

## تفاوت زبان کامپایلر و مفسر چیست؟

برای اینکه تفاوت زبان برنامه نویسی کامپایلر و مفسر را بهتر بدانیم و درک کنیم نیاز داریم که اول بدونیم چی هستن به نحوه اجرای کدها منابع نیاز این دو زبان وسرعت اجرا توجه کنیم.

### زبان کامپایلر Compiler

مجموعه‌ای از برنامه یا برنامه‌های کامپیوتری هستند که زبان برنامه‌نویسی سطح بالا (زبان مبدا) را به زبانی سطح پایین (زبان مقصد) و قابل فهم برای سخت افزار تبدیل می‌کند. به عبارتی دیگر کدهای ماشین به سادگی قابل فهم برای انسان نیستند و به همین خاطر انسان به زبان‌های سطح بالا که قابل فهم تر هستند، برنامه می‌نویسد و با استفاده از کامپایلرها آن را به زبان سطح پایین تبدیل می‌کند اینطوری که کدهای نوشته شده توسط برنامه نویسان رو تبدیل به ۰ و ۱ میکند تا به برای سخت افزار یا سی پی یو این کد قابل درک باشد و بتواند آن را اجرا کند.

### زبان مفسر Interpreter

مفسر یک برنامه کامپیوتری است که دستورهای نوشته شده در یک زبان برنامه‌نویسی سطح بالا را به شکل یک کد قابل فهم برای ماشین اجرا می‌کند. بزرگترین مزیتی که مفسر دارد این است که برنامه نوشته شده اصلی، وابسته به نوع سخت افزار نیست. یعنی برنامه‌ای که به زبان مفسر نوشته شود می‌تواند بر روی هر نوع سیستم عاملی اجرا شود.

### تفاوت مفسر و کامپایلر

#### ۱- چگونگی اجرا شدن

در زبان‌های کامپایلری ابتدا یکبار کدها به صورت کامل ترجمه می‌گردند. سپس برنامه برای اجرا از روی فایل کامپایل شده اجرا می‌گردد. اما در زبان‌های مفسری

کدها به صورت خط به خط تفسیر و اجرا می‌شوند. سپس هر دفعه که بخواهیم برنامه را دوباره اجرا کنیم به همین صورت کد تفسیر و اجرا می‌شود.

## ۲- وابسته بودن یا نبودن به سیستم عامل

یکی دیگر از تفاوت‌های این زبان‌ها این می‌باشد که زبان مفسری برای اجرا نیاز به یک نرم‌افزار مفسر یا interpreter دارد. اما زبان کامپایلری تنها یکبار کامپایل شده و کدهای قابل اجرا در ماشین را ایجاد می‌کند. از آن پس بر روی هر سیستمی بدون نیاز به کامپایلر قابل اجرا می‌باشد.

اما همین نکته این تفاوت را ایجاد می‌کند که زبان کامپایلر وابسته به سیستم‌عامل است، یعنی در صورتی اجرا خواهد شد که در آن سیستم‌عامل کامپایل شده باشد و در سیستم‌عامل‌های دیگر اجرا نخواهد شد. اما زبان مفسر در هر سیستم‌عاملی که نرم‌افزار مفسر آن وجود داشته باشد اجرا و تفسیر می‌گردد. در واقع زبان مفسر وابسته به سیستم‌عامل نیست.

## ۳- سرعت و میزان استفاده از حافظه و پردازنده

زبان‌های Interpreter کدها را به صورت خط به خط تفسیر و اجرا می‌کنند، این کار باعث می‌شود تا مقدار زیادی از CPU و RAM کامپیوتر اشغال شود و مشخص است که سرعت پایین‌تری نیز خواهند داشت. اما کامپایل شدن یکباره‌ی زبان‌های کامپایلر این است که در اجراهای متعدد سبک‌تر بوده و سرعت بهتری خواهند داشت.

## ۴- خطایابی

همین مزیت به صورت یکجا کامپایل شدن زبان‌های کامپایلر در بخش خطایابی و رفع مشکل، نقص به حساب می‌آید. زیرا در زبان‌های مفسر هنگامی که برنامه خط به خط اجرا می‌شود، در هر بخشی از برنامه که مشکلی وجود داشته باشد، قابل پیدا کردن و قابل رفع می‌باشد.

اما در زبان‌های Compiler برای رفع یک مشکل در کدها می‌بایست سورس خود برنامه را مجدد بازیابی کرده، سپس خطا را پیدا و تصحیح کرد که این کار بسیار دشوارتر از کار با زبان‌های مفسری است.

## منابع تحقیق

<https://www.manzoumeh.ir/article/show/%D8%AA%D9%81%D8%A7%D9%88%D8%AA-%D8%B2%D8%A8%D8%A7%D9%86-%D9%87%D8%A7-%D8%A8%D8%B1%D9%86%D8%A7%D9%85%D9%87-%D9%86%D9%88%DB%8C%D8%B3%DB%8C-%DA%A9%D8%A7%D9%85%D9%BE%D8%A7%DB%8C%D9%84%D8%B1%DB%8C-%D9%88-%D9%85%D9%81%D8%B3%D8%B1%DB%8C-%D8%AF%D8%B1-%DA%86%DB%8C%D8%B3%D8%AA>

<https://coderlife.ir/%D8%AA%D9%81%D8%A7%D9%88%D8%AA-%D8%B2%D8%A8%D8%A7%D9%86-%DA%A9%D8%A7%D9%85%D9%BE%D8%A7%DB%8C%D9%84%D8%B1-%D9%88-%D9%85%D9%81%D8%B3%D8%B1-%DA%86%DB%8C%D8%B3%D8%AA%D8%9F-%D8%A7%D9%86%D9%88%D8%A7%D8%B9/>

<https://sharifinfo.blogspot.com/1399/10/16/post-4/%D8%A8%D8%B1%D8%B1%D8%B3%DB%8C-%D9%85%D9%81%D8%B3%D8%B1-%D9%88-%DA%A9%D8%A7%D9%85%D9%BE%D8%A7%DB%8C%D9%84%D8%B1%D9%87%D8%A7>