

Akkordeon Plugin für moziloCMS

Version 1.0.6

06.05.2026

Lauffähigkeit:

- ab PHP 7.0.33 bis einschließlich PHP 8.4
- ab mozilo 2.0 (Rev 51) / 3.x

TEIL 1 - SCHNELLEINSTIEG

=====

Dieses Plugin erzeugt ein auf- und zuklappendes Akkordeon für Inhalte.

1.1. Grundsyntax

Das Akkordeon besteht aus zwei Teilen:

- **Container** umschließender Rahmen
Container-Syntax:
{Akkordeon|**Elemente**}
- **Elemente (1 bis n)** bilden Containerinhalt
Grundbausteine des Akkordeons
Elemente bestehen immer aus zwei Teilen
Elemente-Syntax:

--item--

Titel; klickbar; öffnet Inhaltsbox

--content--

aufklappbarer Inhalt;Inhaltsbox

1.2. Anwendung im Editor

Beim Einfügen des Plugins im Editor wird ein Container mit 2 Elementen erzeugt; weitere Elemente können hinzugefügt werden. Es entsteht etwas in dieser Art:

```
{Container ||  
    Element 1  
    Element 2  
... hier können weitere Elemente angelegt werden ...  
}
```

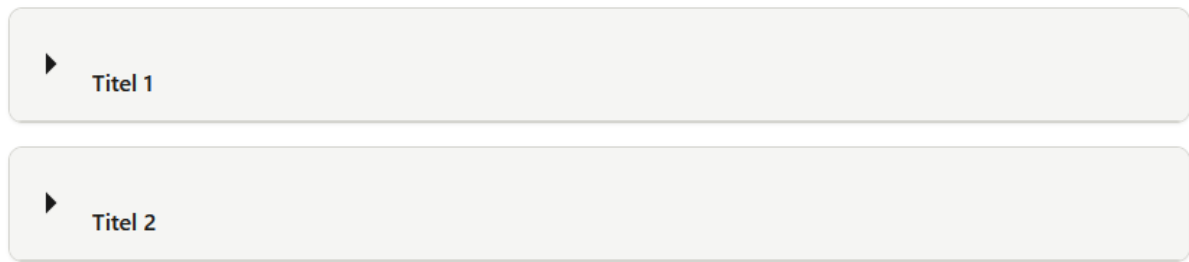
1.3. Das Ganze in mozilo-Syntax:

Beispiel 1

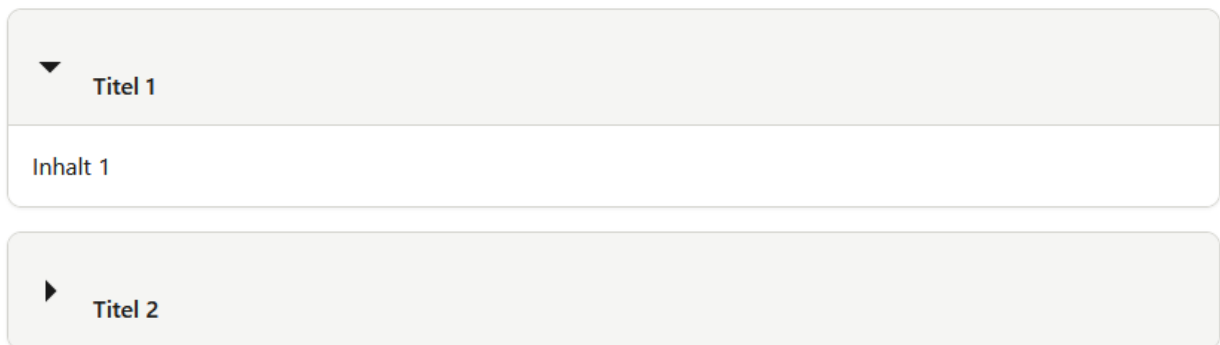
```
{Akkordeon| |  
--item--  
    Titel 1  
--content--  
    Inhalt 1  
  
--item--  
    Titel 2  
--content--  
    Inhalt 2
```

```
... hier weitere Elemente der Form --item-- Titel --content-- Inhalt ...  
}
```

