



Universidad Nacional de Villa Mercedes

ESCUELA DE INGENIERÍA Y CIENCIAS AMBIENTALES

Ejercicio N°1 – Aplicando los aprendido 2

Carrera: Ingeniería en Sistemas de Información.

Materia: Paradigmas de Programación.

Profesor/res: Walter Molina, Diego Puertas.

Alumno: Germán Adrián Muñoz.

Fecha de entrega: 08/10/2023

Ejercicio N°1

Considera el lenguaje TypeScript acotado al paradigma de programación estructurada y analízalo en términos de los cuatro componentes de un paradigma mencionados por Kuhn.

- 1. Generalización simbólica: ¿Cuáles son las reglas escritas del lenguaje?**
- 2. Creencias de los profesionales: ¿Qué características particulares del lenguaje se cree que sean "mejores" que en otros lenguajes?**
- 3. Valores: ¿Qué pensamiento o estilo de programación consideraron mejor los creadores**
- 4. Ejemplares: ¿Qué clase de problemas pueden resolverse más fácilmente en el lenguaje?**

RESPUESTA

Lenguaje TypeScript

1. Las reglas escritas del lenguaje son:
 - TypeScript es un superconjunto de JavaScript, por lo que muchas de las reglas escritas son compartidas con JavaScript.
 - TypeScript agrega el sistema de tipos estáticos, lo que significa que debes declarar explícitamente los tipos de datos de las variables y funciones.
 - Se utiliza la sintaxis de declaración de variables utilizando let, var, o const, al igual que JavaScript.
 - TypeScript también admite estructuras de control como if, else, for, while, switch, entre otras.
 - Se pueden definir funciones y clases.
 - Manipulación de tipos de datos como números, cadenas de caracteres, booleanos, arreglos y objetos.
 - Utilización de operadores aritméticos, lógicos, de asignación y de comparación, similares a JavaScript.
2. Las características que se cree que son mejores son:

- TypeScript agrega un sistema de tipos estáticos que ayuda a detectar errores en tiempo de compilación, lo que mejora la robustez del código.
 - Proporciona ventajas en el desarrollo a gran escala, ya que los tipos fuertes pueden prevenir errores comunes.
 - Ofrece mejor documentación y autocompletado en entornos de desarrollo, lo que facilita la escritura de código.
 - Compila a JavaScript, lo que lo hace compatible con todos los navegadores y entornos de ejecución de JavaScript.
 - Tiene una comunidad de desarrollo activa y creciente.
 - Se puede utilizar tanto en el lado del cliente como en el lado del servidor.
3. Pensamiento o estilo de programación considerado mejor por los creadores:
- Los creadores de TypeScript se enfocaron en mejorar la calidad y la seguridad del código JavaScript mediante el sistema de tipos estáticos.
 - Promovieron la escritura de código más robusto y mantenible a medida que los proyectos crecían en complejidad.
 - TypeScript se diseñó para ser una extensión natural de JavaScript, lo que significa que los desarrolladores de JavaScript pueden adoptarlo con relativa facilidad.
4. Clase de problemas que pueden resolverse más fácilmente en TypeScript:
- TypeScript es especialmente útil en proyectos de desarrollo web y aplicaciones grandes o complejas.
 - Ayuda a prevenir errores relacionados con tipos de datos, lo que es crucial en aplicaciones críticas.
 - Facilita la creación y el mantenimiento de código escalable.
 - Se utiliza comúnmente en el desarrollo de aplicaciones de una sola página (SPA) con bibliotecas y frameworks como Angular y React.

- Es adecuado para proyectos de servidor utilizando Node.js, ya que TypeScript se puede usar tanto en el lado del cliente como en el lado del servidor.
- Excelente elección para el desarrollo de aplicaciones empresariales o proyectos de gran envergadura donde la calidad del código es esencial.