# Moduł pamięci PowerShell: inicjowanie, partycjonowanie, formatowanie, sprawdzanie użycia dysku i zmiana rozmiaru dysku

# Przygotowanie:

- 1. Wyłącz maszynę wirtualną
- 2. Przywróć punkt kontrolny
- 3. Pobierz pliki 2mbr.vhdx, 3.vhdx, 4.vhdx
- 4. Podłącz jak poniżej

2r1_11 ~	<ul> <li>↓</li> <li>♥</li> </ul>
<ul> <li>Sprzęt</li> <li>Dodaj sprzęt</li> <li>Oprogramowanie układowe Istnieją oczekujące zmiany wpisów</li> <li>Zabezpieczenia Włączono moduł TPM</li> <li>Pamięć</li> </ul>	<ul> <li>Kontroler SCSI</li> <li>Możesz dodać dyski twarde i stacje dysków CD/DVD do kontrolera SCSI lub usunąć kontroler SCSI z maszyny wirtualnej.</li> <li>Wybierz typ dysku, który chcesz dołączyć do kontrolera, a następnie kliknij przycisk Dodaj.</li> <li>Dysk twardy</li> <li>Stacja dysków DVD</li> </ul>
12144 MB     Procesor     Procesory wirtualne: 2     Sontroler SCSI	Dysk udostępniony
Dysk twardy r11_22H2-disk001_B1953334 Dysk twardy	Dodaj
<ul> <li>2mbr.vhdx</li> <li>Dysk twardy</li> <li>3.vhdx</li> <li>Dysk twardy</li> <li>4.vhdx</li> </ul>	Po podłączeniu dysku do kontrolera można skonfigurować dysk twardy do używania wirtualnego dysku twardego lub fizycznego dysku twardego. Aby usunąć kontroler SCSI z tej maszyny wirtualnej, kliknij przycisk Usuń. Wszystkie wirtualne dyski twarde dołączone do tego kontrolera zostaną usunięte, ale ich ustawienia pozostaną dostępne.

#### Zawartość

- 1. Przeglądaj informacje o dysku i partycjach za pomocą Get-Disk
- 2. Zainicjuj dysk za pomocą Initialize-Disk
- 3. Partycjonuj dysk za pomocą nowej partycji
- 4. Sformatuj dysk za pomocą Format-Volume
- 5. Sformatuj pamięć USB
- 6. Uzyskaj informacje o wykorzystaniu dysku za pomocą Get-CimInstance
- 7. Zmień rozmiar operacji na dysku za pomocą Resize-Partition

Windows 11 mają moduł pamięci w PowerShell, który oferuje różne polecenia cmdlet związane z zarządzaniem pamięcią dyskową. Aby zobaczyć listę wszystkich poleceń cmdlet, użyj:

Get-Command -Module Storage

PS C:\Windows	\system32> Get-Command -Module Storage		
CommandType	Name	Version	Source
Alias	 Disable-PhysicalDiskIndication	2.0.0.0	Storage
Alias	Disable-StorageDiagnosticLog	2.0.0.0	Storage
Alias	Enable-PhysicalDiskIndication	2.0.0.0	Storage
Alias	Enable-StorageDiagnosticLog	2.0.0.0	Storage
Alias	Flush-Volume	2.0.0.0	Storage
Alias	Get-DiskSNV	2.0.0.0	Storage
Alias	Get-PhysicalDiskSNV	2.0.0.0	Storage
Alias	Get-StorageEnclosureSNV	2.0.0.0	Storage
Alias	Initialize-Volume	2.0.0.0	Storage
Alias	Write-FileSystemCache	2.0.0.0	Storage
Function	Add-InitiatorIdToMaskingSet	2.0.0.0	Storage
Function	Add-PartitionAccessPath	2.0.0.0	Storage
Function	Add-PhysicalDisk	2.0.0.0	Storage
Function	Add-StorageFaultDomain	2.0.0.0	Storage
Function	Add-TargetPortToMaskingSet	2.0.0.0	Storage
Function	Add-VirtualDiskToMaskingSet	2.0.0.0	Storage
Function	Block-FileShareAccess	2.0.0.0	Storage
Function	Clear-Disk	2.0.0.0	Storage
Function	Clear-FileStorageTier	2.0.0.0	Storage
Function	Clear-StorageDiagnosticInfo	2.0.0.0	Storage

# 1. Przeglądanie poleceń cmdlet z modułu pamięci

Widać, że lista jest naprawdę długa. Możesz użyć parametrów -Verb lub -Noun z tym poleceniem, aby dalej przyciąć listę, jak pokazano w następującym poleceniu:

#### Get-Command -Module Storage -Noun "\*disk\*"

PS C:\Windows\	<pre>system32&gt; Get-Command -Module Storage -Noun "</pre>	*disk*"	
CommandType	Name	Version	Source
Alias	Disable-PhysicalDiskIndication	2.0.0.0	Storage
Alias	Enable-PhysicalDiskIndication	2.0.0.0	Storage
Alias	Get-DiskSNV	2.0.0.0	Storage
Alias	Get-PhysicalDiskSNV	2.0.0.0	Storage
Function	Add-PhysicalDisk	2.0.0.0	Storage
Function	Add-VirtualDiskToMaskingSet	2.0.0.0	Storage
Function	Clear-Disk	2.0.0.0	Storage
Function	Connect-VirtualDisk	2.0.0.0	Storage
Function	Disable-PhysicalDiskIdentification	2.0.0.0	Storage
Function	Disconnect-VirtualDisk	2.0.0.0	Storage
Function	Dismount-DiskImage	2.0.0.0	Storage
Function	Enable-PhysicalDiskIdentification	2.0.0.0	Storage
Function	Get-Disk	2.0.0.0	Storage
Function	Get-DiskImage	2.0.0.0	Storage
Function	Get-DiskStorageNodeView	2.0.0.0	Storage
Function	Get-PhysicalDisk	2.0.0.0	Storage
Function	Get-PhysicalDiskStorageNodeView	2.0.0.0	Storage
Function	Get-VirtualDisk	2.0.0.0	Storage
Function	Get-VirtualDiskSupportedSize	2.0.0.0	Storage
Function	Hide-VirtualDisk	2.0.0.0	Storage
Function	Initialize-Disk	2.0.0.0	Storage
Function	Mount-DiskImage	2.0.0.0	Storage
Function	New-StorageSubsystemVirtualDisk	2.0.0.0	Storage

Przeglądanie cmdletów z modułu pamięci za pomocą rzeczownika dysku

Polecenie cmdlet Get-Command z parametrami dysku -Module Storage i -Noun daje nam wszystkie polecenia cmdlet z modułu "Storage", które mają "disk" w sekcji "Noun". Mam nadzieję, że ma to sens w zrozumieniu nomenklatury "czasownik-rzeczownik" PowerShell.

Omówienie wszystkich tych poleceń wykracza poza zakres. Postaram się opisać polecenia, których administrator najczęściej używałby do wykonywania rutynowych operacji zarządzania dyskami.

Wyświetl informacje o dysku i partycji za pomocą Get-Disk

Jak wspomniano wcześniej, schemat partycjonowania jest bardzo ważny, jeśli chodzi o dyski. Aby wyświetlić schemat partycjonowania dla wszystkich dysków w systemie, możesz użyć następującego polecenia:

Get-Disk | Format-Table -Auto

PS C:\I	Windows\system32> (	Get-Disk	Form	mat-Table -Au	to			
Number	Friendly Name	Serial N	lumber	HealthStatus	OperationalStatus	Total Si	ize	Partition Style
0	Msft Virtual Disk			Healthy	Online	1	ΤВ	GPT I
1	Msft Virtual Disk			Healthy	Online	15	GB	MBR
2	Msft Virtual Disk			Healthy	Online	470	GB	GPT
3	Msft Virtual Disk			Healthy	Online	50	GB	RAW

Przeglądanie schematu partycjonowania za pomocą Get Disk

Polecenie wyświetla informacje o trzech dyskach podłączonych do systemu.

Kolumna Styl partycji na końcu pokazuje schemat partycjonowania dla wszystkich dysków.

Moduł magazynu programu PowerShell oferuje również polecenie cmdlet Get-PhysicalDisk . Przyjrzyjmy się wynikom tego polecenia:

### Get-PhysicalDisk

PS C:∖I	windows\system32> (	Get-PhysicalDi	isk					
Number	FriendlyName	SerialNumber	MediaType	CanPool	OperationalStatus	HealthStatus	Usage	Size
0	Msft Virtual Disk		Unspecified	False	OK	Healthy	Auto-Select	1 TB
2	Msft Virtual Disk		Unspecified	False	OK	Healthy	Auto-Select	470 GB
1	Msft Virtual Disk		Unspecified	False	OK	Healthy	Auto-Select	15 GB
3	Msft Virtual Disk		Unspecified	True	OK	Healthy	Auto-Select	50 GB

Wyświetlanie dysków fizycznych za pomocą funkcji Get Physical Disk

Jaka jest więc różnica między poleceniami Get-Disk i Get-PhysicalDisk?

Polecenie Get-Disk pokazuje tylko dyski logiczne, podczas gdy polecenie Get-PhysicalDisk zwraca wszystkie fizyczne urządzenia dysków podłączone do systemu. Polecenie Get-PhysicalDisk jest szczególnie przydatne podczas tworzenia przestrzeni dyskowych i zarządzania nimi. Nie będziemy omawiać przestrzeni do przechowywania. Skupmy się tylko na zarządzaniu dyskami.