ScreenShot - 2 Elemente mit Titel 1 und Titel 2



ScreenShot - 2 Elemente; Titel 1 geöffnet; Inhalt 1 sichtbar



Hinweis:

--item-- / --content-- | reservierte Namen in exakt dieser Schreibweise

Wichtig:

- **--item--** beginnt ein neues Element; danach steht klickbarer Titel
- **--content--** trennt Titel von nachfolgendem Inhalt
- jedes **Element** benötigt genau ein **--content--**
- mehrere Elemente bilden ein Akkordeon
- mehrere Akkordeons pro Seite sind möglich; alle mit identischer Logik!

1.4. Konkrete Anwendung

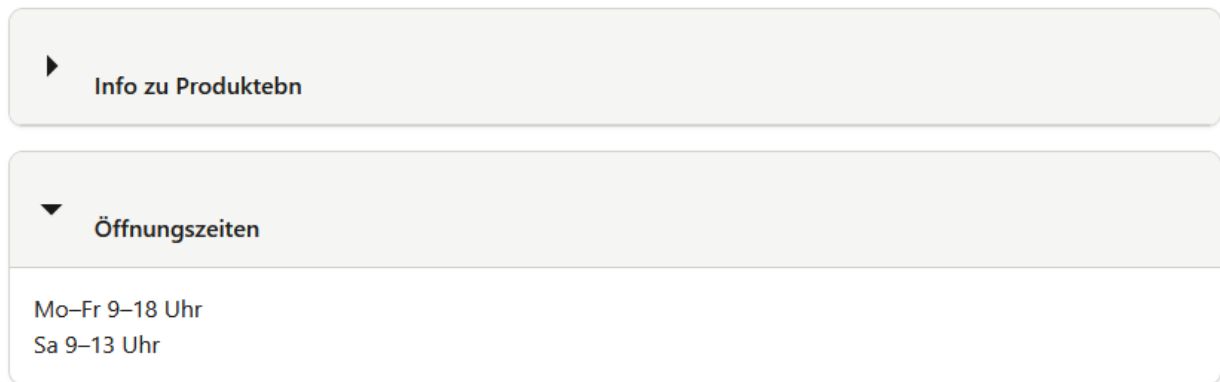
Beispiel 2

```
{Akkordeon||
--item--
Info zu Produkten
--content--
Produktbeschreibung mit Text, Bildern oder Listen

--item--
Öffnungszeiten

--content--
Mo-Fr 9-18 Uhr
Sa 9-13 Uhr
}
```

ScreenShot - 2 Elemente; Titel 2 geöffnet; Inhalt 2 sichtbar



1.5. Standardaufruf - **ohne** ersten Parameter (siehe Beispiele 1+2)

- `{Akkordeon|Elemente}`
Es greifen voreingestellte Formatierungen aus der plugin.css

Formatierungen betreffen:

Breite, Abstand, Farbe, Rahmen, Schatten, Symbol etc.

1.6. Gestaltungsvarianten - **mit** erstem Parameter

- Parameter 1 bekommt feste Begriffe = Schlüsselworte
(Schlüsselworte = Modifikatoren = CSS-Klasse)
- Parameter 1 - **nur ein Eintrag**
`{Akkordeon|volle-breite|Elemente}`
`{Akkordeon|ecken-nicht-rund|Elemente}`
`{Akkordeon|farbschema-sand|Elemente}`
- Parameter 1 kann **mehrere Einträge** bekommen; Trennung mit Leer
`{Akkordeon|volle-breite farbe_dunkel|Elemente}`
`{Akkordeon|schmale-breite farbe-rot|Elemente}`
`{Akkordeon|grosser-schatten farbe-rot|Elemente}`
- Einträge in Parameter Nr. 1 modifizieren das Aussehen des Akkordeons
- Die oben benutzten Begriffe (volle-breite, schmale-breite, farbe-rot etc.) sollen das Prinzip verdeutlichen - existieren so nicht
- Tabelle unten listet die tatsächlich existierenden Begriffe auf mit einer kurzen Erklärung, was ihre Anwendung in Parameter Nr. 1 bewirkt (Punkt 1.7.)

