

Vocabulary

Vocabulary

Anonymous · 09/09/2026

Vocabulary

Every value `EscposPrinter` accepts (justification, print modes, barcode types, etc.) has its own type under `Derafu\Escpos\Enum\*` or `Derafu\Escpos\Constant\*`. You never need to import anything from `Mike42\Escpos\*` to call any method.

These values are **not arbitrary numbers chosen by this library**. They are the ESC/POS protocol itself — the bytes a thermal printer actually expects, defined by the standard and shared by every ESC/POS library regardless of language. That is why they are safe to hardcode here instead of referencing the underlying library's constants: they would not change even if the underlying library were replaced.

Enum vs. constant

- **Enums** (`Derafu\Escpos\Enum\*`) are used when the values are a mutually exclusive choice — you pick exactly one (e.g. left, center or right justification).
- **Constants** (`Derafu\Escpos\Constant\*`) are used when the values are independent bits meant to be combined with `|` (e.g. font + emphasized + double width at the same time). An enum would force unwrapping `->value` on every term just to combine them, and the combined result would no longer be the enum type anyway — a plain `int` constant says what it is more honestly.

Enums

Enum	Cases	Used by
Justification	Left, Center, Right	<code>EscposPrinter::setJustification()</code>
BarcodeType	Upca, Upce, Jan13, Jan8, Code39, Itf, Codabar, Code93, Code128	<code>EscposPrinter::barcode()</code>
Pdf417Option	Standard, Truncated	<code>EscposPrinter::pdf417()</code>
BarcodeTextPosition	None, Above, Below	<i>(vocabulary only — no method exposes it yet)</i>
Font	A, B, C	<i>(vocabulary only)</i>
Color	Color1, Color2	<i>(vocabulary only)</i>

Enum	Cases	Used by
CutMode	Full, Partial	(vocabulary only)
Underline	None, Single, Double	(vocabulary only)
QrErrorCorrectionLevel	L, M, Q, H	(vocabulary only)
QrModel	Model1, Model2, Micro	(vocabulary only)

The “vocabulary only” enums exist because they are a small, closed set defined by the protocol — completing them costs nothing and keeps this package generic instead of shaped only around what its first consumer needed. They are ready for the day a `setFont()`, `setColor()`, `setUnderline()`, `setBarcodeTextPosition()` or `qrCode()` method is added to `EscposPrinter`, without needing new vocabulary at that point.

```
use Derafu\Escpos\Enum\Justification;

$printer->setJustification(Justification::Center);
```

Constants (bitmask flags)

Class	Constants	Used by
PrintMode	FONT_A, FONT_B, EMPHASIZED, DOUBLE_HEIGHT, DOUBLE_WIDTH, UNDERLINE	<code>EscposPrinter::selectPrintMode()</code>
ImageSize	STANDARD, DOUBLE_WIDTH, DOUBLE_HEIGHT	<code>EscposPrinter::image()</code>

```
use Derafu\Escpos\Constant\PrintMode;

// Combine flags with a plain bitwise OR.
$printer->selectPrintMode(PrintMode::FONT_B | PrintMode::EMPHASIZED);
```

What is deliberately not vocabulary

A few things found in `mike42/escpos-php` are **not** mapped, on purpose:

- **Status request codes and raw control bytes** (`STATUS_*`, `ESC`, `GS`, `LF`, ...): these are not parameters of any command a caller chooses — they are either internal bytes the underlying library uses to build each command, or (for `STATUS_*`) not used by any method in the vendored version at all. Mapping them would be vocabulary for a feature that does not exist.
- **Character tables / code pages**: unlike everything above, the correct table number is not a fixed protocol value — it depends on the specific printer’s capability profile. This is a known open item, not yet designed.

Last updated on 09/09/2026