*Lenguaje de programación RUBY*

**Historia:**

Ruby es un lenguaje de programación orientado a objetos creado por Yukihiro "Matz" Matsumoto en Japón, a mediados de la década de 1990. Matsumoto diseñó Ruby para combinar la simplicidad y la elegancia de la sintaxis con la potencia de la programación orientada a objetos. La primera versión pública, Ruby 0.95, fue lanzada en 1995. A lo largo de los años, Ruby ganó popularidad, especialmente con la creación del influyente framework de desarrollo web Ruby on Rails en 2005. Ruby se destacó por su filosofía de "la felicidad del programador" y su comunidad activa, dejando una marca significativa en la programación moderna

*Lenguaje de programación typescript*

**Historia:**

TypeScript es un lenguaje de programación desarrollado por Microsoft, presentado por primera vez en 2012. Creado por Anders Hejlsberg, conocido por su trabajo en Delphi y C#, TypeScript surgió como una extensión de JavaScript para abordar las complejidades y desafíos de desarrollo en proyectos más grandes. Incorpora características de tipado estático opcional y un sistema de tipos avanzado, lo que permite a los desarrolladores detectar errores antes de tiempo y mejorar la mantenibilidad de las aplicaciones JavaScript. Con el tiempo, TypeScript ha ganado popularidad en el desarrollo web y de aplicaciones debido a su interoperabilidad con JavaScript y su enfoque en mejorar la calidad del código.

*Comparación de rendimiento (benchmark)*

Comparar el rendimiento de lenguajes de programación como Ruby y TypeScript en algoritmos específicos como métodos de búsqueda en colecciones o métodos de ordenamiento puede ser un desafío debido a las diferencias inherentes en la naturaleza de estos lenguajes.

**Método de Búsqueda en Colecciones:**

En Ruby, la búsqueda en colecciones generalmente se basa en iteración y métodos como each o find. Estos métodos pueden resultar cómodos para escribir y leer, pero pueden ser menos eficientes en comparación con enfoques más optimizados.

# Búsqueda de un elemento en un arreglo

numbers = [1, 2, 3, 4, 5]

found\_number = numbers.find { |num| num == 3 }

puts found\_number

En TypeScript, utilizando estructuras de datos y tipos adecuados, la búsqueda en colecciones puede ser más rápida y segura, especialmente si se aprovecha el sistema de tipos estáticos para optimizar los accesos a los elementos.

// Búsqueda de un elemento en un arreglo

const numbers: number[] = [1, 2, 3, 4, 5];

const foundNumber = numbers.find(num => num === 3);

console.log(foundNumber);

**Método de Ordenamiento:**

En Ruby, el método de ordenamiento común es sort, que utiliza algoritmos de ordenamiento eficientes, pero conlleva un cierto costo de rendimiento debido a la manipulación de objetos y el uso de comparaciones.

numbers = [3, 1, 4, 1, 5, 9, 2, 6, 5, 3, 5]

sorted\_numbers = numbers.sort

puts sorted\_numbers

En TypeScript, al igual que en otros lenguajes compilados, el uso de algoritmos de ordenamiento puede ser más rápido debido a la optimización del compilador y la gestión de memoria.

# Ordenamiento de un arreglo

En TypeScript:

const numbers: number[] = [3, 1, 4, 1, 5, 9, 2, 6, 5, 3, 5];

const sortedNumbers = numbers.sort((a, b) => a - b);

console.log(sortedNumbers);

El rendimiento en estas situaciones puede depender de varios factores, como la implementación de los algoritmos en cada lenguaje, la eficiencia en la manipulación de objetos y la gestión de memoria. Ruby tiende a ser menos eficiente en rendimiento puro debido a su naturaleza interpretada y la abstracción sobre el código de máquina. TypeScript, al ser transpilado a JavaScript, a menudo se beneficia de la optimización de los motores de JavaScript y puede ofrecer mejor rendimiento en ciertos casos.

Conclusión sobre los ejemplos dados:

En estos ejemplos, se realizan búsquedas y ordenamientos básicos en arreglos de números. Hay que tener en cuenta que estos ejemplos son simplificados y no representan la totalidad de las capacidades de Ruby y TypeScript en términos de optimización y rendimiento. Los algoritmos reales de búsqueda y ordenamiento pueden ser más complejos y dependen de las características específicas de cada lenguaje.

Fuente:

[Programación en Ruby/Historia - Wikilibros (wikibooks.org)](https://es.wikibooks.org/wiki/Programaci%C3%B3n_en_Ruby/Historia#:~:text=El%20lenguaje%20fue%20creado%20por,%22Perl%22%20(perla).)

[Que es Ruby y Otros Detalles | Blog de Programación y Desarrollo | Nube Colectiva](https://blog.nubecolectiva.com/que-es-ruby-y-otros-detalles/#:~:text=Ruby%20es%20un%20lenguaje%20de,con%20licencia%20de%20Software%20Libre.)

[Fundamentos de TypeScript 🦆| #1: ¿Qué es TypeScript? - DEV Community](https://dev.to/duxtech/fundamentos-de-typescript-1-que-es-typescript-488h#:~:text=%7C%20%2313%3A%20Interfaces-,Qu%C3%A9%20es%20TypeScript,precursor%20del%20proyecto%20Anders%20Hejlsberg.)