Aby wyświetlić informacje o partycjach dla dysku, użyj polecenia Get-Partition, jak pokazano poniżej:

# Get-Partition -DiskNumber 0

PS C:\Windows\sy	/stem32> Get-	Partition -DiskNumber 0	
DiskPath: \\?	∕\scsi#disk&v	en_msft∏_virtual_disk#5&14f89abb&0&000	00#{53f56307-b6bf-11d0-94f2-00a0c91efb8b}
PartitionNumber	DriveLetter	Offset	Size Type
1 2 3 4	C	1048576 105906176 122683392 1098693738496	100 MB System 16 MB Reserved 1023.12 GB Basic 778 MB Recovery

Przeglądanie informacji o partycjach dysku za pomocą funkcji Get Partition

Dane wyjściowe polecenia Get-Disk pokazują numer dysku w pierwszej kolumnie pod kolumną "Numer".

Użyliśmy polecenia Get-Partition z parametrem -DiskNumber , aby wyświetlić szczegóły partycji dla dysku 0.

Dane wyjściowe Get-Partition dały nam DriveLetter dla partycji dostępnych na tym dysku.

Aby uzyskać informacje o konkretnej partycji na podstawie jej litery dysku, możesz użyć polecenia Get-Partition z parametrem -DriveLetter .

Get-Partition -I	DriveLetter	<mark>E</mark>		
PS C:\Windows\sy	stem32> Get-N	Partition -DriveLette	er E	
DiskPath: \\?	\scsi#disk&ve	en_msft∏_virtual_	_disk#5&14f89abb&0&00002#{53f5630	7-b6bf-11d0-94f2-00a0c91efb8b}
PartitionNumber	DriveLetter	Offset	Size	Туре
2	E	16777216	469.98 GB	Basic

Przeglądanie informacji o partycji przy użyciu litery dysku

Istnieje wiele poleceń, których możesz użyć, aby uzyskać informacje, ale nie mogę ich wszystkich tutaj omówić. Teraz, gdy mamy już wystarczająco dużo informacji o dyskach, przejdźmy do następnej sekcji.

# 2. Zainicjuj dysk za pomocą Initialize-Disk

Kiedy podłączasz nowy dysk do systemu po raz pierwszy, musisz go zainicjować, zanim zaczniesz z niego korzystać. Pamiętasz następujący ekran w swoim narzędziu do <mark>zarządzania dyskami</mark>?

Zarządzanie komputerem > Zarządzanie dyskami > Anuluj



Dysk 3 to dysk, który nie został zainicjowany w Zarządzaniu dyskami

Użyłem polecenia Get-Disk, aby wyświetlić listę wszystkich dysków. Kolumna Styl partycji informuje mnie, że dysk, który nie został zainicjowany, jest oznaczony jako RAW.

Zainicjuj surowe dyski

Aby zainicjować dysk za pomocą PowerShell, możemy użyć Initialize-Disk z parametrem -PartitionStyle, jak pokazano w poniższym poleceniu:

Get-Disk | where PartitionStyle -eq 'raw' | Initialize-Disk -PartitionStyle GPT

#### PS C:\Windows\system32> Get-Disk | where PartitionStyle -eq 'raw' | Initialize-Disk -PartitionStyle GPT PS C:\Windows\system32> \_

Efekt sprawdź w zarządzaniu dyskami:



Uruchomiłem więc ponownie Get-Disk z filtrem, aby uzyskać tylko dyski "RAW" i przesłałem wynik do polecenia Initialize-Disk . Parametr -PartitionStyle GPT zainicjował dysk jako GPT.

Aby zainicjować dysk jako MBR, możesz zastąpić GPT MBR.

Wyświetl informacje o dysku i partycji za pomocą Get-Disk

Aby wyświetlić schemat partycjonowania dla wszystkich dysków w systemie, użyj następującego polecenia:

Get-Disk | Format-Table -Auto

PS C:\I	Windows\system32>	Get-Disk	: For	mat-Table -Au	to			
Number	Friendly Name	Serial	Number	HealthStatus	OperationalStatus	Total Si	ze	Partition Style
0	Msft Virtual Disk			Healthy	Online	1	ΤВ	GPT
1	Msft Virtual Disk			Healthy	Online	15 (	GB	MBR
2	Msft Virtual Disk			Healthy	Online	470	GB	GPT
3	Msft Virtual Disk			Healthy	Online	50 (	GB	GPT

#### 3. Partycjonuj dysk za pomocą nowej partycji

Teraz, gdy nasz dysk jest zainicjowany, musimy utworzyć partycję (partycje), zanim system Windows zacznie z niego korzystać. Aby utworzyć partycje, możemy użyć polecenia cmdlet New-Partition, jak pokazano poniżej:

New-Partition -DiskNumber 3 -Size 20GB -AssignDriveLetter

PS C:\Windows\sy	stem32> New-F	Partition -DiskNumber 3 -Size	20GB -AssignDriveLetter	
DiskPath: \\?	\scsi#disk&ve	en_msft∏_virtual_disk#5&14	1f89abb&0&00003#{53f5630	7-b6bf-11d0-94f2-00a0c91efb8b}
PartitionNumber	DriveLetter	Offset	Size	Туре
2	F 🛹	16777216	 20 GB	Basic

Tworzenie nowej partycji z automatyczną literą dysku

Parametr -DiskNumber służy do określenia konkretnego dysku.

