****

**CENTRO UNIVERSITARIO TECNOLÓGICO**

**CEUTEC**

**ASIGNATURA:**

**INTRODUCCIÓN AL DESARROLLO DE APLICACIONES WEB**

**DOCENTE:**

**CHRISTIAN SÁNCHEZ**

**TEMA DE LA TAREA:**

**TYPESCRIPT**

**ELAVORADO POR:**

**ALLAN HUMBERTO MENDOZA GOMEZ**

**SECCIÓN: T1026**

**06 DE AGOSTO DE 2024**

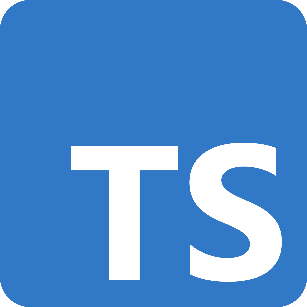
**TEGUCIGALPA, HONDURAS**

**¿Qué es TypeScript?**

TypeScript es una extensión al lenguaje de programación JavaScript que se caracteriza por ampliar su sintaxis en el ámbito de los tipos. En este sentido, es un lenguaje de programación propio que se basa en JavaScript para darnos herramientas de desarrollo en cualquier escala de proyectos.

Además de agregar elementos a la sintaxis de JavaScript, TypeScript se conecta de manera más profunda con los editores de código, revisando errores de manera más oportuna. Una prueba de esto es su integración con Angular, View y React, plataformas fundamentales para un desarrollador web. Mientras que Angular trabaja de manera exclusiva con este lenguaje, View y React tienen la opción de convertir nuestra forma de programar a este lenguaje.

Al igual que JavaScript, este es un lenguaje de programación que recibe mucho soporte por parte de su creador. Además, con cada año que pasa vemos que la comunidad que usa este lenguaje crece cada vez más.

**Logo:**

**Resumen breve de su historia y evolución:**

TypeScript es un lenguaje de programación desarrollado por Microsoft que se lanzó por primera vez en octubre de 2012. Fue creado por Anders Hejlsberg, conocido por su trabajo en C# y Turbo Pascal. TypeScript es un superconjunto de JavaScript que añade tipos estáticos opcionales

TypeScript ha evolucionado significativamente desde su lanzamiento inicial y ha introducido muchas características nuevas a lo largo de los años. Algunas de las características más destacadas incluyen:

**Tipos opcionales y anotaciones de tipo**

TypeScript permite especificar tipos de datos opcionales y anotaciones de tipo para variables, parámetros de función y valores de retorno. Esto ayuda a mejorar la claridad y la seguridad del código.

**Interfaces y clases**

TypeScript permite definir interfaces y clases, lo que facilita la implementación de patrones de diseño orientados a objetos y la creación de código más modular y reutilizable.

**Módulos y namespaces**

TypeScript proporciona soporte para módulos y namespaces, lo que permite organizar el código de manera más modular y facilita la reutilización de código en diferentes partes de una aplicación.

**Decoradores**

Los decoradores son una característica poderosa que permite agregar metadatos y comportamientos adicionales a clases, métodos y propiedades en tiempo de compilación.

**Diferencias entre TypeScript y JavaScript.**

* TypeScript es compilado y Javascript es interpretado de cara al desarrollador, (importante Javascript trabaja con un proceso llamado JIT: Just In Time Compiler, entonces no es 100% interpretado).
* TypeScript es de tipado de datos fuerte, es decir toda variable debe tener su tipo definido, a diferencia de Javascript que es libre de tipado o de tipado débil.
* Mejor y mayor implementación del paradigma de la programación orientada a objetos, con TypeScript es más sencillo y cómodo escribir código orientado a objeto, permite Clases, Herencia, Interfaces (que Javascript no soporta).
* Detección de errores, tomando en cuenta las 2 diferencias anteriores, TypeScript nos va a permitir generar un código más robusto, con menor tasa de errores, al ser compilado, cualquier error de sintaxis será detectado antes de ser generado el código Javascript (¿a ver, pero vamos a generar código Javascript al final? te estaras preguntando esto), al tener tipado fuerte, vamos a evitar problemas de inconsistencia en la gestión de variables.

**Diferencias entre TypeScript y C#.**

* TypeScript es un superconjunto de JavaScript que añade tipado estático, lo que ayuda a encontrar errores durante el desarrollo. Es popular para aplicaciones web y se integra bien con frameworks como Angular.
* C# (C Sharp) es un lenguaje de programación desarrollado por Microsoft, principalmente para aplicaciones de escritorio y desarrollo web usando.NET. Es conocido por su sintaxis clara y fuerte soporte para la programación orientada a objetos.

Ambos tienen sus puntos fuertes y se usan en diferentes contextos, dependiendo del proyecto y las necesidades del desarrollador.

**¿En qué se frameworks frontend se puede usar TypeScript?**

TypeScript se puede usar con varios frameworks frontend.

* React
* Angular
* Vue.js
* Svelte
* Ember.js

**Enlace de documentación oficial del lenguaje de programación**

<https://www.typescriptlang.org/docs/>

**Porcentaje de uso en el mundo**

Según las encuestas y estudios de mercado recientes, TypeScript es utilizado en aproximadamente el 25% de los proyectos frontend.

**Muestre ejemplos de al menos 4 variables con este lenguaje de programación**

let edad: number = 23;  
let nombreCompleto: string = “Allan Humberto Mendoza Gomez”;

let programador: Boolean = true;

let edades: number[] = [19, 25, 50];