

Introduction

Typed array reading, composable validation types and explicit array transformations in PHP

Anonymous · 26/10/2026

Typed Array Reading and Transformations

github **derafu/array** last commit **yesterday**  CI **passing** code size **171.8 KiB** open issues **0**

downloads **44** downloads **44 this month**

derafu/array reads array values with explicit types and provides transformations for paths, numeric text, trees, tables and subsets. It requires PHP 8.5 or later, derafu/value and derafu/translation.

Installation

```
composer require derafu/array
```

Reading Untrusted Data

```
use Derafu\Array\ArrayReader;

$reader = ArrayReader::fromValue(
    json_decode('{ "host": "localhost", "port": 5432 }', true),
    source: 'connection.json',
);

$host = $reader->requireString('host'); // string
$port = $reader->getInt('port', 5432); // int
```

`fromValue()` rejects a value that is not an array. Use `fromArray()` when the input is already typed as an array. Readers are immutable snapshots of array contents; objects stored inside an array remain the same objects.

Reading is strict: `'5432'` is not an integer. Use `getAsXXX()/requireAsXXX()` for explicit field conversions, or `ArrayCaster` for its recursive array operation. Date and backed-enum types also interpret their supported representations.

Reading Nested Values

All typed getters and presence checks accept dot notation. An existing literal key matching the complete expression takes precedence over navigation.

```
$reader = ArrayReader::fromArray(['connection' => ['port' => 5432]]);
$reader->requireInt('connection.port'); // 5432
```

See [Readers](#) for presence and validation rules, and [Paths](#) for the same literal-key precedence when reading, setting and wrapping values.

The Classes

All six array utility classes live in `Derafu\Array`. Composable value types and scalar conversions come from [Derafu Value](#).

Class	Purpose	Guide
<code>ArrayReader</code>	Optional and required typed reads, nested readers and key presence.	Readers
<code>ArrayPath</code>	Nested access, dot notation and wrapping a value in a list.	Paths
<code>ArrayCaster</code>	Recursive value interpretation delegated to <code>ValueParser</code> .	Casting
<code>ArrayTree</code>	Parent-linked rows into trees and trees into outlines.	Trees
<code>ArrayTable</code>	Columns into rows, pairs into maps and keys into columns.	Tables
<code>ArrayCombinations</code>	Subsets of a list.	Combinations

`ArrayReader` is a `final readonly` object. The other array classes expose static transformations. Reader type arguments are instances of `Derafu\Value\ValueType`. Exceptions raised by the library's validation are translatable; see [Translations](#).

Lists, Maps and Arrays

Type	Accepted keys
List	Consecutive integers starting at 0.
Map	Strings only. Numeric strings that PHP converts to integer keys are rejected.
Array	Any PHP array keys: integers, strings or both, consecutive or not.

Empty arrays are accepted by all three. Choose `arrayOf()` when positions or identifiers must be preserved without requiring a list. Reading a nested array as a reader does not require it to be a map.

Last updated on 26/10/2026