Parametr -Size służy do określenia rozmiaru partycji. Aby wykorzystać całe nieprzydzielone miejsce dostępne na dysku, możesz użyć parametru -UseMaximumSize .

Parametr -AssignDriveLetter umożliwia systemowi Windows automatyczne wybieranie litery dysku. Aby określić konkretną, wybraną literę dysku, możesz użyć parametru -DriveLetter .

Aby utworzyć kolejną partycję na tym samym dysku, użyję następującego polecenia:

New-Partition -	-DiskNumbe	er 3 -UseMaximumSize -	DriveLetter H	
PS C:\Windows\sy	stem32> New-F	Partition -DiskNumber 3 -Us	eMaximumSize -DriveLetter H	• •
DiskPath: \\?	\scsi#disk&ve	en_msft∏_virtual_disk#5	&14f89abb&0&00003#{53f5630	7-b6bf-11d0-94f2-00a0c91efb8b}
PartitionNumber	DriveLetter	Offset	Size	Туре
3	Н	21491613696	29.98 GB	Basic

Tworzenie nowej partycji z ręczną literą dysku

Mój dysk miał 50 GB, a pierwsza partycja, którą utworzyłem, miała 20 GB.

Parametr -UseMaximumSize dla drugiej partycji utworzył nową partycję o przybliżonym rozmiarze 30 GB z ręcznie określoną literą dysku "G".

### 4. Sformatuj dysk za pomocą Format-Volume

Zanim zaczniemy zapisywać nasze pliki na partycjach, które właśnie utworzyliśmy, musimy je sformatować. W poprzedniej sekcji przypisaliśmy litery dysków "F" i "H" naszym nowym partycjom. Aby sformatować wolumin, możemy użyć polecenia Format-Volume, jak pokazano poniżej:

Format-Volume -DriveLetter H -FileSystem NTFS -NewFileSystemLabel "DATA1"

PS C:\Window	vs\system32> A	Format-Volume -[	)riveLetter	r H -FileSyste	em NTFS -NewFileSys	stemLabel "DATA	1"-
DriveLetter	FriendlyName	FileSystemType	DriveType	HealthStatus	OperationalStatus	SizeRemaining	Size
	DATA1		Fixed	Hoalthy		20 01 CP	20 09 69
п	DATAL	NIFS	Fixed	неатспу	UK	29.91 GB	29.98 GB

Formatowanie woluminu dysku

Parametr -DriveLetter służy do określenia partycji, którą zamierzamy sformatować.

Parametr -FileSystem służy do określenia systemu plików. Można określić typy systemu plików NTFS, ReFS, exFAT, FAT32 i FAT.

Parametr -NewFileSystemLabel służy do określenia etykiety woluminu. To wszystko do inicjowania, partycjonowania i formatowania dysku. Najlepszą rzeczą w PowerShell jest jego zdolność do potokowania. Możemy połączyć wszystkie polecenia inicjowania, partycjonowania i sformatowania dysku w jedno polecenie, które będzie wyglądać tak, jak pokazano poniżej:

Get-Disk | where PartitionStyle -eq 'raw' | Initialize-Disk -PartitionStyle GPT -PassThru | New-Partition -AssignDriveLetter -UseMaximumSize | Format-Volume -FileSystem NTFS -NewFileSystemLabel "MYDATA"

Jedyną nowością, jaką zauważysz w powyższym poleceniu, jest parametr -PassThru, który przesyła dane wyjściowe polecenia Initialize-Disk w dół potoku. Parametr -PassThru jest potrzebny, ponieważ Initialize-Disk domyślnie nie generuje żadnych danych wyjściowych.

Polecenie pobiera wszystkie dyski "RAW", inicjuje je za pomocą schematu "GPT", tworzy pojedynczą partycję z parametrem -UseMaximumSize, automatycznie przypisuje literę dysku, a następnie formatuje wolumin w systemie plików NTFS.

