

# PROGRAMACIÓN

## Guía de estudio 6: Que empiece la función

### Tema: Estructuras cíclicas / iterativas. While, Do-While y for.

<b>Nivel:</b> Secundario - Modalidad Educación Técnico-Profesional.
<b>Ciclo:</b> Segundo ciclo.
<b>Especialidades:</b> Informática, Programación.

#### Introducción

¿Se pusieron a pensar en algún momento qué tan largo quedaría lo que está dentro del main en un juego, por ejemplo? Bueno, por suerte hay forma de dividirlo. Para eso (entre otras cosas) existen las funciones. Sus principales funciones (¡valga la redundancia!) son ayudar a la organización y entendimiento del código y hacer “bloques” de código reutilizables. ¡Veamos cómo se hace eso!

**¿Qué estamos aprendiendo?** Funciones void (vacías) y con retorno en C#.

#### Recursos

Video nro 15 del curso:

[https://youtu.be/3oX\\_xmiMjtM](https://youtu.be/3oX_xmiMjtM)



Video nro 16 del curso:

<https://youtu.be/CpGFZgkUCfc>



## Secuencia didáctica

Como hemos introducido, veremos cómo desarrollar funciones. Esta guía es la primera de los 2 guías de funciones. Las funciones son realmente útiles y nos ayudan a no marearse tanto cuando nos encontramos con tanto código. Vean los videos y despues los esperan las usuales preguntas.

<a href="https://youtu.be/3oX_xmiMjtM">https://youtu.be/3oX_xmiMjtM</a>	
<a href="https://youtu.be/CpGFZgkUCfc">https://youtu.be/CpGFZgkUCfc</a>	

## Actividades

Después de mirar el video del link o del QR, te invitamos a que reflexiones con las siguientes preguntas.

*No olvides escribir tus respuestas, te facilitarán realizar trabajos prácticos y repasar los temas que hayas estudiado con esta guía.*

- 1) ¿Qué identificador debemos usar en nuestra función si no queremos que devuelva ningún valor? ¿Para qué me servirían estas funciones?
- 2) Hacer una función que se llame "hacerCuadrado". No deberá devolver ningún valor y dentro de la función se deberá agregar el código necesario para que se imprima un cuadrado de "\*". El programa deberá imprimir por pantalla tantos cuadrados como el usuario le indique.
- 3) Hacer dos funciones. Una llamada "activarMotores" y otra llamada "frenar".

- a) activarMotores: contendrá una variable llamada "metrosParaAvanzar" inicializada con un valor elegido por vos. Deberá imprimir el texto "motores a max velocidad" y deberá devolver el valor de la variable declarada (un número entero).
- b) frenar: deberá imprimir "Robot frenado".
- c) Main (programa principal): deberá cambiar el estado del robot (avanzando o frenado) cada vez que se toque la barra espaciadora. El programa terminará cuando se ingrese "x" y deberá imprimir el total de metros avanzados antes de salir.

*Nota:* Podés (y recomendamos) usar la función `ReadKey` para detectar el espacio. Acá te dejamos un ejemplo de cómo se usaría esta función.

```
char hola;
hola = Console.ReadKey().KeyChar;
if (hola == ' ') Console.WriteLine("hola");
```

- 4) Modificar lo que consideres necesario en este código para que funcione correctamente. La idea es que luego de aparecer el nombre en pantalla, aparezca el saludo de bienvenida.

```
static void Main(string[] args)
{
    int Persona;
    persona = Console.ReadLine();
    Console.WriteLine(persona);
    Console.GetKey();
}

static void printBienvenida()
{
    Console.Write("¡Hola! Bienvenido a esta gigantesca fiesta. Hay toboganes de agua, golosinas y un jacuzzi. ¡Podés usar lo que quieras. Gracias por venir.");
    return 0;
}
```

- 5) Se deberá hacer un menú que muestre las opciones "1 - Iniciar sesión", "2 - Registrarse" y "3 - Salir". Habrá un único usuario válido.
  - a) "Iniciar sesión" -> verifica si el usuario es correcto.
  - b) "Registrarse" -> modifica el usuario válido. Lo reemplaza por lo que se ingrese.
  - c) "Salir" -> termina el programa.

En ningún momento del main se podrá usar "`Console.WriteLine`" (pero si se pueden usar en otras funciones, jijiji).

6) Teniendo el siguiente programa:

```
static void Main(string[] args)
{
    for(int i = 0; i<5; i++)
    {
        pruebaPrint();
    }
    Console.ReadKey();
}

//-----
static void pruebaPrint()
{
    int num = 0;
    num++;
    Console.WriteLine(num);
}
```

¿Qué se mostrará en la pantalla? Comprobarlo ejecutando el código. ¿Pasó lo que esperabas? ¿Qué conclusiones sacás de esto?

## Para seguir aprendiendo...

¿Qué pasa si queremos usar una variable tanto en el Main como dentro de una función? Bueno, una forma de hacerlo (aunque no la que recomendamos, en la guía que viene veremos una mejor forma) es utilizar lo que denominamos "variables globales". Si declaramos las variables con la palabra "static" adelante justo arriba dell Main, podemos utilizar esta variable en cualquier función que declaremos (incluído el Main). Te proponemos hacer el ejercicio 2) pero que en vez de ingresar la cantidad de cuadrados, se ingrese el tamaño deseado.

Un ejemplo de las variables globales es:

```
static int conexion = 0;
static void Main(string[] args)
{
    conexion = 5;
    sumar10();
    Console.WriteLine(conexion);
    Console.ReadKey();
}
static void sumar10()
{
    conexion += 10;
}
```