1.7. Schlüsselworte = Modifikatoren (bestimmen Aussehen)

- Namensgebung folgt einheitlich folgendem Schema:
- Es werden 5 Bereiche unterschieden. Aus dem Namen erkennt man, wie die Formatierungen wirken (Layout, Elementgestaltung, Farbe, Inhalt, Symbol vor Titel):
 - akk-layout- - Elementbreite, Abstand, Ausrichtung
 - akk-frame- - Rand, Schatten, Ecken
 - akk-theme- - Farbschema
 - akk-content- - Anordnung Text, Bilder etc. im Inhaltsbereich
 - akk-icon- - Symbol vor Titel

Folgende Schlüsselworte sind bereits integriert und können als Parameter Nr. 1 eingesetzt werden:

Name Schlüsselwort	Beschreibung
Layout	
akk-layout-small	- schmale Breite
akk-layout-medium	- mittlere Breite (Standard)
akk-layout-full	- volle Breite
akk-layout-tile	- Titel groß mit Bild
Frame	
akk-frame-square	- eckig (kein Radius)
akk-frame-elevated	- stärkerer Schatten
Theme	
akk-theme-sand	- hell / beige
akk-theme-dark	- dunkel
Content	
Icon	
akk-icon-klammer	- Büroklammer
akk-icon-dreieck	- Dreieck
akk-icon-pfeilkreis	- Pfeil in Kreis
akk-icon-pfeil	- Pfeil
akk-icon-dreieck-weiss	- Dreieck weiß (auf dunklem Hintergrund nutzen)

Konkretes **Beispiel 3**:

- Akkordeon schmaler akk-layout-small
- Farbe invertiert / dunkel akk-theme-dark
- Symbol, Dreieck weiß akk-icon-dreieck-weiss

```
{Akkordeon|akk-layout-small akk-theme-dark akk-icon-dreieck-weiss|
```

```
--item--
```

```
Öffnungszeiten
```

```
--content--
```

```
Mo - Fr        9:00 - 18:00
```