#### Usuń wszystkie partycje (woluminy) z zasobu Dysk1

Clear-Disk -Number 1 -RemoveData -Confirm:\$false

# Zainicjuj dysk jako GPT (lub MBR, jeśli preferujesz)

Initialize-Disk -Number 1 -PartitionStyle GPT

#### Sformatuj dysk za pomocą jednego polecenia

Get-Disk | where {\$\_.DiskNumber -eq '1'} | New-Partition -AssignDriveLetter -UseMaximumSize | Format-Volume -FileSystem NTFS -NewFileSystemLabel "USBDATA"



### 5. Sformatuj pamięć USB

Aby sformatować wymienny dysk pamięci, użyj następującego polecenia:

Format-Volume -DriveLetter D -FileSystem NTFS -NewFileSystemLabel "USBDATA"

PS C:\Windows\system32> Format-Volume -DriveLetter D -FileSystem NTFS -NewFileSystemLabel "USBDATA" DriveLetter FriendlyName FileSystemType DriveType HealthStatus OperationalStatus SizeRemaining Size D USBDATA NTFS Fixed Healthy OK 14.95 GB 15 GB

#### Formatowanie dysku "USB"

PowerShell oferuje również polecenie cmdlet Clear-Disk, które czyści dysk, usuwając wszystkie informacje i dane o partycjach oraz usuwając je. Możesz użyć parametru -Number lub -FriendlyName, aby określić dysk do wyczyszczenia.

Ostrzeżenie: Zachowaj szczególną ostrożność podczas korzystania z polecenia "Wyczyść dysk" na dysku. Mały błąd w określeniu właściwego dysku może potencjalnie usunąć wszystkie ważne dane.

Następujące polecenie wyczyści mój dysk USB, zainicjuje go, utworzy partycję, sformatuje go w systemie plików NTFS i przypisze do niego etykietę woluminu:

Clear-Disk -Number 1 -RemoveData -RemoveOEM -Confirm:\$False -PassThru | Initialize-Disk -PartitionStyle MBR -PassThru | New-Partition -AssignDriveLetter -UseMaximumSize | Format-Volume -FileSystem NTFS -NewFileSystemLabel "USBDATA"

PS C:\Window e MBR -PassT TA"	vs\system32> ( Thru   New-Par	Clear-Disk -Numb rtition -AssignD	ber <b>1</b> -Remo DriveLetten	oveData -Remov r -UseMaximum9	/eOEM -Confirm:\$Fa] Size   <b>Format-Volu</b> m	lse -PassThru <b>ne</b> -FileSystem	Initialize-Disk -PartitionStyl NTFS -NewFileSystemLabel "USBD#
DriveLetter	FriendlyName	FileSystemType	DriveType	HealthStatus	OperationalStatus	SizeRemaining	Size
D	USBDATA	NTFS	Fixed	Healthy	ОК	14.95 GB	15 GB

Wyczyść, zainicjuj, podziel i sformatuj dysk

Wyczyść, zainicjuj, partycjonuj i sformatuj dysk

Parametr -Number służy do określenia dysku do wyczyszczenia.

Parametr -RemoveData służy do usuwania danych z dysku.

Parametr -RemoveOEM służy do usuwania z dysku partycji odzyskiwania OEM, jeśli takie istnieją.

Parametr -Confirm:\$false służy do wyłączania komunikatu potwierdzającego przed wyczyszczeniem dysku.

Parametr -PassThru wysyła obiekt wyjściowy z jednego polecenia w dół potoku. Ten parametr jest potrzebny, gdy polecenie domyślnie nie generuje żadnych danych wyjściowych, ale nadal chcesz przekazać obiekt wyjściowy potoku do następnego polecenia.

Podczas próby uruchomienia polecenia cmdlet Clear-Disk na dysku, który jest w stanie nie zainicjowanym, pojawi się komunikat o błędzie, Dysk nie został zainicjowany, jak pokazano na poniższym zrzucie ekranu:

Clear-Disk -Number 1 -RemoveData -RemoveOEM -Confirm:\$False

PS C:\Windows\system32> Clear-Disk -Number 1 -RemoveData -RemoveOEM -Confirm:\$False
PS C:\Windows\system32> \_

Wyświetl informacje o dysku po wyczyszczeniu

Get-Disk -Number 1

PS C:\\	vindows/system32> Get-Disk -Number 1	, , , ,			
Number	Friendly Name Serial Number	HealthStatus	OperationalStatus	Total Size	Partition Style
 1	Msft Virtu	 Healthy	Online	15 GB	RAW

Lub w środowisku graficznym

"ODysk 1 📃	
Nieznany	
15,00 GB 1	15,00 GB
Nie zainicjowa	Nieprzydzielone

Wyczyszczony dysk nie został zainicjowany

Aby obejść ten problem, musisz użyć polecenia Get-Disk. W takim przypadku możesz bezpośrednio zainicjować, podzielić i sformatować dysk za pomocą jednego polecenia, jak pokazano poniżej:

Get-Disk | where {\$\_.DiskNumber -eq '1'} | Initialize-Disk -PartitionStyle MBR -PassThru | New-Partition -AssignDriveLetter -UseMaximumSize | Format-Volume -FileSystem NTFS -NewFileSystemLabel "USBDATA"

