

Contentview Field for Kirby CMS

A custom field type that displays content in a view-only format, respecting proper accessibility standards.

Installation

1. Create the directory structure in your Kirby project:

 Copy

```
site/plugins/contentview-field/  
├─ index.php  
└─ src/  
    ├─ index.js  
    └─ components/  
        └─ ContentView.vue
```

2. Copy the provided files into their respective locations.
3. Update the plugin namespace in the files from `your-name/contentview-field` to match your project's namespace convention.
4. Make sure the Panel is using the latest version by forcing a reload (clear cache if necessary).

Usage

In your blueprint files, you can use the `contentview` field type:

yaml

 Copy

```
fields:  
  myContent:  
    label: My Content  
    type: contentview  
    value: "{{ page.someField }}"  
    formatted: true # or false for plain text
```

Field Options

- **value:** The content to display (can be a string or a Kirby template tag)
- **formatted:** Whether to preserve HTML/Kirbytext formatting (boolean, default: true)
- **class:** Additional CSS classes to add to the field (string, optional)

Conditional Display

You can use Kirby's conditional display features to show different fields to different users:

```
editorContent:
  label: Content
  type: textarea
  when:
    user.role: [admin, editor]

viewerContent:
  label: Content
  type: contentview
  value: "{{ page.editorContent }}"
  when:
    user.role:
      not: [admin, editor]
```

Customization

You can customize the appearance of the contentview field by modifying the CSS in `ContentView.vue`. The field uses Kirby's design variables for consistency with the Panel's UI.

Troubleshooting

If the field doesn't appear correctly:

1. Check browser console for JavaScript errors
2. Make sure all files are in the correct locations
3. Clear Kirby's cache (`site/cache/`)
4. Reload the Panel