Introducción

Existen muchos lenguajes de programación en el mundo, los cuales han ido evolucionando con el paso del tiempo y avances del hardware, y con las multiples plataformas y usos de la programación se hace imperante conocer al menos la estructura básica de estos.

Software y hardware han avanzado de la mano impulsados por grandes personajes que sin duda cambiarón la historia de la humanidad.

El proposito de esta investigación es dar a conocer tanto un lenguaje de programación como un personaje destacado en el mundo de la informaticá para ello hemos elegido como lenguaje C# y a Steve Wozniak.

C# se ha convertido en uno de los LDP más usados en la actualidad debido a su flexibilidad en multiplataforma y su fácil migración desde los lenguajes más utilizados para la enseñanza como son C++ y Java.

Stevie Wozniak, coofundador de appl y creador de computadoras como Apple I y Apple II, considerado uno de los padres de la revolución de las computadoras

C#

C# fué lanzado en julio del año 2000 por Microsoft como parte de su plataforma .NET, que hace un énfasis en la transparencía de [redes](https://es.wikipedia.org/wiki/Red_de_computadoras" \o "Red de computadoras), con independencia de plataforma de [hardware](https://es.wikipedia.org/wiki/Hardware" \o "Hardware) y que permite un rápido desarrollo de [aplicaciones](https://es.wikipedia.org/wiki/Aplicaci%C3%B3n_inform%C3%A1tica" \o "Aplicación informática), bajó el liderazgo de Anders Heljsberg, Scott Wiltamuth y Peter Golde. Basado en C++ y Java incorporando , principalmente, la capacidad de combinar operadores propia del primero (no incorpora la herencia múltiple) con la plena orientación a objetos del segundo. La orientación a objetos es tal que el propio programa está encapsulado en una clase. Actualmente C# se encuentra entre los 10 lenguajes más utilizados. Apesar de su corta historia, ha recibido la aprobación del estándar de dos organizaciones: en el 2001 se aprueba el ECMA y en el 2003 el ISO.

<https://es.scribd.com/doc/9018328/Historia-C>

El estándar ECMA-334 lista las siguientes metas en el diseño para C#:

-Lenguaje de programación orientado a objetos simple, moderno y de propósito general.

-Inclusión de principios de [ingeniería de software](https://es.wikipedia.org/wiki/Ingenier%C3%ADa_de_software" \o "Ingeniería de software) tales como revisión estricta de los tipos de datos, detección de intentos de usar variables no inicializadas, y recolección de basura automática.

-Capacidad para desarrollar [componentes de software](https://es.wikipedia.org/wiki/Componente_de_software" \o "Componente de software) que se puedan usar en ambientes distribuidos.

-Portabilidad del código fuente.

-Fácil migración del programador al nuevo lenguaje.

-Soporte para [internacionalización](https://es.wikipedia.org/wiki/Internacionalizaci%C3%B3n_y_localizaci%C3%B3n" \o "Internacionalización y localización).

-Permitir crear tanto pequeños proyectos como proyectos a gran escala.

-Ahorro en cuanto a memoria y procesado.

Se utiliza principalmente en el desarrollo de aplicaciones para Windows Phone, desarrollo de paginas web tales como StackOverFlow, cuyo entorno es en base a C#, SQL Server y Windows IIS, también es utilizado en el desarrollo de juegos en tecnologías como Unity 3D, MonoGame y UnityFS.

<http://www.ecma-international.org/publications/files/ECMA-ST/Ecma-334.pdf>

Los principales IDEs disponibles son:

-Visual Studio: Al ser un IDE perteneciente a Microsoft, los creadores de C#, la integración del LDP es completa, es el más usado.

- Visual Studio Code : Utilizado generalmente para crear paginas webs basadas en C#.

-[#develop](http://www.icsharpcode.net/" \t "https://stackify.com/best-csharp-tools/_blank): es un version ligera que sirve como alternatica a Visual Studio Code.

- MonoDevelop :Developed by Xamarin, tiene muchas funciones similares a Visual Studio Code y tambien es multiplataforma . Con MonoDevelop, puedes crear rapidamente aplicaciones de escritorio como web para Windows, Linux y Mac.

-Otros IDEs menos utilizados son Eclipse, Code::Blocks, Project Rider y Rider.

Plataformas suportadas, C# fue creado con el proposito de tener gran portabilidad de codigo fuente por lo que soporta muchas plataformas tales como Microsoft Windows, Unix, Android, iOS, Windows Phone, Mac OS y GNU/Linux.