Zainicjuj partycję i sformatuj dysk

PS C:\Window PS C:\Window on -AssignDu	vs\system32> ( vs\system32> ( riveLetter -Us	Clear-Disk -Numb Get-Disk   where SeMaximumSize	per 1 -Remo ≥ {\$DiskM Format-Vo]	oveData -Remov Number -eq '1 Lume -FileSyst	veOEM -Confirm:\$Fa '}   Initialize-Dis tem NTFS -NewFileSy	lse 5 <b>k</b> -PartitionSt /stemLabel "USE	yle MBR DATA"	-PassThru	New-Partiti
DriveLetter	FriendlyName	FileSystemType	DriveType	HealthStatus	OperationalStatus	SizeRemaining	Size		
D	USBDATA	NTFS	Fixed	Healthy	ок	14.95 GB	15 GB		

Gdy użyjesz Clear-Disk, a następnie spróbujesz użyć polecenia Set-Disk w celu zdefiniowania stylu partycji, otrzymasz błąd Nieobsługiwany.

Clear-Disk -Number 1 -RemoveData -RemoveOEM -Confirm:\$False -PassThru | Set-Disk -PartitionStyle MBR



Ustaw dysk: nieobsługiwany

Błąd Nieobsługiwane występuje, ponieważ polecenie Wyczyść dysk już wyczyściło i ustawiło dysk w stanie "Nie zainicjowany". Aby naprawić ten błąd, musisz użyć polecenia Initialize-Disk. Dysk można zainicjować, podzielić na partycje i sformatować za pomocą jednego polecenia w następujący sposób:

Get-Disk | where {\$\_.DiskNumber -eq '1'} | Initialize-Disk -PartitionStyle MBR -PassThru | New-Partition -AssignDriveLetter -UseMaximumSize | Format-Volume -FileSystem NTFS -NewFileSystemLabel "USBDATA"

PS C:\Window on -AssignDu	ws\system32> ( riveLetter -Us	Get-Disk   where seMaximumSize	e {\$DiskM Format-Vo]	<b>lumber</b> -eq <b>'1</b> <b>lume</b> -FileSyst	<pre>}   Initialize-Dis cem NTFS -NewFileSy</pre>	sk -Partitions /stemLabel "US	tyle MBR BDATA"	-PassThru	New-Partiti
DriveLetter	FriendlyName	FileSystemType	DriveType	HealthStatus	OperationalStatus	SizeRemaining	, Size		
D	USBDATA	NTFS	Fixed	Healthy	ОК	14.95 GE	15 GB		

Zainicjuj partycję i sformatuj dysk za pomocą jednego polecenia

# 6. Uzyskaj informacje o wykorzystaniu dysku za pomocą Get-CimInstance

Aby wyświetlić statystyki wykorzystania pamięci dla wszystkich dysków w systemie, możesz użyć następującego polecenia:

Get-CimInstance Win32\_LogicalDisk | Format-Table DeviceId, VolumeName, @{n="TotalSize (GB)";e={[math]::Round(\$\_.Size/1GB,2)}}, @{n="UsedSpace (GB)";e={[math]::Round(\$\_.Size/1GB -\$\_.FreeSpace/1GB,2)}}, @{n="FreeSpace (GB)";e={[math]::Round(\$\_.FreeSpace/1GB,2)}}

To polecenie pokazuje DriveLetter, VolumeName, Total Size, Used Space i Free Space dla każdej partycji w systemie w ładnym formacie tabelarycznym.

<pre>PS C:\Windows\system32&gt; Get-CimInstance Win32_LogicalDisk   Format-Table DeviceId, VolumeName, @{n="TotalSize (GB)";e={[math]: Round(\$Size/1GB,2)}}, @{n="UsedSpace (GB)";e={[math]::Round(\$Size/1GB - \$FreeSpace/1GB,2)}}, @{n="FreeSpace (GB)";e={[m ath]::Round(\$FreeSpace/1GB,2)}}</pre>									
DeviceId	VolumeName	TotalSize (GB)	UsedSpace (GB)	FreeSpace (GB)					
с:		1023,12	32,6	990,52					
D:	USBDATA	15	0,05	14,95					
E:	Nowy	469,98	0,11	469,87					
F:		0	0	0					
н:	DATA1	29,98	0,07	29,91					

Wyświetlanie całkowitego rozmiaru wykorzystanego miejsca i wolnego miejsca za pomocą funkcji Get Disk

Spójrz na kolumnę UsedSpace. Dysk H pokazuje zajętą pamięć 0,07 GB.

# 7. Zmień rozmiar operacji dyskowych za pomocą Resize-Partition

W poprzedniej sekcji utworzyliśmy pojedynczą partycję 20 GB na dysku 3 i przypisaliśmy do niej literę dysku G.

