Periods

Period, an immutable range of days that can be a month, a quarter, a semester, a year or any cycle

Anonymous · 05/11/2026

Periods

```
use Derafu\Date\ValueObject\Period;
```

A `Period` is a range of days, from a first day to a last day, **both included**. The time of the day is not part of it. It can be any range: a month, a quarter, a year, several months, or from the 15th of a month to the 14th of the next one.

It does not change: moving it gives another period. It is `Stringable` (`2025-11-15/2025-12-14`) and serializes to JSON as `{"start": "2025-11-15", "end": "2025-12-14"}`.

Create

Method	What it does
`Period::between($start, $end)`	From a first day to a last day (dates or texts).
`Period::month($year, $month)`	A month.
`Period::fromMonth($month)`	The period of a `Month`.
`Period::months($start, $end)`	From the first day of a month to the last day of another one.
`Period::quarter($year, $quarter)`	A quarter, from 1 to 4.
`Period::semester($year, $semester)`	A semester, 1 or 2.
`Period::year($year)`	A year.
`Period::starting($date, $unit, $count = 1)`	Some units from a date: it ends the day before the same date of the units that follow.

```
Period::starting('2025-11-15', DateUnit::MONTH);      // 2025-11-15/2025-12-14
Period::starting('2025-01-01', DateUnit::MONTH, 3);   // 2025-01-01/2025-03-31
Period::starting('2025-07-01', DateUnit::YEAR);       // 2025-07-01/2026-06-30
```

An end before the start, a quarter that is not 1 to 4, a semester that is not 1 or 2 and a number of units less than 1 are an `InvalidPeriodException`.

Ask

Method	What it gives
`getStart()`, `getEnd()`	The first and the last day, at the start of the day.
`getStartMonth()`, `getEndMonth()`	The months of both.
`getDays()`	The number of days.
`getMonths()`, `countMonths()`	The months that have days of the period, and how many.
`contains($dateOrPeriod)`	Whether a date, or a whole period, is inside.
`overlaps($period)`, `equals($period)`	Whether they share days, or are the same.

Alignment

`isAlignedTo($unit)` tells if the period starts at the beginning of a unit and ends at its end. A period from January 1 to June 30 is aligned to months, quarters and semesters, and it is not aligned to years. Every period is aligned to days, and a week starts on Monday.

Move and split

```
$quarter = Period::quarter(2025, 4);
$quarter->next();       // 2026-01-01/2026-03-31
$quarter->previous(2);  // 2025-04-01/2025-06-30

Period::semester(2025, 2)->split(DateUnit::QUARTER);
// 2025-07-01/2025-09-30 and 2025-10-01/2025-12-31
```

`next()` and `previous()` give the period that follows or precedes with the same length. A period that is a whole number of months from its first day (a month, a quarter, or from March 15 to April 14) moves by that number of months; any other one moves by the same number of days.

```
Period::between('2025-03-15', '2025-04-14')->next();   // 2025-04-15/2025-05-14
Period::between('2025-01-01', '2025-01-10')->next();   // 2025-01-11/2025-01-20
```

`split($unit, $count = 1)` cuts the period in consecutive periods of that unit, counted from its first day. The last one ends where the period ends, so it can be shorter.

The year

Method	What it does
`isSingleYear()`	Whether it starts and ends in the same year.

Method	What it does
`getYear()`	The year, or an `InvalidPeriodException` if it has days of more than one year.
`requireSingleYear()`	The same period, or an `InvalidPeriodException`.
`startsOnYearStart()`	Whether it starts on January 1.
`isBeforeStartInSameYear($date)`	Whether a date is before the start, in the year of the start.

Last updated on 05/11/2026