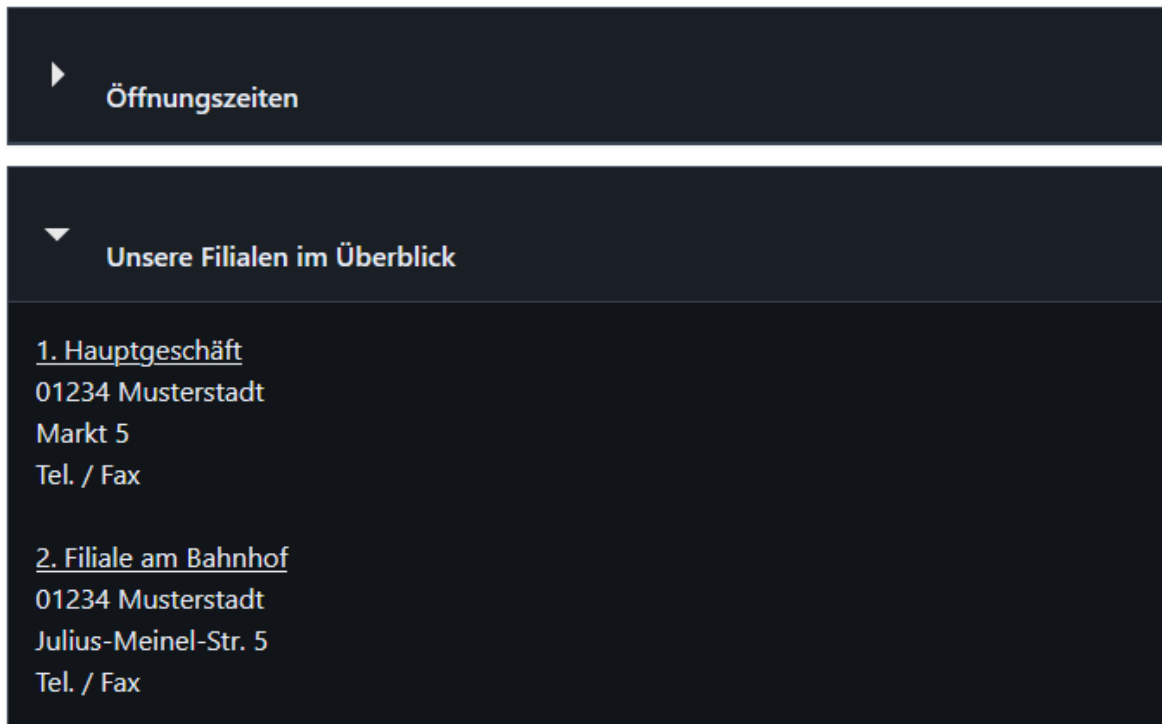
```
Sa             9:00 - 12:00
```

```
-item--
```

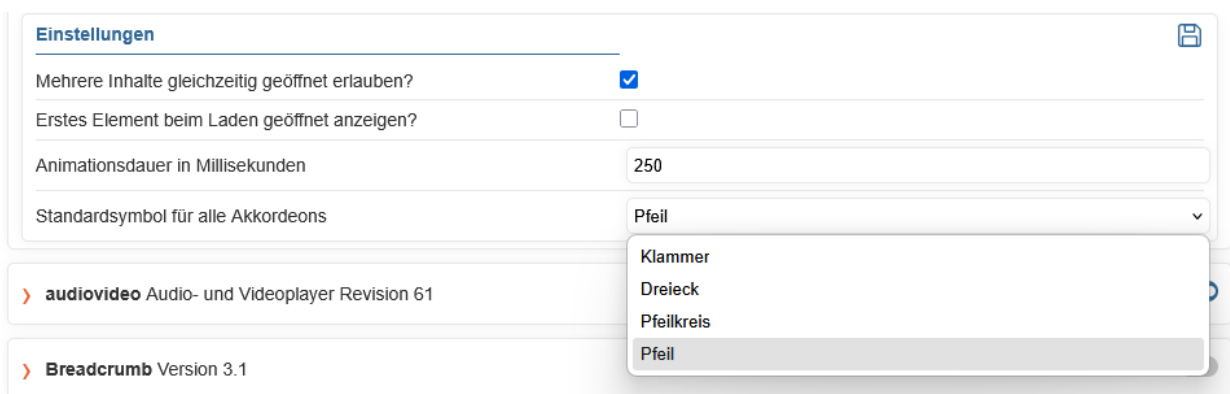
```
Unsere Filialen im Überblick
```

```
--content--
1. Hauptgeschäft
    01234 Musterstadt
    Markt 5
    Tel. / Fax
2. Filiale am Bahnhof
    01234 Musterstadt
    Julius-Meinel-Str. 5
    Tel. / Fax
}
```

ScreenShot - 2 Elemente; Titel 2 geöffnet; Inhalt 2 sichtbar



1.8. Einstellungen im Adminbereich



- Mehrere Inhalte gleichzeitig geöffnet erlauben? J/N
mehrere Elemente eines Akkordeons können gleichzeitig offen sein
ansonsten wird vorher geöffnetes Element automatisch geschlossen
- Erstes Element beim Laden geöffnet anzeigen? J/N
erstes Element jedes Akkordeons ist automatisch geöffnet

- Animationsdauer in Millisekunden 100 - 3000ms
Geschwindigkeit der Animation (Aufklappen / Schließen)
- Standardsymbol für alle Akkordeons
DropDown-Liste mit 4 Möglichkeiten (Büroklammer, Dreieck, Pfeil in Kreis, Pfeil)

HINWEIS!

