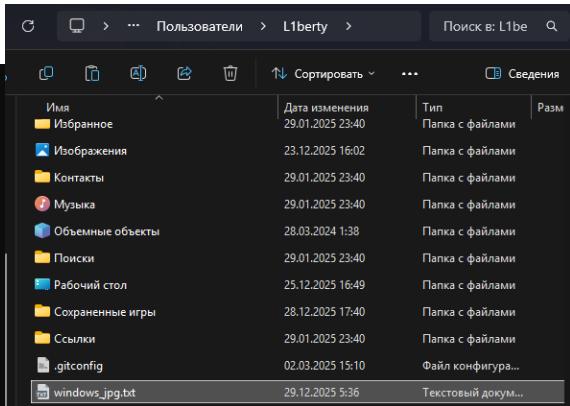


#### 4 задание

**Get-ChildItem** – команда для получения содержимого каталога

**Filter \*.jpg** – фильтр выбирает те файлы, которые заканчиваются .jpg

5, 10	Только файлы jpg	Именам	Любые
-------	------------------	--------	-------



```
PS C:\Users\Llberty> Get-ChildItem C:\Windows -Recurse -Filter *.jpg -ExecutionPolicy Bypass | Sort-Object Name | Tee-Object -FilePath windows.jpg.txt

Katalog: C:\Windows\Microsoft.NET\Framework\v4.0.30319\ASP.NET\Web-INF\Files\Images

Mode                LastWriteTime         Length Name
----              -----        -----    -----   -----
-a--       01.04.2020 10:22           22580 ASHdotNET_Logo.jpg

Katalog: C:\Windows\WindowsImageCache\WindowsWebServer\Images

Mode                LastWriteTime         Length Name
----              -----        -----    -----   -----
-a--       01.04.2020 10:22           22580 ASHdotNET_Logo.jpg

Katalog: C:\Windows\WindowsImageCache\WindowsWebServer\Images\b49f9f7f11d9ba3a_4.0.15912.0_none_7c94bdeecbfcbf2

Mode                LastWriteTime         Length Name
----              -----        -----    -----   -----
-a--       01.04.2020 10:22           22580 ASHdotNET_Logo.jpg

Katalog: C:\Windows\WindowsImageCache\WindowsWebServer\Images\b49f9f7f11d9ba3a_4.0.15912.0_none_34e75587b70986dc

Mode                LastWriteTime         Length Name
----              -----        -----    -----   -----
-a--       01.04.2020 10:22           22580 ASHdotNET_Logo.jpg

Katalog: C:\Windows\Microsoft.NET\Framework\v4.0.30319\ASP.NET\Web-INF\Files\Images

Mode                LastWriteTime         Length Name
----              -----        -----    -----   -----
-a--       01.04.2020 10:24           22580 ASHdotNET_Logo.jpg

Katalog: C:\Windows\WindowsImageCache\WindowsExperience-Optimizer_31bf38566d36ee35_19.0.26108.7619_none_fb2b0ff2158405d\Photon\Pub\Public\Experience\Account\assets\assets\Images

Mode                LastWriteTime         Length Name
----              -----        -----    -----   -----
-a--       14.09.2020 2:16            38562 copilot-upsell.jpg
```

#### 5 задание

**Get-Process** выводит список выполняемых процессов

**Out-File** сохраняет результат в текстовый файл

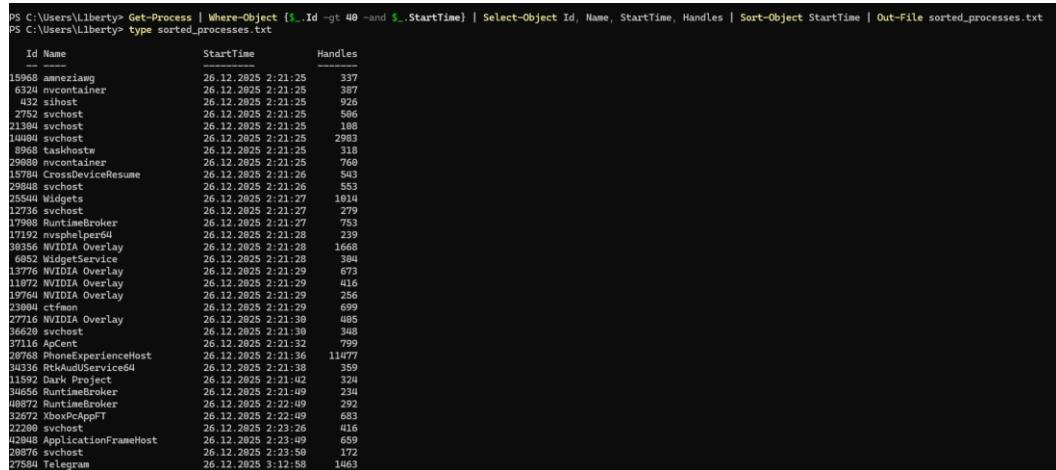
**Count** подсчет количества процессов

```
PS C:\Users\Llberty> Get-Process | Sort-Object Name | Out-File processes.txt
PS C:\Users\Llberty> (Get-Process).Count
255
PS C:\Users\Llberty> |
```

#### 6 задание

Создаем файл **sorted\_processes**, в котором содержится список всех процессов id которых больше 40, упорядоченный по времени старта

2, 6	Id,	Имя	процесса,	Время старта	Id > 40
------	-----	-----	-----------	--------------	---------



```
PS C:\Users\Llberty> Get-Process | Where-Object {$_.'Id' -gt 40 -and $_.'StartTime'} | Select-Object Id, Name, StartTime, Handles | Sort-Object StartTime | Out-File sorted_processes.txt
PS C:\Users\Llberty> type sorted_processes.txt
```