PS C:∖Wir	ndows\sys	tem32> Get-D	)isk -Number 3	Get-Partit	ion			
DiskPa	ath: \\?\	scsi#disk&ve	en_msft∏_v	irtual_disk#5	&14f89abb&0&0	000003#{53f5630	7-b6bf-11d0-94f2-00a0c91efb8b}	
Partitior	nNumber	DriveLetter	Offset			Size	Туре	
1		1	.7408			15.98 MB I	Reserved	
2		F 🗖	16777216			20 GB	Basic	
3		H	21491613696			29.98 GB	Basic	
PS C:\Wir :Round(\$_ ath]::Rou DeviceId	.Size/10 .Size/10 und(\$Fr	tem32≻ Get-C B,2)}}, @{n= eeSpace/1GB, me TotalSize	<pre>Uninstance wi "UsedSpace (Gi 2)}} e (GB) UsedSpa</pre>	n32_LogicalDi B)";e={[math] ce (GB) FreeS	sk   Format-1 ::Round(\$Si	able De <del>viceid,</del> .ze/1GB - \$Fro	VolumeName, @{n= TotalSize (GB) reeSpace/1GB,2)}}, @{n="FreeSpace	";e={[matn]: ? (GB)";e={[m
с:		16	23,12	32,6	990,52			
D:	USBDATA		15	0,05	14,95			
E:	Nowy	4	69,98	0,11	469,87			
F:			0	0	0			
Н: 🖚	DATA1-		29,98 💳	0,07—	29,91 💳			

Wyświetlanie całkowitego rozmiaru wykorzystanego miejsca i wolnego miejsca na dysku g za pomocą funkcji Get Disk

Na powyższym obrazku widać, że dysk H ma zajęte miejsce 0,07 GB. Jeśli więc chcesz zmienić rozmiar tej partycji, nie możesz zmienić jej rozmiaru poniżej 0,07 GB.

Aby zmienić rozmiar partycji na 25 GB, użyj polecenia Resize-Partition, jak pokazano poniżej:

```
Resize-Partition -DriveLetter H -Size 25GB
```

PS C:\Windows\system32> Resize-Partition -DriveLetter H -Size 25GB PS C:\Windows\system32> \_

Mamy błąd z powodu H: która zajmuje obszar za G

- Dysk 3 Podstawowy	(F:)	DATA1 (H:)	
49,98 GB	20,00 GB RAW	25,00 GB NTFS	4,98 GB
Online	Zdrowy (Podstawowa	Zdrowy (Podstawowa	Nieprzydzielone

Zmiana rozmiaru partycji

Partycja została pomyślnie przeskalowana do 25 GB.

Należy usunąć partycje F:

Remove-Partition -DriveLetter F

```
PS C:\Windows\system32> Remove-Partition -DriveLetter F
Confirm
Are you sure you want to perform this action?
This will erase all data on disk 3 partition 2.
[Y] Yes [A] Yes to All [N] No [L] No to All [S] Suspend [?] Help (default is "Y"): Y
PS C:\Windows\system32> _
```

Zmiana rozmiaru

Jeśli spróbujesz ustawić wartość parametru -Size na niższą niż zajęte miejsce, pojawi się błąd Rozmiar nieobsługiwany.

Resize-Partition -DriveLetter H -Size 0,05GB



Rozmiar partycji nie jest obsługiwany

Aby naprawić ten błąd, musisz ustawić wartość parametru -Size na wyższą wartość niż użyte miejsce.

Podobnie, jeśli spróbujesz ustawić wartość parametru -Size na wartość wyższą niż Total Size, otrzymasz błąd Niewystarczająca dostępna pojemność.

#### Resize-Partition -DriveLetter H -Size 70GB



Zmień rozmiar partycji Za mało dostępnej pojemności

Aby obejść ten błąd, musisz ustawić wartość parametru -Size na wartość niższą niż "Rozmiar całkowity".

Załóżmy teraz, że chcesz zmienić rozmiar partycji do maksymalnej pojemności.

Niestety żaden parametr nie jest dostępny z poleceniem cmdlet Resize-Partition do obsługi tej operacji. Ale zawsze jest sposób. Moduł magazynu oferuje polecenie cmdlet Get-PartitionSupportedSize, które zwraca dwie wartości: SizeMin i SizeMax.

#### Get-PartitionSupportedSize -DriveLetter H



Wyświetlanie SizeMin i SizeMax za pomocą Get PartitionSupportedSize

Możemy użyć tego polecenia, wraz z Resize-Partition, aby uzyskać pożądany rezultat.

Przyjrzyjmy się następującym poleceniom:

\$supportedSize = (Get-PartitionSupportedSize -DriveLetter H)

Resize-Partition -DriveLetter H -Size \$supportedSize.SizeMax



Po uruchomieniu powyższych poleceń rozmiar partycji powinien zostać zmieniony do maksymalnej obsługiwanej pojemności. Pojawił się błąd.

