Support Project

Essential PHP Utilities

Anonymous · 21/08/2026 · #php

Essential PHP Utilities

A collection of essential PHP utility classes that provide common functionality for string manipulation, array handling, file operations, date management, and more.

Features

- □ String manipulation utilities.
- □ Array handling helpers.
- □ Date and time management.
- □ File system operations.
- □ CSV file handling.
- □ Object manipulation tools.
- □ Debugging utilities.
- □ Object factory and hydration.
- □ Data serialization helpers.
- □ Comprehensive test coverage.

Why Derafu\Support?

This package focuses on solving specific business and data processing needs that are often overlooked by standard PHP utilities:

- **Business Date Handling**: Working days calculation, fiscal period management, and date ranges that understand holidays and weekends.
- **Robust CSV Processing**: Consistent handling of different encodings, separators, and quote styles across systems.
- **Practical Data Transformations**: Convert between different data structures (trees, tables, lists) while preserving data integrity.
- **Real-world File Operations**: Safe file handling with proper error management and automatic MIME type detection.

If your application deals with business dates, data processing, or file management, these utilities can save you from reinventing common solutions.

Installation

Install via Composer:

```
composer require derafu/support
```

Usage Examples

String Manipulation (Str)

```
use Derafu\Support\Str;

// Generate UUID.
$uuid = Str::uuid4();

// Replace placeholders.
$result = Str::format('Hello {{name}}!', ['name' => 'John']);

// Normalize strings for URLs.
$slug = Str::slug('Hello World!'); // "hello-world".
```

Array Handling (Arr)

```
use Derafu\Support\Arr;

// Auto-cast array values.
$result = Arr::cast($array);

// Convert array to tree structure.
$tree = Arr::toTree($items, 'parent_id', 'children');
```

Date Management (Date)

```
use Derafu\Support\Date;

// Add working days.
$newDate = Date::addWorkingDays('2024-01-15', 5, $holidays);

// Format date in Spanish.
$formatted = Date::formatSpanish('2024-01-15'); // "Lunes, 15 de enero del 2024".
```

```
// Calculate periods
$nextPeriod = Date::nextPeriod(202401); // 202402.
```

File Operations (File)

```
use Derafu\Support\File;

// Get file MIME type.
$mime = File::mimetype('document.pdf');

// Compress directory.
File::compress('/path/to/dir');

// Send file through browser.
File::send('document.pdf');
```

CSV Handling (Csv)

```
use Derafu\Support\Csv;

// Read CSV file.
$data = Csv::read('file.csv', ';');

// Generate CSV content.
$csvString = Csv::generate($data);

// Send CSV as download.
Csv::send($data, 'export.csv');
```

Object Manipulation (Obj)

```
use Derafu\Support\Obj;

// Fill object properties.
$object = Obj::fill($instance, $data);

// Get public properties.
$properties = Obj::getPublicProperties($instance);
```

Object Factory and Hydration

```
use Derafu\Support\Factory;
use Derafu\Support\Hydrator;

// Create and hydrate objects.
$instance = Factory::create($data, MyClass::class);
```

```
// Hydrate existing instance.
$hydrated = Hydrator::hydrate($instance, $data);
```

Debug Utilities

```
use Derafu\Support\Debug;

// Inspect variable.
$info = Debug::inspect($var, 'myVar');

// Print debug information.
Debug::print($var);
```

Last updated on 21/08/2026