Die Admin-Einstellungen gelten für alle Akkordeons - alle
Plugineinfügungen auf der gesamten Website!

Damit ist klar: mehrere Akkordeon-Plugins auf der Seite verhalten sich im
Einklang!

- Wenn Auswahl:
„Erstes Element beim Laden geöffnet anzeigen?“
angehakt ist, dann ist eben bei jedem Plugin das erste Element
geöffnet!
- Standardsymbol gilt für alle Akkordeons; kann mit Schlüsselwort für
einzelnes Akkordeon übersteuert werden (z.B. Dreieck weiß auf
dunklem Hintergrund)

1.9. Tipps für die Praxis

- Titel möglichst kurz und klar halten
- Titel kann mehrzeilig sein; Bild möglich - wird verkleinert
- Inhalte können auch Bilder, Listen oder Plugins enthalten

1.10. Typische Fehler

Kein "--item--" gefunden
es wurde kein Element definiert

Kein "--content--"
Trennung zwischen Titel und Inhalt fehlt

TEIL 2 - ERWEITERUNG UND TECHNIK =====

2.1. Konzept

Das Plugin trennt klar zwischen:

- Funktion (Plugin, JavaScript)
- Darstellung (CSS-Klassen und Variablen)

Darstellung wird über Klassen gesteuert, nicht über Admin-Parameter.

2.2. Eigene Klassen

Eigene Klassen können erstellt und verwendet werden:

```
{Akkordeon|meine-klasse|Elemente}
```

Empfohlene Namenskonvention für meine-klasse je nach Bereich:

```
akk-layout-mein_name  
akk-frame-mein_name  
akk-theme-mein_name
```

akk-content-mein_name
akk-icon-mein_name

2.2. Wo werden eigene Klassen angelegt?

A) NICHT empfohlen!

- In plugin.css
- neue Modifikator-Klassen anlegen bzw. vorhandene Klassen mit CSS neu definieren / modifizieren
- plugin.css wird bei Update überschrieben

B) Empfohlen

- In eigener Datei akk-custom.css
- Ist im Auslieferungszustand nicht vorhanden
- Dafür gibt es eine akk-custo-vorlage.css mit Beispielen
- Akk-custom.css einmalig neu anlegen (bzw. akk-custom-vorlage.css kopieren / umbenennen)
- **akk-custom.css wird bei Update nicht überschrieben!**
- akk-custom.css wird bei Existenz automatisch eingebunden

In der **akk-custom-vorlage.css** sind exemplarisch folgende Klassen zusätzlich angelegt:

Name	Schlüsselwort	Beschreibung
Layout		
akk-layout-sidebar		- Akkordeon als schmale linke Spalte
Frame		
akk-frame-flat		- komplett rahmenlos (clean / minimal)
Theme		
akk-theme-blue		- Farbschema Blau
Content		
akk-content-compact		- Kompakter, sehr dichter Inhalt

Vorteil der akk-custom.css:

- Eigene Gestaltungsvarianten werden vom Update nicht angefaßt; bleiben danach voll erhalten
 - Man kann sich von anderen Entwicklern / Bekannten Gestaltungsvarianten anlegen lassen und dies in einer akk-custom.css ablegen; die akk-custom.css dann einfach in das Plugin-Verzeichnis kopieren
- Beispiel:

```
/* -----  
Layout: Akkordeon als schmale linke Spalte  
----- */  
  
.mz-akk-box.akk-layout-sidebar {  
  --mz-akk-max-width: 28rem;  
  margin-left: 0;  
  margin-right: auto;  
}  
- Aufruf im Editor für dieses Beispiel wäre:  
  {Akkordeon|akk-layout-sidebar|Elemente}
```

2.3. CSS-Variablen (Design-System)

Das Akkordeon verwendet CSS-Variablen als zentrale Stellschrauben.