Id	Name	StartTime	Handles
15968	amneziaang	26.12.2025 2:21:25	337
6324	nvicontainer	26.12.2025 2:21:25	387
432	sihost	26.12.2025 2:21:25	926
2752	svchost	26.12.2025 2:21:25	506
1394	svchost	26.12.2025 2:21:25	188
1600	svchost	26.12.2025 2:21:25	2993
8968	taskhostw	26.12.2025 2:21:25	318
29880	nvicontainer	26.12.2025 2:21:25	760
15784	CrossDeviceResume	26.12.2025 2:21:26	543
29848	svchost	26.12.2025 2:21:26	553
25541	Widget	26.12.2025 2:21:27	1014
17730	svchost	26.12.2025 2:21:27	279
17731	WindowsFeedbackService	26.12.2025 2:21:27	763
17192	wpvphelper64	26.12.2025 2:21:28	239
38356	NVIDIA Overlay	26.12.2025 2:21:28	1668
6082	WidgetService	26.12.2025 2:21:28	384
13776	NVIDIA Overlay	26.12.2025 2:21:29	673
11972	NVIDIA Overlay	26.12.2025 2:21:29	416
19764	NVIDIA Overlay	26.12.2025 2:21:29	256
30120	NVIDIA Overlay	26.12.2025 2:21:29	599
27716	NVIDIA Overlay	26.12.2025 2:21:30	485
36620	svchost	26.12.2025 2:21:30	348
37116	ApCent	26.12.2025 2:21:32	799
28768	PhoneExperienceHost	26.12.2025 2:21:36	11477
34336	RtkAudiService64	26.12.2025 2:21:38	359
13592	Dark Project	26.12.2025 2:21:40	324
10000	WindowsFeedback	26.12.2025 2:21:40	231
48877	RuntimeBroker	26.12.2025 2:22:09	292
32672	xboxP2AppPT	26.12.2025 2:22:49	683
22208	svchost	26.12.2025 2:23:26	416
42848	ApplicationFrameHost	26.12.2025 2:23:49	659
28876	svchost	26.12.2025 2:23:59	172
27584	Telegram	26.12.2025 3:12:58	1463

## 7 задание

Делаем все тоже самое то и в 6 задании, только добавляем **ConvertTo-HTML** и **Invoke-Item** для создания и открытия html

```
PS C:\Users\Liberty> Get-Process | Where-Object {$_ .Id -gt 40 -and $_ .StartTime} | Select-Object Id, Name, StartTime, Handles | Sort-Object StartTime | ConvertTo-HTML > processes.html
PS C:\Users\Liberty> Invoke-Item processes.html
PS C:\Users\Liberty>
```

<b>Id</b>	<b>Name</b>	<b>StartTime</b>	<b>Handles</b>
15968	amneziawg	26.12.2025 2:21:25	337
6324	nvcontainer	26.12.2025 2:21:25	387
432	sihost	26.12.2025 2:21:25	931
2752	svchost	26.12.2025 2:21:25	508
21304	svchost	26.12.2025 2:21:25	108
14404	svchost	26.12.2025 2:21:25	2999
8968	taskhostw	26.12.2025 2:21:25	316
29080	nvcontainer	26.12.2025 2:21:25	760
15784	CrossDeviceResume	26.12.2025 2:21:26	543

## 8 задание

Находим суммарный объем всех файлов **bmp, jpg** в каталоге windows и его подкаталогах

Идем по всем подкаталогам с помощью **-Recurse**

**-Sum** суммирует объем

Что суммировать задаем с помощью **-Property Length**

**-ErrorAction SilentlyContinue** скрывает ошибку доступа к папкам

```
PS C:\Users\Liberty> Get-ChildItem C:\Windows -Recurse -Include *.bmp, *.jpg -ErrorAction SilentlyContinue | Measure-Object Length -Sum
Count : 190
Average :
Sum   : 104351913
```

## 9 задание

**Get-CimInstance Win32\_Processor** – получает информацию о процессоре через **CIM**

```
PS C:\Users\Liberty> Get-CimInstance Win32_Processor
DeviceID Name                                     Caption                               MaxClockSpeed SocketDesignation Manufacturer
----- --                                     --                               --          --          --
CPU0    11th Gen Intel(R) Core(TM) i5-11400F @ 2.60GHz Intel64 Family 6 Model 167 Stepping 1 2592           U3E1           GenuineIntel
```

## 10 задание

1...10 – последовательность чисел от 1 до 10.

Время измеряем с помощью **measure-command {dir}** в миллисекундах (**TotalMilliseconds**)

С помощью **Measure-Object – Maximum –Minimum – Average** вычисляем статистику и выводим на экран

**dir**

```
PS C:\Users\Lliberty> 1..10 | ForEach-Object { Measure-Command { dir } } | Measure-Object TotalMilliseconds -Maximum -Minimum -Average

Count    : 10
Average  : 2,25683
Sum      :
Maximum  : 10,5692
Minimum  : 0,6618
Property : TotalMilliseconds
```

**ps**

```
PS C:\Users\Lliberty> 1..10 | ForEach-Object { Measure-Command { Get-Process } } | Measure-Object TotalMilliseconds -Maximum -Minimum -Average

Count    : 10
Average  : 5,62276
Sum      :
Maximum  : 13,8787
Minimum  : 3,6082
Property : TotalMilliseconds
```