



## **Asignatura**

DESARROLLO DE SOFTWARE I

## **Tema**

**Evidencias del Curso de C#**

## **Integrantes:**

Díaz Brito, Alex David

## **Maestro:**

LORENZO ANTONIO MARTÍNEZ LEBRÓN

## **ID**

1125848

## **FECHA**

8-10-24

### **1. Introducción al curso (08:08)**

Este video presenta una introducción al curso, donde se explican los fundamentos del lenguaje de programación C# y su historia. El instructor hace un recorrido por las diferentes

aplicaciones de C#, que incluyen el desarrollo de aplicaciones móviles, juegos con Unity, y software de escritorio. Además, se establece la estructura del curso y las herramientas que se utilizarán, como Visual Studio.

- **Conceptos clave:** Se destacan la relevancia de C# en la actualidad, cómo se usa en diferentes plataformas y la importancia de aprender un lenguaje orientado a objetos (OOP).

## 2. Instalar Visual Studio (05:54)

En este episodio, se muestra cómo descargar e instalar **Visual Studio**, el IDE (entorno de desarrollo integrado) que se usará para desarrollar en C#. Se explica el proceso de instalación desde el sitio web de Microsoft, cómo configurar el entorno con las opciones necesarias, y qué características seleccionar para comenzar a programar.

- **Conceptos clave:** La importancia de contar con un entorno de desarrollo completo, configuraciones básicas como la instalación de los paquetes .NET, y cómo elegir plantillas de proyectos adecuadas para el desarrollo en C#.

## 3. Descripción del primer programa de C# (17:12)

Este video descompone el código de un programa básico de C# línea por línea. El instructor explica cada parte, como la función Main(), la estructura básica del programa, la declaración de variables y la ejecución de instrucciones. También se cubre cómo el programa sigue una secuencia lineal de ejecución, desde la primera hasta la última instrucción.

- **Conceptos clave:** Estructura básica de un programa en C#, función Main() como punto de entrada del programa, y la importancia de seguir la sintaxis adecuada.

## 4. Creación del primer programa de C# (16:40)

Aquí se hace un tutorial práctico para escribir y ejecutar el primer programa en C#. Este programa simple imprime un mensaje en la consola usando Console.WriteLine(). Se explica cómo crear un proyecto en Visual Studio, agregar código, compilar y ejecutar el programa.

- **Conceptos clave:** Primeros pasos en la programación de C#, introducción a la consola de comandos, el uso de Console.WriteLine() para imprimir texto en la pantalla, y cómo compilar y ejecutar programas.

## 5. Descripción de los tipos de datos y las variables (28:13)

En esta lección se introducen los **tipos de datos** más comunes en C#, como int, float, bool, y string. Se explica cómo se utilizan estos tipos para almacenar diferentes tipos de información. También se exploran las **variables**, cómo declararlas, inicializarlas y asignarles valores. Además, se hace énfasis en la conversión entre tipos de datos y la importancia de seleccionar el tipo correcto.

- **Conceptos clave:** Variables, tipos de datos (primitivos y no primitivos), la diferencia entre valores y referencias, y cómo gestionar la memoria al declarar variables.

## 6. Instrucción if Decision (23:42)

Se introduce la declaración if, que permite ejecutar código condicionalmente. El instructor explica cómo las decisiones se toman basadas en comparaciones lógicas usando operadores como ==, !=, >, y <. También se aborda la estructura de else para manejar casos en los que la condición no se cumple.

- **Conceptos clave:** Control de flujo con if y else, operadores de comparación y su uso para tomar decisiones dentro del programa.

## 7. For Iteración (instrucción) (18:37)

En esta lección se enseña el ciclo for, una de las estructuras de control de flujo más importantes. El ciclo for se usa para repetir una instrucción o un bloque de código un número específico de veces. El instructor muestra cómo se construye un ciclo for, con una inicialización, una condición y un incremento, y cómo se usa para recorrer arrays o listas.

- **Conceptos clave:** Ciclos de repetición, estructura y sintaxis del ciclo for, y cómo usarlo para tareas repetitivas en programación.

## 8. Descripción de matrices (17:49)

Este video aborda el concepto de **matrices** o arrays, que permiten almacenar múltiples valores del mismo tipo en una estructura indexada. Se enseña cómo declarar, inicializar y acceder a los elementos de un array. También se explica cómo recorrer un array utilizando un ciclo for.

- **Conceptos clave:** Definición de arrays, acceso a los elementos por índice, uso de ciclos para recorrer arrays, y operaciones básicas como la búsqueda y modificación de elementos.

## 9. Definir y llamar a métodos (31:16)

Los **métodos** son bloques de código reutilizables que permiten modularizar el programa. En esta lección, se aprende cómo definir métodos en C#, cómo pasar argumentos a estos métodos y cómo devolver valores desde ellos. El instructor explica la importancia de los métodos para organizar el código y hacerlo más eficiente.

- **Conceptos clave:** Definición de métodos, paso de parámetros, retorno de valores, y la importancia de los métodos para mantener el código organizado y reutilizable.

## 10. Instrucción While Iteración (18:12)

El ciclo while es similar al ciclo for, pero se ejecuta mientras una condición sea verdadera. El instructor explica cómo usar el ciclo while para crear bucles que no tienen un número predefinido de iteraciones, lo cual es útil en situaciones donde el número de repeticiones depende de condiciones dinámicas.

- **Conceptos clave:** Ciclo while, condiciones de salida, comparación con el ciclo for, y ejemplos prácticos de cuándo usar while en lugar de for.

## **11. Trabajar con fechas y horas (10:35)**

Este video explora el tipo de datos DateTime en C#, que permite trabajar con fechas y horas. Se cubre cómo crear objetos de tipo DateTime, cómo formatear fechas y realizar cálculos con ellas, como sumar o restar días. Además, se explica cómo comparar fechas para determinar cuál es más reciente o si dos fechas son iguales.

- **Conceptos clave:** Manejo de fechas y horas en C#, uso del tipo DateTime, formatos de fecha, y operaciones comunes como agregar días, restar tiempos y comparar fechas.

## **12. Trabajar con cadenas (23:39)**

Las **cadenas** son secuencias de caracteres (texto). En esta lección se aprende cómo manipular cadenas de texto en C#. Se cubren operaciones básicas como la concatenación de cadenas, la conversión de texto a mayúsculas o minúsculas, y cómo buscar y reemplazar caracteres o subcadenas dentro de una cadena.

- **Conceptos clave:** Manipulación de cadenas, concatenación, métodos de las cadenas (ToUpper(), ToLower()), y cómo realizar operaciones de búsqueda y modificación de texto.