

UNIVERSIDAD DE LAS FUERZAS ARMADAS
"ESPE"

Nombre: Aaron Moncayo

FECHA: 06-12-2023

NRC: 13930

¿Cuál es la diferencia entre un for y un foreach?

La instrucción for ejecuta una instrucción o un bloque de instrucciones mientras una expresión booleana especificada se evalúa como true. En el ejemplo siguiente se muestra la instrucción for, que ejecuta su cuerpo mientras que un contador entero sea menor que tres:

```
for (int i = 0; i < 3; i++)  
{  
    Console.Write(i);  
}
```

// Output:

// 012

En el ejemplo anterior se muestran los elementos de la instrucción for:

int i = 0

La sección condición que determina si se debe ejecutar la siguiente iteración del bucle. Si se evalúa como true o no está presente, se ejecuta la siguiente iteración; de lo contrario, se sale del bucle. La sección condición debe ser una expresión booleana.

La sección condición del ejemplo anterior comprueba si un valor de contador es menor que tres:

i < 3

La sección iterador, que define lo que sucede después de cada iteración del cuerpo del bucle.

La sección iterador del ejemplo anterior incrementa el contador:

i++

El cuerpo del bucle, que es una instrucción o un bloque de instrucciones.

La sección iterador puede contener cero o más de las siguientes expresiones de instrucción, separadas por comas:

expresión de incremento de prefijo o sufijo, como ++i o i++

expresión de decremento de prefijo o sufijo, como --i o i--

asignación

invocación de un método

Expresión await

Creación de un objeto mediante el operador new.

EJEMPLO:

```
int i;
int j = 3;
for (i = 0, Console.WriteLine($"Start: i={i}, j={j}"); i < j; i++, j--,
    Console.WriteLine($"Step: i={i}, j={j}"))
{
    //...
}
// Output:
// Start: i=0, j=3
// Step: i=1, j=2
// Step: i=2, j=1
```

Foreach

La instrucción foreach ejecuta una instrucción o un bloque de instrucciones para cada elemento de una instancia del tipo que implementa la interfaz System.Collections.IEnumerable o System.Collections.Generic.IEnumerable<T>, como se muestra en el siguiente ejemplo:

```
List<int> fibNumbers = [0, 1, 1, 2, 3, 5, 8, 13];
foreach (int element in fibNumbers)
```

```
{  
    Console.Write($"{element} ");  
}
```

// Output:

// 0 1 1 2 3 5 8 13

-La instrucción foreach no está limitada a esos tipos. Puede usarla con una instancia de cualquier tipo que cumpla las condiciones siguientes:

- . Un tipo tiene el método público GetEnumerator sin parámetros. El método GetEnumerator puede ser el método de extensión de un tipo.

- . El tipo de valor devuelto del método GetEnumerator tiene la propiedad pública Current y el método público MoveNext sin parámetros, cuyo tipo de valor devuelto es bool.

Ejemplo:

se usa la instrucción foreach con una instancia del tipo System.Span<T>, que no implementa ninguna interfaz:

```
Span<int> numbers = [3, 14, 15, 92, 6];
```

```
foreach (int number in numbers)
```

```
{  
    Console.Write($"{number} ");  
}
```

// Output:

// 3 14 15 92 6

-Si la propiedad Current del enumerador devuelve un valor devuelto de referencia (ref T donde T es el tipo de un elemento de colección), puede declarar una variable de iteración con el modificador ref o ref readonly, como se muestra en el ejemplo siguiente:

EJEMPLO:

```
Span<int> storage = stackalloc int[10];
int num = 0;
foreach (ref int item in storage)
{
    item = num++;
}
foreach (ref readonly var item in storage)
{
    Console.Write($"{item} ");
}
// Output:
// 0 1 2 3 4 5 6 7 8 9
```