



TEMA:

Estructura de decisión
Switch, Ciclos y
Matrices.




**¿QUE IMPORTANCIA
TIENE ESTE TEMA?**




OBJETIVO DE APRENDIZAJE:

- El o la participante estará en capacidad de definir y utilizar la estructura de decisión Switch, los bucles y matrices de 1 y 2 dimensiones en aplicaciones de visual C#.



**¿QUE CONOCIMIENTOS O
EXPERIENCIA TIENE
SOBRE ESTE TEMA?**



ESTRUCTURA DE DECISION SWITCH

Estructura Switch

- La estructura switch permite un funcionamiento equivalente a IF, pero ofrece una mejor legibilidad del código.
- Sintaxis

```
switch (variable)
{
    case valor1:
        Bloque de código 1;
        Break;
    case valor2:
        Bloque de código 2 ;
        Break;
    case valor3:
        Bloque de código 3 ;
        Break;
    default:
        Bloque de código 4;
        Break;
}
```

Ejemplo switch (continúa...):

```
double a;  
double b;  
double resultado;  
//Creamos un menu de opciones  
Console.WriteLine("MENU DE OPCIONES:");  
Console.WriteLine();  
Console.WriteLine("1. Area de un triangulo");  
Console.WriteLine("2. Area de un cuadrado");  
//Capturamos la opcion elegida  
int opcion = Convert.ToInt32(Console.ReadLine());
```

Ejemplo switch

```
switch (opcion)
{
    case 1:
        //Solicitamos los valores
        Console.WriteLine("Introduzca la base: ");
        double.TryParse(Console.ReadLine(), out a);
        Console.WriteLine("Introduzca la altura: ");
        double.TryParse(Console.ReadLine(), out b);
        resultado = a * b / 2;
        break;
    case 2:
        //Solicitamos los valores
        Console.WriteLine("Introduzca el lado 1: ");
        double.TryParse(Console.ReadLine(), out a);
        Console.WriteLine("Introduzca el lado 2: ");
        double.TryParse(Console.ReadLine(), out b);
        resultado = a * b;
        break;
    default:
        Console.WriteLine("Opcion incorrecta...");
        Console.ReadKey();
        return;
}
Console.WriteLine("El area es: {0}",resultado);
Console.ReadKey();
```




ESTRUCTURAS DE BUCLE O CICLOS

Contadores y Acumuladores

- Contador:
 - $\text{Variable} = \text{Variable} + 1$
- Acumulador
 - $\text{Variable} = \text{Variable} + \text{Valor}$



Estructuras de bucle

- While
- do...while
- for
- foreach

While

- Sintaxis:

```
while (condición)
{
    Bloque de código;
}
```

Ejemplo While:

```
//Muestra los numeros del 1 al 100  
int num=0;  
while (num < 100)  
{  
    num = num + 1;//Contador  
    Console.WriteLine(num);  
}
```

Ejemplo While:

```
//Muestra los numeros del 1 al 100 y los suma
int num = 0;
int totAcumulado = 0;
while (num < 100)
{
    num = num + 1;//Contador
    totAcumulado = totAcumulado + num;//Acumulador
    Console.WriteLine("Numero: {0} \t Acumulado:
                        {1}",num, totAcumulado);
}
```

Do while

- Sintaxis:

```
Do
```

```
{
```

```
    Bloque de código;
```

```
}
```

```
while (condicion);
```

Ejemplo Do While:

```
//Números impares del 1 al 100  
int num = 1;  
do  
{  
    Console.WriteLine(num);  
    num = num + 2;  
}  
while (num < 100);
```


For

- Sintaxis:

```
for(inicialización del contador ;condición ;instrucción de iteración)
{
    Bloque de código;
}
```

Ejemplo for

```
//Muestra los numeros del 1 al 100
for (int num = 1; num <= 100; num++)
{
    Console.WriteLine(num);
}
```

Ejemplo for

/*muestra los números pares, salta múltiplos de 10 y termina rompiendo el bucle cuando se sobrepasa el valor 90:*/

```
for (int num = 0; num <= 100; num = num + 2)
{
    if (num % 10==0) continue;
    if (num >= 90) break;
    Console.WriteLine(num);
}
Console.ReadKey();
```

foreach

- Sintaxis:

```
foreach (elemento en matriz)
{
    Bloque de código;
}
```

Ejemplo foreach

- **Con un bucle clásico:**

```
string[] productos= { "Platano", "Arroz", "Carne", "Habichuela" };  
int fila;  
for (fila= 0; contador < productos.Length; fila++)  
{  
    Console.WriteLine(productos[fila]);  
}
```

- **Con el bucle foreach:**

```
string[] productos= { "Platano", "Arroz", "Carne", "Habichuela" };  
foreach (string palabra in productos)  
{  
    Console.WriteLine(palabra);  
}
```



MATRICES

Matrices

- Sintaxis:

- Declaración:

`string[] productos;`

`string[,] ventas;`

- Inicialización:

`productos=new string[4];`

`ventas = new string[4, 2];`

1 dimensión

2 dimensiones

Matriz de una dimensión

- Matriz Productos

Posicion	Datos
0	Platano
1	Arroz
2	Carne
3	Habichuela
...	...

```
Console.WriteLine(productos[2]); //Carne
Console.WriteLine(productos[3]); //Habichuela
Console.WriteLine(productos[0]); //Platano
```


Matriz de dos dimensiones

- Matriz Ventas

Posicion	0	1	...
0	Platano	3	...
1	Arroz	2	...
2	Carne	1	...
3	Habichuela	0.5	...
...

```
Console.WriteLine(ventas[0,0]); //Platano
Console.WriteLine(ventas[3,1]); //0.5
Console.WriteLine(ventas[2,1]); //1
```

Ejemplo matriz 2 dimensiones

```
string[,] ventas = new string[3, 2];
for (int fila = 0; fila < 3; fila++)
{
    Console.WriteLine("Producto: ");
    ventas[fila, 0] = Console.ReadLine();
    Console.WriteLine("Cantidad: ");
    ventas[fila, 1] = Console.ReadLine();
}

Console.WriteLine("LISTA DE PRODUCTOS");
for (int fila = 0; fila < 3; fila++)
{
    Console.WriteLine("Producto: {0} \t Cantidad: {1} ", ventas[fila, 0],
        ventas[fila, 1]);
}
Console.ReadKey();
```