

Alumno: José Arturo Reyes Roa

ID: 1116842

Materia: IDS347L - TENDENCIAS EN DESARROLLO DE SOFTWARE

Docente: LORENZO SOLANO MARTINEZ

Ficha de F#

I) Completar ficha del lenguaje asignado.

Propósito:

El propósito de F# es principalmente general, por como este lenguaje funciona para una gran variedad de tareas que se podrían hacer en otros lenguajes de programación. Sin embargo se pudo ver que también tiene algunas funcionalidades específicas en las que se destaca como la computación científica, financiera y el análisis de datos.

Paradigma:

F# es multiparadigma:

- **Funcional:** permite el uso de funciones inmutables para conseguir los resultados esperados además de que maneja la recursividad, para evitar el uso de bucles.
- **Imperativo:** permite el uso de bucles, variables mutables, y otros tipos de solicitudes comunes en varios lenguajes de programación.
- **Orientado a objetos:** permite declarar clases y objetos, soportando funciones como la herencia y el polimorfismo, además de otras características de este paradigma.
- **Concurrente asíncrono:** permite las tareas asíncronas y flujo de trabajo asíncrono.

Fortalezas y debilidades:

Fortalezas:

- Debido a que es un lenguaje funcional, permite crear código eficiente y dinámico, de una forma clara y concisa.
- Tiene interoperabilidad con .NET, lo cual permite el acceso a todas sus bibliotecas de funciones.
- Al ser concurrente asíncrono también ayuda al rendimiento y escalabilidad de las aplicaciones.
- Detecta errores en el código, facilitando su corrección.
- Es capaz de hacer inferencias de tipos de variable.
- Tiene alta compatibilidad entre sistemas operativos.

Debilidades:

- Tiene una curva de aprendizaje un poco complicada, sobre todo para el que no está acostumbrado al uso de funciones.

- Al ser un lenguaje no tan popular, no es muy usado en los ámbitos laborales comunes.
- Hay pocos IDE's que soportan de forma robusta F#.
- En comparación a C#, este lenguaje es inferior en algunas tareas en cuanto a eficiencia.
- No tiene muchas bibliotecas especializadas lo cual puede afectar al rendimiento y uso de F#.

Implementación:

F# es un lenguaje de programación compilado, es decir, usa traductores que generan un código de computadora a partir del código fuente, normalmente resultando en un lenguaje de programación que devuelve respuestas en poco tiempo eficientemente.

Runtime / Máquina Virtual

F# se ejecuta en el Runtime de .NET, es decir, en la máquina virtual del entorno de ejecución CLR (Common Language Runtime).

Plataformas soportadas:

Windows, macOS y Linux, FreeBSD, OpenBSD, DragonFly BSD, Solaris. (Cualquier Sistema que soporte .NET)

Usuarios notables:

Microsoft (lo usa en Azure), JetBrains, instituciones financieras (recordando que F# se destaca en tareas financieras, así que es común que estas la usen).

Historia:

Creador: Don Syme, líder y figura principal en su creación, es un investigador británico y que trabaja actualmente en Microsoft Research Cambridge.

Primer Release: fue la versión F# 1.0, en mayo de 2005. Esta incluyó las bases del lenguaje, al igual que un conjunto inicial de características y herramientas de la programación funcional y orientada a objetos en el entorno de .NET.

Último Release: es la versión F# 8.0, el 14 de noviembre del 2023. Esta versión incluye varias mejoras y características que permiten hacer programas más simples y uniformes, es decir cambios de calidad de vida (QoL).

Estado: Activo, Muerto, etc.

El estado actual de F# es activo, y en continuo desarrollo, además esta en aumento de popularidad, cada vez más desarrolladores lo están usando.

II) Realizar "Hello World" con el lenguaje asignado. (videos)



The image shows a Visual Studio code editor window with a file named 'program.fs'. The code contains three lines: a comment, a line number, and a printf statement. Below the editor, the 'Consola de depuración de Microsoft Visual Studio' window is open, displaying the output 'Hello world' and a detailed error message about the process closing.

```
1 // For more information see https://aka.ms/fsharp-console-apps
2 printfn "Hello world"
3
```

Consola de depuración de Microsoft Visual Studio

Hello world

C:\Users\Arturo\source\repos\Ficha de F#\Ficha de F#\bin\Debug\net8.0\Ficha de F#.exe (proceso 23888) se cerró con el código 0.

Para cerrar automáticamente la consola cuando se detiene la depuración, habilite Herramientas ->Opciones ->Depuración -> Cerrar la consola automáticamente al detenerse la depuración.

Presione cualquier tecla para cerrar esta ventana. . .

III) Demostrar como se interactúa con el Environment con el lenguaje asignado. (videos)

Ver en el código

Argumentos de línea de comando

Standard Streams: Standard Input, Output y Error

Variables de ambiente

File I/O

Network I/O