Introduction

Months, periods, working days, date parsing and translatable date messages in PHP, with immutable objects and no hidden clock

Anonymous · 05/11/2026

Derafu: Date



Calendar months, periods of any length, working days, strict date parsing and the words that describe a date, built on `DateTimeImmutable`.

Nothing in it changes a date that is given and nothing reads the current time by itself: what depends on “now” takes a PSR-20 clock. The words that describe a date (`Monday`, `3 days ago`) are not written in a language: they are translatable messages, so a language is added with a catalogue and without code.

Install

```
composer require derafu/date
```

It requires PHP 8.5 or later, `derafu/translation` (the messages) and `psr/clock`. It has no other dependency.

The classes

Class	What it is for	Page
`Month`	A month of the calendar (`2025-11`), also as the number `202511`.	Months
`Period`	A range of days: a month, a quarter, a year, or from the 15th to the 14th.	Periods
`DateUnit` and `Weekday`	The units of time, to move a date, and the days of the week.	Date Units
`DateParser`	Read and validate dates, and the periods written as numbers.	Parsing

Class	What it is for	Page
`DateCalculator`	Age, days and months between dates, weeks, lists of years.	Calculations
`WorkCalendar`	Working days, with the holidays that are given to it.	Working Days
`DateFormatter`	Describe dates, months and elapsed time as translatable messages.	Formatting
`SystemClock` and `FixedClock`	The current moment, and a moment that does not change.	Clocks

The errors are in Errors and Translations.

Example

```
use Derafu\Date\Enum\DateUnit;
use Derafu\Date\Service\DateFormatter;
use Derafu\Date\ValueObject\Month;
use Derafu\Date\ValueObject\Period;
use Derafu\Date\ValueObject\WorkCalendar;

// A billing cycle that goes from the 15th to the 14th.
$cycle = Period::starting('2025-11-15', DateUnit::MONTH);   // 2025-11-15/2025-12-14
$next = $cycle->next();                                     // 2025-12-15/2026-01-14

// The tenth working day of a month, with the holidays of the country.
$calendar = new WorkCalendar(['2025-11-01', '2025-12-08']);
$day = $calendar->getNthWorkingDay(Month::of(2025, 12), 10);

// A message that is translated where the language is known.
$message = (new DateFormatter())->formatDate($day);
$message->trans($translator, 'es');   // Lunes, 15 de Diciembre del 2025
```

Last updated on 05/11/2026