Tipado de lenguajes y framework

DigitalHouse >
Coding School

Certified Developer
The Ultimate Tech Degre

Índice

1. Tipado débil
2. Tipado fuerte
3. Tipado estático
4. Tipado dinámico
5. Frameworks

Los **lenguajes tipados fuerte** y **débil** se distinguen según si permiten o no violaciones de los tipos de datos una vez declarados.

1 Tipado débil

Lenguajes de tipado débil

En estos lenguajes no indicamos, la mayoría de las veces, el tipo de variable. Aquí podemos asignar, por ejemplo, un valor entero a una variable que anteriormente tenía una cadena. Pero, no solo eso, también podemos operar con variables de distintos tipos.

Su principal **ventaja** es que es mucho más rápido de desarrollar, pero una clara **desventaja** es que podemos cometer muchos más errores si no tenemos cuidado.

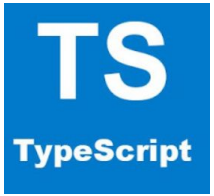
2 | Tipado fuerte

Lenguajes de tipado fuerte

En estos lenguajes se nos obliga a indicar el tipo de dato al declarar la variable. Además, dicho tipo no puede ser cambiado una vez definida la variable.

La **ventaja** es que al ser código más expresivo, cometeremos menos errores.

La **desventaja** es que son mucho más estrictos a la hora de programar y que hay que escribir mucho más código.



Go

3 | Tipado estático

Lenguajes de tipado estático

En el tipado estático, la **comprobación** de tipificación se realiza **durante la compilación** y no durante la ejecución. Comparado con el tipado dinámico, el estático permite que los errores de tipificación sean detectados antes y que la **ejecución** del programa sea **más eficiente y segura**.

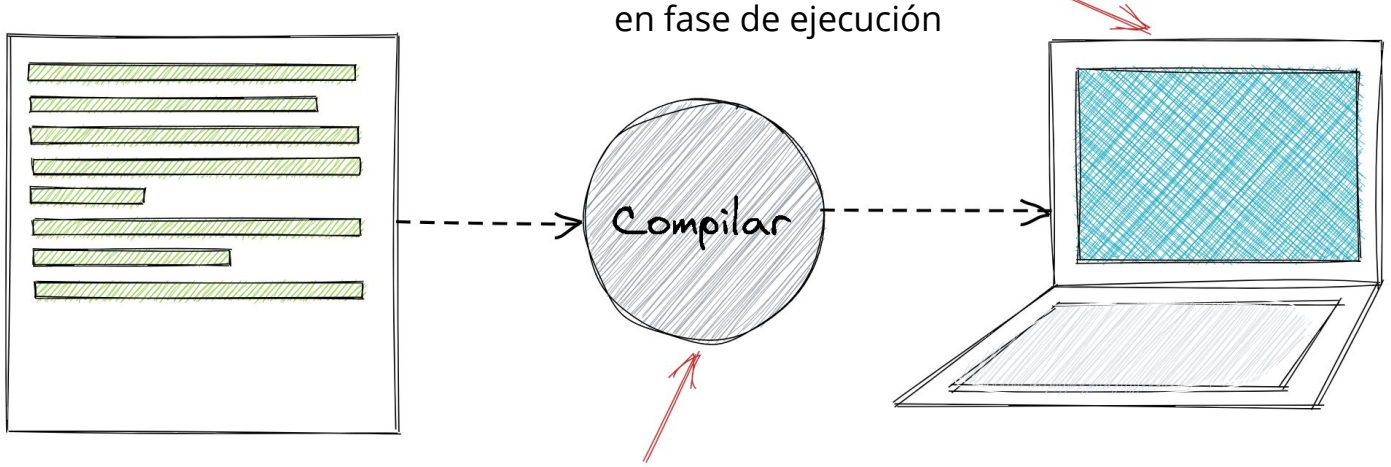
4 | Tipado dinámico

Lenguajes de tipado dinámico

La comprobación de tipificación se realiza durante su ejecución en vez de durante la compilación. Comparado con el tipado estático, este es más flexible, a pesar de ejecutarse más lentamente y ser más propenso a contener errores de programación.

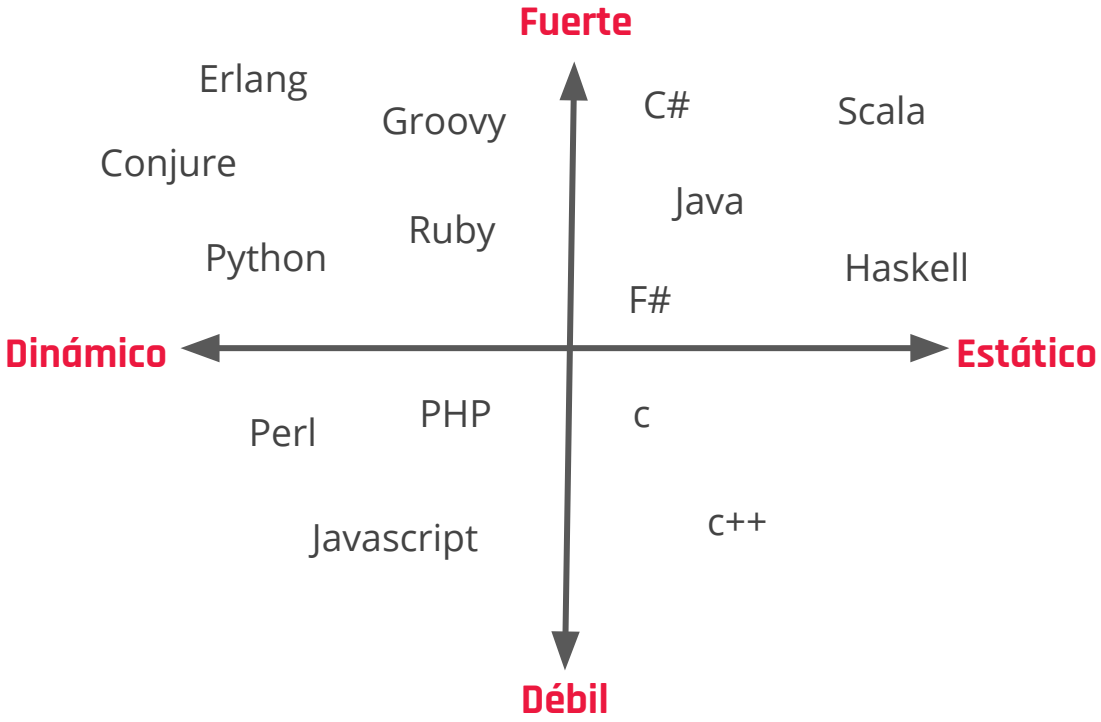
Tipado dinámico y tipado estático

Tipado dinámico
Se comprueban los tipos
en fase de ejecución



Tipado estático
Se comprueban los tipos
en fase de compilación

Posicionamiento de cada lenguaje de programación



5 | Frameworks

Frameworks ~ Marco de Trabajo

Es una estructura previa / esqueleto que se puede aprovechar para desarrollar un proyecto.

El Framework es una especie de plantilla, un esquema conceptual, que simplifica la elaboración de una tarea, ya que solo es necesario complementarlo de acuerdo a lo que se quiere realizar.


DigitalHouse>
Coding School