Diese können in den eigenen Klassen überschrieben werden.

Das zentrale Element für die Gestaltung ist:

- Container **.mz-akk-box**
- .mz-akk-box wird mit eigenen Klassen kombiniert und überschreibt damit bei höherer Spezifität die Standardvorgaben

Beispiel:

```
.mz-akk-box.meine-klasse {  
    --mz-akk-max-width: 50rem;  
    --mz-akk-bg: #f0f4ff;  
}
```

ODER konkret

```
.mz-akk-box.akk-frame-flat {  
    ...  
}
```

2.4. Variablen im Überblick

Layout	
--mz-akk-max-width	maximale Breite
--mz-akk-gap	Abstand zwischen Elementen
Typografie	
--mz-akk-font	Schriftart
--mz-akk-font-size	Schriftgröße
mz-akk-text	Textfarbe
Theme	
--mz-akk-bg	Hintergrund
--mz-akk-header-bg	Titelhintergrund
--mz-akk-header-bg-hover	Hover-Effekt
--mz-akk-border	Linienfarbe
--mz-akk-focus	Fokusfarbe
Frame	
--mz-akk-radius	Ecken
--mz-akk-shadow	Schatten
Header	
--mz-akk-header-padding-y --mz-akk-header-padding-x	Innenabstände Titel
Content	
--mz-akk-body-padding-*	Innenabstände Inhalt
--mz-akk-content-flow-space	Abstand zwischen Elementen

Animation	
--mz-akk-duration	Dauer
--mz-akk-ease	Verlauf
Icons	
--mz-akk-icon-url	Symbol
--mz-akk-icon-size	Größe
--mz-akk-title-img-h	Bildgröße im Titel
Sonstiges	
--akk-scroll-offset	Scroll-Offset

2.5. Hinweise für Erweiterungen

- Änderungen möglichst über Variablen vornehmen; deren Wertzuweisung ändern
- mehrere Klassen können kombiniert werden
- direkte CSS-Overrides nur bei Bedarf
- akk-custom.css bleibt updatefest

2.6. Technische Hinweise

- mehrere Akkordeons pro Seite möglich
- JavaScript wird automatisch eingebunden
- Animation erfolgt über CSS

TEIL 3 ÜBERSICHT ÜBER ENTHALTENE GESTALTUNGSVARIANTEN

=====

Folgendes Beispiel wird für die Styling-Varianten unten benutzt:

```
{Akkordeon||
--item--
[fett|Titel Element 1]
--content--
[fett|Inhalt Element 1]
Hier können Inhalte stehen: Textabsätze, Bilder, Tabellen, Plugins

--item--
Titel Element 2
--content--
Inhalt Element 2

--item--
Titel Element 3
--content--
Inhalt Element 3
}
```

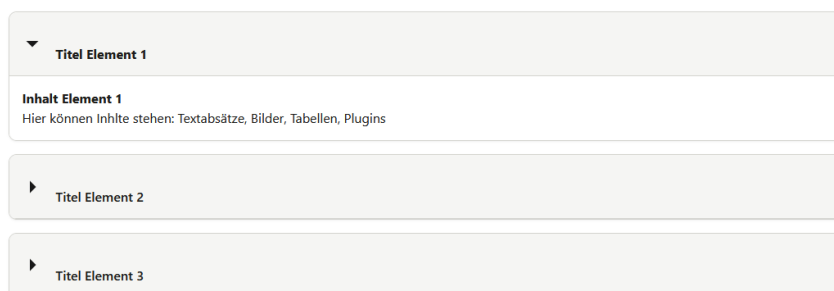
Bei gleichem Inhalt wird das Aussehen (Gestaltung) des Akkordeons nur über die erste Zeile gesteuert `{Akkordeon||... }`.

