Los lenguajes de programación compilado:

Son lenguajes de alto nivel que requieren que las instrucciones (código fuente del programa), sean traducidas, -mediante un programa compilador-, a un lenguaje que entienda la máquina (lenguaje máquina), con el fin de generar una versión ejecutable del programa. Ejemplo de lenguajes compilados son Pascal, C, C++, Cobol, Fortran, entre otros.

Lenguaje interpretado:

Es aquel que se envía directamente a la computadora, no se transforma en el camino, por lo que cuando llegue a la máquina tiene que hacer algo llamado: interpretación. Es decir, el software se ejecutará y transformará ese código a instrucciones para la computadora en ese mismo momento. Entre ellos están JavaScript, [PHP](https://ed.team/cursos/php), [Python](https://ed.team/cursos/python), [Ruby](https://ed.team/cursos/ruby) y muchos más.

1. Los lenguajes compilados son lenguajes de alto nivel en los que las instrucciones se traducen a código máquina, creando un archivo "traducido" para una ejecución rápida. Por el contrario, un lenguaje interpretado es aquel en el que las instrucciones se traducen o interpretan una a una, cada vez que se ejecute el programa. Los lenguajes interpretados son típicamente unas 10 veces más lentos que los programas compilados.
2. EL lenguaje interpretado es mucho menos eficiente que la ejecución de un programa compilado, tampoco es eficiente en tiempo ya que cada instrucción debe pasar por una interpretación en tiempo de ejecución.
3. El lenguaje interpretado tiene más flexibilidad frente a los lenguajes compilados, más facilidad para lograr independencia de plataformas (portabilidad) y menor tamaño de programa.

Los programas interpretados suelen ser más lentos que los compilados debido a la necesidad de traducir el programa mientras se ejecuta, pero a cambio son más flexibles como entornos de programación y depuración lo que se traduce, por ejemplo, en una mayor facilidad para reemplazar partes enteras del programa o añadir módulos completamente nuevos.