



Михаил Губин

PHP 8

Михаил Сергеевич Губин
PHP 8. Новинки языка и
программы для работы с ним

http://www.litres.ru/pages/biblio_book/?art=57487801

ISBN 9785005138330

Аннотация

PHP – один из самых простых в освоении языков программирования. С помощью учебника PHP и правильной мотивации вы можете написать первые скрипты и выполнить команды в течение нескольких часов.

Содержание

Язык PHP	5
Коротко о главном	5
PHP 8 и Windows	8
Новые возможности	10
Типы объединения	10
ЛТ	11
Нулевой безопасный оператор	12
Именованные аргументы	13
Атрибуты	14
Выражение соответствия	16
Конец ознакомительного фрагмента.	17

РНР 8

**Новинки языка и
программы для
работы с ним**

Михаил Сергеевич Губин

© Михаил Сергеевич Губин, 2020

ISBN 978-5-0051-3833-0

Создано в интеллектуальной издательской системе Ridero

Язык PHP

Коротко о главном

PHP – это самый быстрый и широко используемый серверный язык сценариев для создания веб-сайтов и веб-приложений. Есть много веских причин, почему некоторые разработчики предпочитают PHP другим серверным языкам, таким как Python и Ruby. PHP включает в себя подробную документацию, множество готовых к использованию скриптов, огромное сообщество и хорошо поддерживаемые фреймворки. Чтобы облегчить работу PHP, эксперты разработали некоторые инструменты разработки PHP, которые повышают эффективность программирования. PHP-инструменты предназначены для учета идеальной IDE (интегрированной среды разработки), где разработчики PHP могут эффективно создавать привлекательные, инновационные и многофункциональные PHP-проекты.

PHP – один из самых простых в освоении языков программирования. С помощью учебника PHP и правильной мотивации вы можете написать первые скрипты и выполнить команды в течение нескольких часов. Текущая версия PHP (PHP 7) ввела оптимизацию скорости, что привело к улуч-

шению рейтинга. Версия 8 PHP еще больше улучшает это и вводит другие новые функции.

PHP 8 выпущен 26 ноября 2020 года. Это новая основная версия, а это значит, что она внесет некоторые кардинальные изменения, а также множество новых функций и улучшений производительности. Прямо сейчас PHP 8 находится в режиме замораживания функций, что означает, что новые функции больше не могут быть добавлены.

Из-за критических изменений существует большая вероятность того, что вам придется внести некоторые изменения в свой код, чтобы запустить его на PHP 8. Однако если вы всегда были в курсе последних выпусков, обновление не должно быть слишком трудным, так как большинство критических изменений были устаревшими раньше в версии 7.* версии.

Помимо кардинальных изменений, PHP 8 также приносит хороший набор новых функций, таких как JIT-компилятор, типы союзов, атрибуты и многое другое.

Версия PHP 8.0 наверняка поможет разработчикам, упростив сложные задачи и обновив новые функции. Эта книга поможет вам узнать особенности версии PHP 8. Многие php-разработчики работали над этим обновлением, и этот

последний релиз наверняка принесет пользу ИТ-аутсорсинговой компании.

PHP 8 и Windows

Печальная новость для владельцев Windows: Microsoft объявила, что она не будет предлагать поддержку в «любом качестве» для PHP для Windows 8.0, когда он будет выпущен.

Microsoft оказывает поддержку в разработке и создании языка программирования PHP на Windows. Эта поддержка включает в себя разработку исправлений безопасности для PHP и создание собственных сборок Windows.

С PHP 8.0 лидер Microsoft PHP Windows Дейл Хирт заявил в списке рассылки PHP Internals, что они не будут поддерживать PHP 8.0 для Windows.

«Мы знаем, что текущая каденция составляет 2 года с момента выпуска для исправлений ошибок и 1 год после этого для исправлений безопасности. Это означает, что PHP 7.2 выйдет из поддержки в ноябре. PHP 7.3 перейдет в режим исправления безопасности только в ноябре. PHP 7.4 будет продолжать иметь еще один год исправлений ошибок, а затем один год исправлений безопасности. Мы стремимся поддерживать разработку и создание PHP на Windows для 7.2, 7.3 и 7.4 до тех пор, пока они официально поддерживаются. Однако мы

не собираемся поддерживать PHP для Windows в любом качестве для версии 8.0 и выше»

– написал Хирт.

