

Power Shell Cap. 4

Variables II: Tipos de dato



La memoria manda

- » Recurso limitado
- » Protección
- » Ajuste fijo o ajuste dinámico
- » Longitud de la reserva

Tipos de datos

- » **[string]**: Cadenas de caracteres
- » **[char]**: un sólo carácter **Unicode**
- » **[int]**: Entero con signo de 32 bits: -2147483648 a 2147483647
- » **[long]**: Entero con signo de 64 bits: ...
- » **[single]**: Número en coma flotante de 32 bits
- » **[double]**: Número en coma flotante de 64 bits

Tipos de datos II

- » **[datetime]**: Fecha y hora: 1 enero 100 hasta 31 diciembre 9999
- » **[bool]**: Valor lógico: True o False
- » **[array]**: Conjunto de valores
- » **[hashtable]**: Objeto que representa una tabla hash

Obtener el tipo de una variable

- » Una vez definida una variable y dado un valor podemos saber qué tipo de variable es

```
$nombre = "ADD Costa Tropical"  
Write-Host $nombre.GetType().name
```

- » Orientación a objetos
- » Lenguaje de alto nivel

Definición de variables con tipo

- » Implícita: el tipo de dato se asigna automáticamente según el valor que le hayamos dado
- » Explícita: definimos el tipo de dato a la hora de definir la variable, cuando le demos un valor, se amoldará al tipo establecido

Definición implícita

```
$nombre = "ADD Costa Tropical"
```

» La variable \$nombre se le asigna el tipo String

```
$precio = 4.99
```

» La variable \$precio se le asigna el tipo double

Definición implícita II

```
$precio = 4.99
```

```
$precio = "justo"
```

- » La variable \$nombre se le asigna el tipo double
- » Después se le cambia el tipo a String en tiempo de ejecución
- » Ejemplo en PowerShell

Definición explícita

```
$temperatura = 4.99
```

» En modo implícito será double (64 bits)

```
[float] $temperatura = 4.99
```

» En modo explícito será float (32 bits)

» Dependerá de la naturaleza de los datos guardados

Definición explícita II

- » Cuando una variable tiene un tipo definido, y le asignamos valor, se amoldará al tipo de
- » Si no es posible amoldarse, dará como resultado un error en la asignación del valor
- » En la adaptación del valor podemos perder información (redondeo)

```
[float] $temperatura = 4.99  
$temperatura = "Hola que tal"
```

Casting, especificar tipo de dato

```
$fecha = "3/11/2009"
```

```
Write-Host $fecha
```

» Si no le decimos lo contrario, se almacenará como string

```
$fecha = [datetime]"3/11/2009"
```

```
Write-Host $fecha
```

» En este caso la variable \$fecha será de tipo date

Constantes

- » Las constantes son variables que no cambian su valor en tiempo de ejecución
- » Si lo intentamos, nos saltará un error

```
New-Variable -Name $nombre -Value "ADD Costa Tropical" -Option ReadOnly
```

- » Ejemplo de uso: iva, pi...

Espacio ocupado en memoria

- » Depende del tipo de variable
- » **Longitud fija:** El tamaño de la variable en la memoria no cambiará aunque cambie el dato que almacena. En este grupo tenemos la mayoría de los tipos de datos básicos: char, byte, int, long, single, double, decimal, datetime y bool.
- » **Longitud variable:** En este caso, el tamaño de la variable se ajusta a dato que le estemos asignando. En este grupo se encuentran: string, array, hashtable y xml.

Preguntas y respuestas

No tengas miedo, es gratis 😊

ADD Costa Tropical

- » <https://addcostatropical.org>
- » Youtube => ADD Costa Tropical
- » @addcostatropical
- » #somosadd
- » Facebook
- » Instagram

ADD Costa Tropical

» Investiga, aprende y comparte

Gracias

» Un placer compartir con vosotros