Aby ponownie wyświetlić statystyki użycia dysku, użyj następującego polecenia:

Get-CimInstance Win32_LogicalDisk   Format-Table DeviceId, VolumeName, @{n="TotalSize
(GB)";e={[math]::Round(\$Size/1GB,2)}}, @{n="UsedSpace (GB)";e={[math]::Round(\$Size/1GB
<pre>\$FreeSpace/1GB,2)}}, @{n="FreeSpace (GB)";e={[math]::Round(\$FreeSpace/1GB,2)}}</pre>

PS C:\Wi PS C:\Wi PS C:\Wi :Round(\$ ath]::Ro	<pre>PS C:\Windows\system32&gt; \$supportedSize = (Get-PartitionSupportedSize -DriveLetter H) PS C:\Windows\system32&gt; Resize-Partition -DriveLetter H -Size \$supportedSize.SizeMax PS C:\Windows\system32&gt; Get-CimInstance Win32_LogicalDisk   Format-Table DeviceId, VolumeName, @{n="TotalSize (GB)";e={[math]: Round(\$Size/1GB,2)}}, @{n="UsedSpace (GB)";e={[math]::Round(\$Size/1GB - \$FreeSpace/1GB,2)}}, @{n="FreeSpace (GB)";e={[math]::Round(\$Size/1GB - \$FreeSpace/1GB,2)}}, @{n="FreeSpace (GB)";e={[math]::Round(\$Size/1GB - \$FreeSpace/1GB,2)}}, @{n="FreeSpace (GB)";e={[math]::Round(\$Size/1GB - \$FreeSpace/1GB,2)}}</pre>									
C:		1023,12	32,83	990,29						
D:	USBDATA	15	0,05	14,95						
E:	Nowy	469,98	0,11	469,87						
H:•	DATA1 *	29,98 '	0,07 •	29,91°						

Zmiana rozmiaru partycji do maksymalnego obsługiwanego rozmiaru za pomocą Resize Partition

Aby zmienić rozmiar partycji do maksymalnego obsługiwanego rozmiaru za pomocą polecenia Resize-Partition, możesz użyć polecenia Get-PartitionSupportedSize, które zwraca minimalny i maksymalny rozmiar partycji. Następnie użyj tych wartości w poleceniu Resize-Partition.:

#### 1. Uzyskaj obsługiwane rozmiary partycji:

\$supportedSize = Get-PartitionSupportedSize -DriveLetter H

#### 2. Zmień rozmiar partycji do maksymalnego obsługiwanego rozmiaru:

Resize-Partition -DriveLetter H -Size \$supportedSize.SizeMax

Powyższe polecenia najpierw uzyskują minimalny i maksymalny rozmiar partycji dla partycji oznaczonej literą H, a następnie zmieniają jej rozmiar do maksymalnej obsługiwanej pojemności.



Aby przenieść partycję H na początek Dysku 3, musisz wykonać kilka kroków. Niestety, PowerShell nie oferuje bezpośredniego polecenia do przenoszenia partycji. Możesz osiągnąć ten cel, tworząc kopię

partycji, usuwając oryginalną partycję, a następnie tworząc nową partycję na początku dysku i przywracając dane.:

1. Utwórz kopię zapasową danych z partycji H: Upewnij się, że masz kopię zapasową wszystkich danych na partycji H, ponieważ będziesz musiał usunąć tę partycję.

# 2. Usuń partycję H:

Remove-Partition -DriveLetter H



3. Utwórz nową partycję na początku dysku:

New-Partition -DiskNumber 3 -UseMaximumSize -DriveLetter H

PS C:\Windows\sy	stem32> New-F	Partition -DiskNumber 3	3 -UseMaximumSize -DriveLetter H	
DiskPath: \\?	\scsi#disk&ve	en_msft∏_virtual_di	isk#5&14f89abb&0&00003#{53f5630	7-b6bf-11d0-94f2-00a0c91efb8b}
PartitionNumber	DriveLetter	Offset	Size	Туре
2	Н	16777216	49.98 GB	Basic

### 4. Sformatuj nową partycję:

Format-Volume -DriveLetter H -FileSystem NTFS -NewFileSystemLabel "DATA1"

PS C:\Windo	ws\system32>	Format-Volume -[	DriveLetter	∩ H -FileSyste	em NTFS -NewFileSys	stemLabel <mark>"DATA</mark>	1"
DriveLetter	FriendlyName	FileSystemType	DriveType	HealthStatus	OperationalStatus	SizeRemaining	Size
н	DATA1	NTFS	Fixed	Healthy	ОК	49.89 GB	49.98 GB

5. Przywróć dane na nową partycję: Skopiuj dane z kopii zapasowej na nową partycję H.

Pamiętaj, że te kroki spowodują usunięcie wszystkich danych z partycji H, więc upewnij się, że masz kopię zapasową przed rozpoczęciem.