Чтобы внести некоторую ясность в сообщение Хирта, менеджер релизов PHP Сара Големон опубликовала сообщение в Reddit, объяснив, что это не означает, что PHP 8.0 не будет поддерживаться в Windows. Это просто означает, что Microsoft не будет строить и поддерживать его.

Microsoft не заявила, почему они больше не будут поддерживать PHP 8.0, но это может быть связано с обширной поддержкой PHP, уже существующей в подсистеме Windows для Linux (WSL).

Microsoft активно разрабатывает WSL, который позволяет пользователям устанавливать различные дистрибутивы Linux, работающие непосредственно в Windows 10.

Поскольку эти дистрибутивы уже поддерживают PHP 7.4 и будут поддерживать PHP 8.0 после выпуска, Microsoft может счесть ненужным продолжать поддерживать собственную сборку PHP в Windows.

НОВЫЕ ВОЗМОЖНОСТИ

Типы объединения

Учитывая динамическую типизированную природу PHP, существует множество случаев, когда типы объединения могут быть полезны. Типы объединения – это набор из двух или более типов, которые указывают, что любой из них может быть использован.

```
public function foo (FoolBar $input): int|float;
```

Обратите внимание, что «void» никогда не может быть частью типа объединения, так как он указывает на отсутствие возвращаемого значения вообще. Кроме того, обнуляемые союзы могут быть записаны с помощью «|null» или с помощью существующего обозначения»?»:

```
public function foo (Foolnull $foo): void;  
public function bar (?Bar $bar): void;
```

ЖТ

ЖТ – just in time-компилятор обещает значительное повышение производительности, хотя и не всегда в контексте веб-запросов. Мне кажется, что ЖТ не имеет большого значения, если вообще имеет значение для РНР-проектов.

Нулевой безопасный оператор

Если вы знакомы с оператором слияния null, вы уже знакомы с его недостатками: он не работает при вызовах методов. Вместо этого вам нужны промежуточные проверки или полагайтесь на необязательных помощников, предоставляемых некоторыми фреймворками:

```
$startDate = $booking-> getStartDate ();  
$dateAsString = $startDate? $startDate->  
asDateTimeString (): null;
```

С добавлением нулевого безопасного оператора мы теперь можем иметь нулевое коалесцирующее поведение на методах!

```
$dateAsString = $booking-> getStartDate ()? ->  
asDateTimeString ();
```

Именованные аргументы

Именованные аргументы позволяют передавать значения функции, указывая имя значения, так что вам не нужно принимать во внимание их порядок, а также вы можете пропустить необязательные параметры!

```
function foo (string $a, string $b,?string $c = null,?string
$d = null) {/* ... */}
foo (b: 'value b', a: 'value a', d: 'value d');
```

Атрибуты

Атрибуты, обычно известные как аннотации на других языках, предлагают способ добавления метаданных в классы без необходимости разбора docblocks.

Что касается быстрого взгляда, вот пример того, как выглядят атрибуты, из RFC:

```
use App\Attributes\ExampleAttribute;
@@ExampleAttribute class Foo
{
    @@ExampleAttribute public const FOO = 'foo';
    @@ExampleAttribute public $x;
    @@ExampleAttribute public function foo (
        @@ExampleAttribute $bar) {}
}
@@Attribute class ExampleAttribute {
    public $value; public function __construct ($value)
    {$this-> value = $value;
    }
}
```

Обратите внимание, что этот базовый атрибут раньше назывался атрибутом PHP в исходном RFC, но впоследствии был изменен другим RFC. Если вы хотите глубоко погрузиться в то, как работают атрибуты, и как вы можете построить свой собственный.

Также обратите внимание, что синтаксис атрибута все еще

может измениться, он все еще не определен.

Выражение соответствия

Вы можете назвать его старшим братом выражения `switch`: `match` может возвращать значения, не требует операторов `break`, может комбинировать условия, использует строгие сравнения типов и не выполняет никакого принуждения типов.

Конец ознакомительного фрагмента.

Текст предоставлен ООО «ЛитРес».

Прочитайте эту книгу целиком, [купив полную легальную версию](#) на ЛитРес.

Безопасно оплатить книгу можно банковской картой Visa, MasterCard, Maestro, со счета мобильного телефона, с платежного терминала, в салоне МТС или Связной, через PayPal, WebMoney, Яндекс.Деньги, QIWI Кошелек, бонусными картами или другим удобным Вам способом.