

Tablas

1. Tablas de una dimensión

La tabla de una dimensión es el caso más sencillo, los valores se colocan uno tras otro, y es suficiente indicar el número de índice para utilizar el contenido. Una tabla de una dimensión se denomina a veces lista.

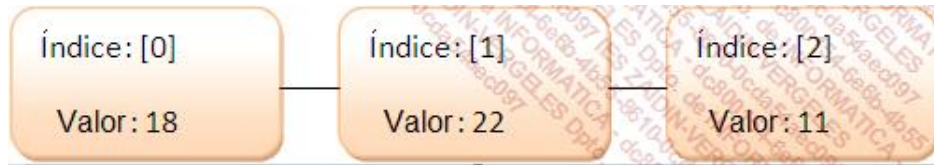


Ilustración de una tabla de una dimensión

Por ejemplo:

El valor 18 está contenido en la tabla en el índice 0.

El valor 22 está contenido en la tabla en el índice 1.



Con PowerShell los índices de tabla empiezan por 0 y no por 1 como con otros lenguajes.

a. Inicializar una tabla de una dimensión

Para crear una tabla e inicializarla, basta asignar varios valores separados por una coma. Por ejemplo: `$tab = 1,5,9,10,6` es una tabla de tipo entero que contiene 1 en el índice 0; 5 en el índice 1; 9 en el índice 2; etc.



Una tabla puede también inicializarse con el operador de intervalo, por ejemplo: `$tab = 1..20` es una tabla de enteros que contiene todos los valores que van de 1 a 20.

Conviene indicar que el tipo de objeto introducido en la tabla es asignado de forma automática, pero como con las variables simples, si lo desea podrá forzar el tipo de los datos contenidos en la tabla.

Ejemplo:

```
PS > [int[]]$tab = 1,2,3
```

Observará los corchetes `[]` inmediatamente después del nombre del tipo. Los corchetes simbolizan el hecho de que se trata de una tabla de valores del tipo en cuestión. En este ejemplo, la tabla `$tab` puede contener únicamente valores enteros.

Pero una tabla también puede ser heterogénea, y en este caso, la asignación de los tipos se hace valor por valor.

Ejemplo:

```
PS > $tab = [int]1,[double]2.5,[char]'A'
```

b. Leer las tablas de una dimensión

Para leer una tabla de una dimensión se pueden utilizar varios métodos.

El más sencillo es introducir su nombre en la consola, en este caso, todos los elementos de la tabla se visualizarán. Pero para leer un valor de un índice concreto, basta con indicar entre corchetes el índice deseado.

```
PS > $tab[0]
1
```

Para leer varios valores de índices concretos, basta, esta vez, con indicar entre corchetes los índices separados por comas.

```
PS > $tab[0,2]
1
3
```

Aquí, sólo se obtendrán los valores de los índices 0 y 2, pero el valor del índice 1 no se obtiene.



También puede mostrar varios valores con el operador de intervalo, por ejemplo: `$tab[1..20]` aquí visualizará los valores de los índices del 1 al 20.

Ahora, supongamos que queremos únicamente leer el valor contenido en el último índice. Uno de los métodos consiste en saber cuántos valores están contenidos en nuestra tabla. Esto lo podemos saber mediante la propiedad `Length`:

```
PS > $tab[$tab.Length-1]
3
```

Tenga en cuenta que estamos eliminando una unidad de la propiedad `Length` porque los índices comienzan por 0 y no por 1.

Pero existe otro método más sencillo: los índices negativos.

Cuando utilice un índice negativo, estará haciendo referencia al número del índice desde el final de la tabla.

Ejemplo:

```
PS > $tab[-1]
3
PS > $tab[-3..-1]
1
2
3
```

El método más común para leer una tabla sigue siendo recorrer las tablas con bucles (*While*, *For*, *Foreach*). Para más información, consulte la sección Los bucles (*While*, *For* et *Foreach*) de este capítulo acerca de los bucles y condiciones.

c. Operaciones sobre las tablas de una dimensión

Concatenar dos tablas

Con PowerShell la concatenación de tablas se hace con el operador «+». Supongamos que por un motivo cualquiera tenemos la necesidad de concatenar dos tablas (o más). Para ello, bastará con sumar las tablas con el operador «+».

Ejemplo de la adición de dos tablas de caracteres llamadas `$cadena1` y `$cadena2`:

```
PS > $cadena1 = 'P','O','W','E','R'
PS > $cadena2 = 'S','H','E','L','L'
PS > $cadena1 + $cadena2
P
```

```
o  
w  
e  
r  
s  
h  
e  
l  
l
```

Añadir un elemento a una tabla

En PowerShell, para añadir un valor a una tabla se utiliza el operador «+=». De este modo tecleando la línea siguiente, añadimos el valor 4 a nuestra tabla:

```
PS > $tab= 1,2,3  
PS > $tab += 4  
PS > $tab  
1  
2  
3  
4
```

Modificar el valor de un elemento

La modificación de un elemento en una tabla se realiza con el operador «=».

