



BIENVENIDOS!

Manuel Domínguez.



mftienda@Gmail.com



@mafradoti



www.linkedin.com/in/mfienda



<https://github.com/mftienda>





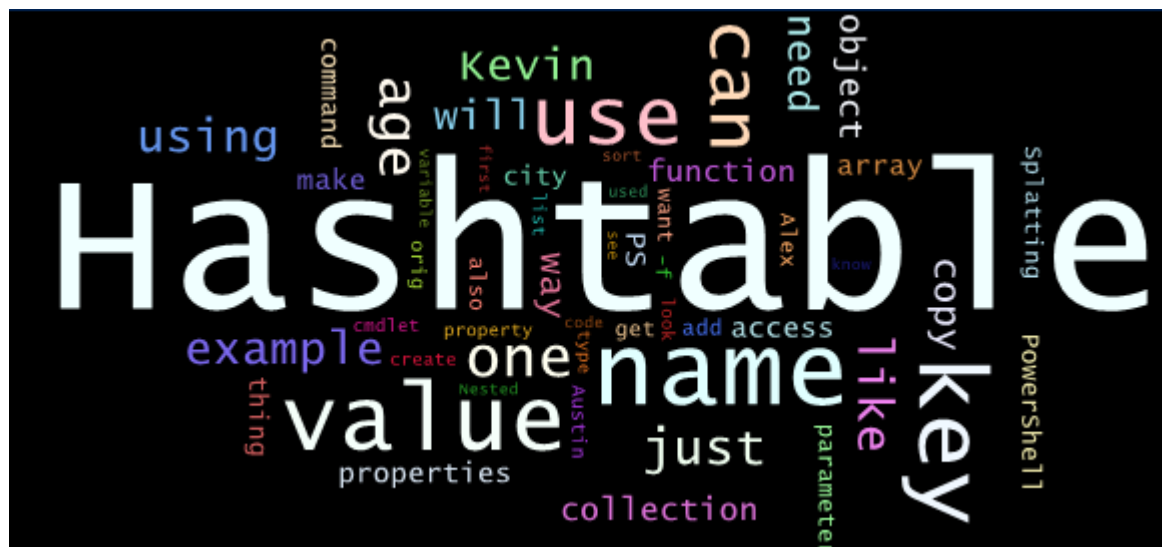
Ud.- Propiedades de los objetos

Índice

- 1.- Introducción
- 2.- Ver las propiedades y métodos de un objeto.
- 3.- Obtener una propiedad.
- 4.- Aplicar un método.



1.- Introducción





2.- Ver las propiedades y los métodos de un objeto

Get-Member → Nos da información sobre las propiedades de un objeto, así como los métodos o funciones que podemos aplicarle.

Ejemplo:

```
PS D:\> Get-LocalUser | Get-Member
```



2.- Ver las propiedades y métodos de un objeto

Get-Member

Ejercicio:

1.- Definimos una variable y vemos su contenido en pantalla:

```
PS C:\Users\Windows> $saludos ='Buenos días'  
PS C:\Users\Windows> $saludos  
Buenos días  
PS C:\Users\Windows>
```

2.- Veamos las propiedades y los métodos asociados: **\$saludos |Get-Member**

ToUpper	Method	string ToUpper(), string ToU...
ToUpperInvariant	Method	string ToUpperInvariant()
Trim	Method	string Trim(Params char[] tr...
TrimEnd	Method	string TrimEnd(Params char[]...
TrimStart	Method	string TrimStart(Params char...
Chars	ParameterizedProperty	char Chars(int index) {get;}
Length	Property	int Length {get;}



3.- Obtener una propiedad.

Get-Member

Ejercicio:

3.- Veamos una propiedad: **Length**

```
PS D:\> $saludos.Length  
11
```



4.- Aplicar un método.

Get-Member

Ejercicio:

4.-Apliquemos un método: **ToUpper ()**

```
PS D:\> $saludos.ToUpper()  
BUENOS DÍAS
```

Observemos como el **método**, termina en () y la propiedad no.



5.- Añadir una propiedad calculada.

Propiedad calculada

Para poner una **columna calculada**, se debe definir como un **hash table** (es una estructura de datos que asigna claves a valores).

1er formato:

```
$hash = @ {Clave1=Valor1; Clave2=Valor2; Clave3=Valor3}
```

2º formato:

```
$hash = @ {  
    Clave1 = 'Valor1'  
    Clave2 = 'Valor2'  
    Clave3 = 'Valor3'  
}
```




5.- Añadir una propiedad calculada.

Propiedad calculada

Los posibles valores para **format-table** son:

Name o Label: Nombre de la columna.

Expression: expresión de cálculo de la columna.

FormatString: formato de la columna.

Ejemplo: "N2" (número con separadores de miles y dos decimales)

Width: ancho de la columna.

Alignment: alineamiento de la columna (Left, Center o Right)



5.- Añadir una propiedad calculada.

Propiedad calculada: format-table

Ejemplo 1:

- Abre el Notepad.
- Crear una etiqueta llamada **Consumo** CPU, que muestre la propiedad CPU, como un valor numérico con 3 decimales.

```
PS D:\> get-process notepad|ft Name,cpu

Name      CPU
----      -
notepad  0,09375

PS D:\> get-process notepad|ft Name, @{Name="Consumo CPU"; Expression =
{$_.CPU}; FormatString="N3"}

Name      Consumo CPU
----      -
notepad    0,094
```



5.- Añadir una propiedad calculada.

Propiedad calculada: format-table

Ejemplo : Crear una etiqueta llamada **Tamaño (GB)**, que muestre la propiedad size de Get-Disk en GB, como un valor numérico con 2 decimales.

```
PS D:\> get-disk|ft Number,Size
```

Number	Size
1	1000204886016
0	240057409536

```
PS D:\> get-disk|ft Number,Size,@{Name="Tamaño (GB)";Expression={$_.size/1GB}};FormatString="N2"
```

Number	Size	Tamaño (GB)
1	1000204886016	931,51
0	240057409536	223,57



5.- Añadir una propiedad calculada.

Propiedad calculada: format-table

Ejemplo : Crear una etiqueta llamada **Tiempo de Ejecución**, que muestre el tiempo que lleva corriendo el proceso Notepad.

Get-Date → Fecha y hora actuales.

Start.Time → Cuando se inicio el proceso.

```
PS D:\> Get-Process Notepad | Format-Table ProcessName,@{Label="Tiempo de ejecución"; Expression=((Get-Date) - $_.StartTime)}
```



Dudas





Sugerencias/mejoras del tema



Sugerencias /mejoras del tema