I. Reuse from...

Here are the key points to remember when using palettes similar to those offered by projects listed in the section ??.

1. `@prism` uses standardized `CamelCase` notation. Therefore, palette names such as `berlin` and `gist_heat` become `Berlin` and `GistHeat` respectively.

2. `Matplotlib` palettes with a name ending with `_r` (reversed color order) are not included in `@prism`.

3. The following presents palettes from projects other than `Matplotlib` that have been kept but renamed: $\boxed{\Rightarrow}$ indicates a name modification, with the `@prism` name displayed on the right.

CartoColors	`Prism`	$\Rightarrow$	`PrismCC`
Plotly	`Rainbow`	$\Rightarrow$	`RainbowPly`
Tableau	`Gray`	$\Rightarrow$	`GrayTab`

4. The following palettes are excluded because they duplicate `@prism` palettes either directly or in reversed order, except that exact duplicates (same name and colors) are omitted when they don't come from `Matplotlib`, and we use $\boxed{=}$ for equality, $\boxed{\rightleftharpoons}$ for reversal, and the rightmost palettes are the ones retained in `@prism`.

Cubehelix	`Classic`	$=$	`Cubehelix`
Matplotlib	`GistGray`	$\rightleftharpoons$	`Binary`
	`GistGrey`	$\rightleftharpoons$	`Binary`
	`GistYarg`	$=$	`Binary`
	`GistYerg`	$=$	`Binary`
	`Gray`	$\rightleftharpoons$	`Binary`
	`Grey`	$\rightleftharpoons$	`Binary`
	`Greys`	$=$	`Grays`
Plotly	`D3`	$=$	`Tab10`
cmocean	`Balance`	$=$	`Vik`
	`Gray`	$=$	`Binary`

Caution.

Most `@prism` implementations add the `pal` prefix to standardized `CamelCase` names. See the section ??.

Note.

Most `@prism` implementations provide methods to easily obtain reversed palettes, sub-palettes, and color-shifted palettes. See the section ??.