

Feature/Aspect	PHP 5.6	PHP 7	PHP 8
<b>Scalar Type Declarations</b>	No scalar type declarations.	Introduced scalar type declarations ( <code>int</code> , <code>float</code> , <code>string</code> , <code>bool</code> ).	Added support for stricter return type declarations and introduced <code>mixed</code> type.
<b>Return Type Declarations</b>	No return type declarations.	Introduced return type declarations for functions and methods.	Added support for more precise return types such as <code>static</code> , <code>self</code> , and <code>parent</code> .
<b>Nullable Types</b>	No support for nullable types.	Introduced nullable types using <code>?</code> syntax.	Enhanced support for union types, allowing more flexibility in type declarations.
<b>Error Handling</b>	Uses traditional error handling with <code>try-catch</code> for exceptions.	Improved error handling with the introduction of the <code>Throwable</code> interface and the <code>try-catch</code> block for both exceptions and errors.	Added the <code>throw</code> expression, allowing the throwing of exceptions in expressions.
<b>Spaceship Operator</b>	Not available.	Introduced the spaceship operator ( <code>&lt;=&gt;</code> ) for combined comparison.	No significant changes.
<b>Anonymous Classes</b>	Not available.	Introduced support for anonymous classes.	No significant changes.
<b>Closure Call</b>	Limited support for <code>\$this</code> in closures.	Improved support for <code>\$this</code> in closures.	No significant changes.
<b>Generator Return Expressions</b>	No support for value return from generators.	Added support for returning values from generators using the <code>return</code> keyword.	No significant changes.
<b>Performance</b>	Generally slower compared to PHP 7 and 8.	Significant performance improvements, making PHP 7 faster than PHP 5.	Continued performance enhancements in PHP 8, further optimizing execution speed.
<b>Just-In-Time Compilation (JIT)</b>	Not available.	Introduced JIT compilation in PHP 8, improving overall performance.	JIT compilation is available, providing additional performance benefits for certain workloads.
<b>Attributes (Annotations)</b>	Not available.	Introduced attributes (metadata) in PHP 8.	Attributes continue to be supported for adding metadata to classes, methods, and properties.
<b>Union Types</b>	No support for union types.	Introduced union types, allowing more flexibility in type declarations.	Union types are further enhanced, providing more expressive type hinting.
<b>Match Expression</b>	Not available.	Introduced the <code>match</code> expression in PHP 8, providing a more powerful and concise alternative to <code>switch</code> .	No significant changes.
<b>Null Coalescing Operator (??)</b>	Not available.	Introduced the null coalescing operator ( <code>??</code> ) for concise null-checking.	No significant changes.