Zwischen `||` werden die definierten Schlüsselworte aus 1.7. eingesetzt und erzeugen den gewünschten Effekt.

Vor jedem ScreenShot werden die benutzten Schlüsselworte (Modifikatoren, CSS-Klassen) benannt!

3.1. OHNE zusätzliche Klassen; Standard

```
{Akkordeon||...}
```



Nachfolgende Beispiele **MIT** Modifikator-Klasse für Demonstration der prinzipiellen Gestaltung. Benutzt wird nur **ein** Parameter 1

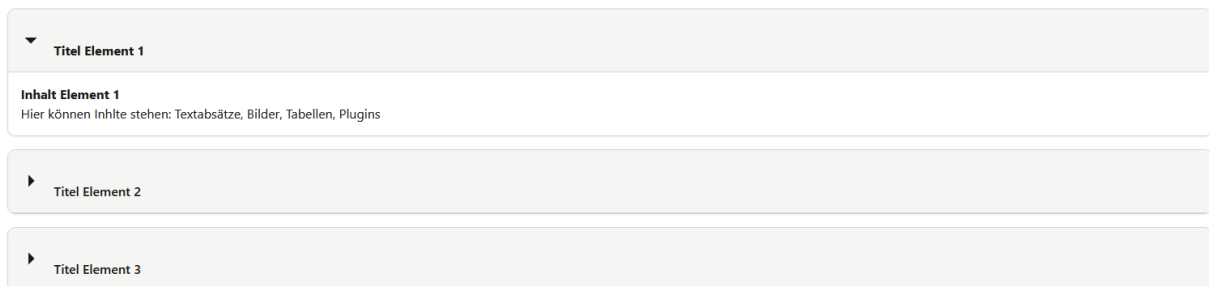
3.2. Breite geringer

```
{Akkordeon|akk-layout-small|...}
```



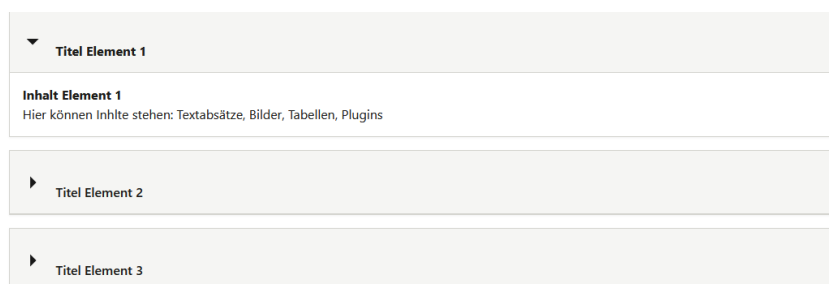
3.3. Volle Breite

```
{Akkordeon|akk-layout-full|...}
```



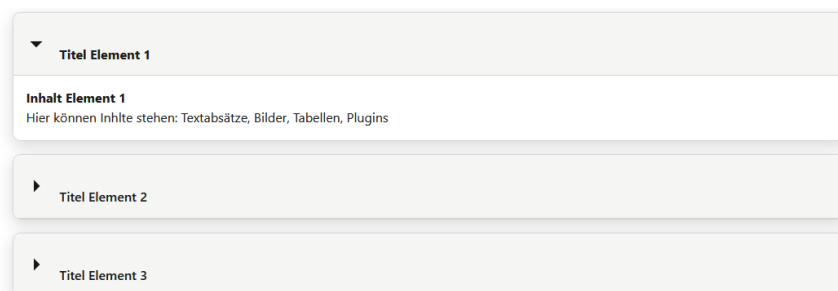
3.4. Ecken nicht gerundet

```
{Akkordeon|akk-frame-square|...}
```