Ejemplo, escribiendo `$tab[2]=1` modificamos el valor contenido en el índice 2 (el tercer valor, por tanto). En realidad, se realiza una nueva asignación, y se borra el valor antiguo por el nuevo.

Ejemplo:

```
PS > $tab = 'A', 'B'  
PS > $tab[0] = 'C'  
PS > $tab  
C  
B
```

Existe una segunda técnica para modificar un valor existente. Para ello, debemos recurrir a un método específico para los objetos de tipo tabla: `SetValue`.

Utilizando `SetValue` e indicándole por parámetro el nuevo valor seguido del índice de la tabla realizamos la asignación.

```
PS > $tab = 'A', 'B'  
PS > $tab.SetValue('C',0)  
PS > $tab  
C  
B
```

d. Suprimir un elemento

Con PowerShell, no es posible suprimir un elemento de una tabla. Pensándolo bien, hay una explicación lógica: con cada supresión de un elemento, se necesita un reajuste de los índices para cada valor, y pronto nos perderíamos. Sin embargo, existe una solución alternativa, que permitirá bordear el problema. Esta consiste en efectuar un duplicado de una tabla, excluyendo uno o varios índices.

Ejemplo: Supresión de elementos en una tabla

Tomemos el ejemplo de las últimas notas del examen de fin de curso.

```
PS > $notas = 6, 9, 5, 7, 4, 6
```

Al no haber aprobado Álgebra (4) decide eliminar esta nota ya que no le satisface en absoluto.

Para ello, procederemos simplemente a copiar de nuevo los elementos de la tabla a excepción del valor del índice 4:

```
PS > $notas = $notas[0..3 + 5]
PS > $notas
6
9
5
7
6
```

Si se desconocen los índices, o si el número de notas a suprimir es demasiado grande, también se puede utilizar lo que se denomina un filtro. Aunque todavía no hayamos visto los filtros, he aquí cómo, gracias a un filtro y a un operador de comparación, podemos obtener un duplicado de la tabla con únicamente los valores superiores a 5.

```
PS > $notas2 = $notas | where-object {$_ -ge 5}
PS > $notas2
6
9
5
7
6
```

2. Tablas de varias dimensiones

Cuando hablamos de tablas de varias dimensiones, hablamos de tablas de varios índices, con tantos índices como dimensiones. Así, para pasar de una tabla de una dimensión a una tabla de dos dimensiones, basta añadir un índice que permita detectar esta nueva dimensión.

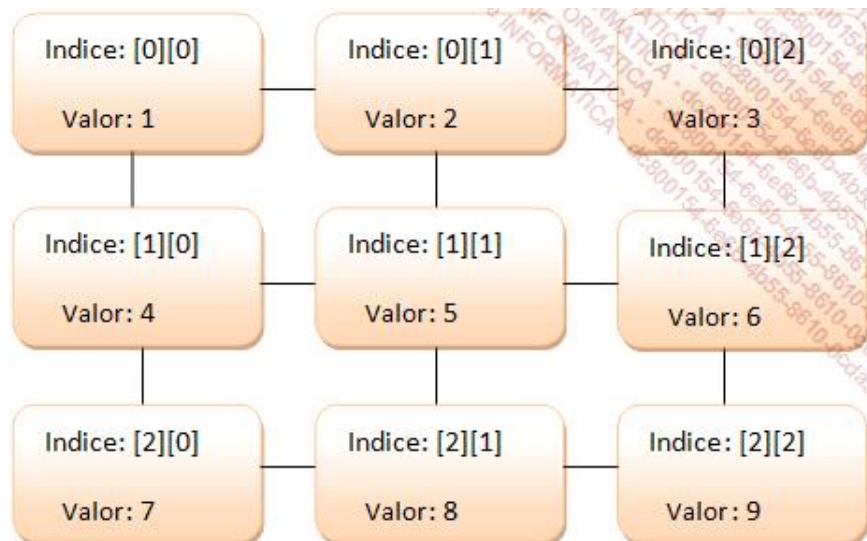


Ilustración de una tabla de dos dimensiones

La lectura de las tablas de varias dimensiones es similar a las de una dimensión. La única dificultad es tener que jugar con los índices. Tomaremos como ejemplo la tabla anterior.

La lectura de la tabla con el índice «0», nos dará la primera línea de esta tabla:

```
PS > $tab[0]
1
2
3
```

Para obtener un valor concreto, debemos simplemente fijar el índice de la dimensión horizontal y el correspondiente a la vertical.

```
PS > $tab[0][2]
3
```