3.5. mehr Schatten

```
{Akkordeon|akk-frame-elevated |...}
```



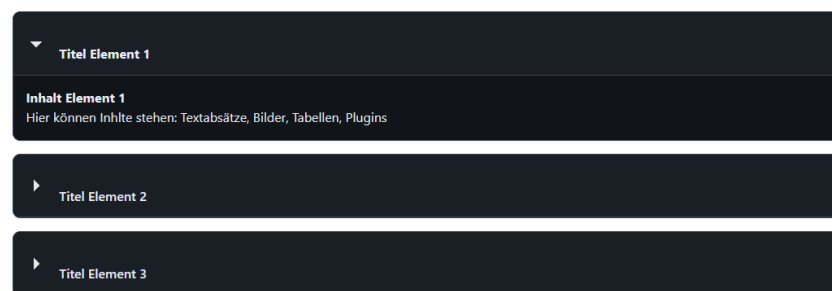
3.6. Farbschema Sand

```
{Akkordeon|akk-theme-sand|...}
```



3.7. Farbschema invertiert, dunkel

```
{Akkordeon|akk-theme-dark|...}
```



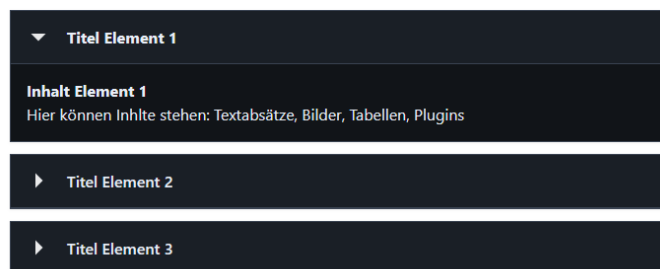
Hier Beispiele für **mehrere** Klassen in Kombination. Darauf achten, daß keine Widersprüche entstehen.

Empfehlung für Reihenfolge in Parameter 1:

layout frame theme content icon

3.8. `{Akkordeon|akk-layout-small akk-frame-square akk-theme-dark akk-icon-dreieck-weiss|...}`

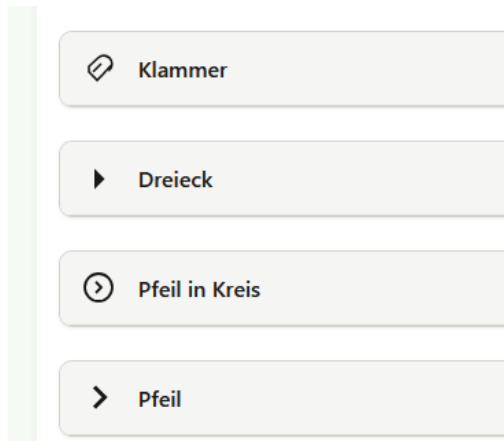
- Breite schmal
- Keine Abrundung der Ecken
- Farbschema Dunkel
- Symbol Dreieck, weiss



3.9. Symbole vor Titel

Vier Beispiele in der Reihenfolge von Oben nach Unten:

```
{Akkordeon|akk-icon-klammer|...}  
{Akkordeon|akk-icon-dreieck|...}  
{Akkordeon|akk-icon-pfeilkreis|...}  
{Akkordeon|akk-icon-pfeil|...}
```



3.10. Titel größer; Bild links; Text rechts; Inhalt mit FlexCards

```
{Akkordeon|akk-layout-tile|  
--item--  
[bildlinks|@=bild.jpg=@] Fütterungen ...  
--content--  
{FlexCards|Row|fc-content-img-top-sm|  
{FlexCards|Card|100|Vögel|[link=[bild|Vögel.jpg]|https://www....]|}  
{FlexCards|Card|100|Fuchs|[link=[bild|Vögel.jpg]|https://www....]|}  
Weitere Elemente  
}  
--item--  
--content--  
}
```

