

## Manual:Extension/Forms

The **Forms** extension allows formatting a wiki page or a user dialog as a form. It is also used in conjunction with the [Workflows](#) extension where it provides input forms for workflow data.

### Contents

1 Introduction .....	2
2 Create a checklist form .....	2
3 Create a checklist instance .....	5
4 Send email .....	5
5 Adjust the styling .....	6
5.1 Common.css .....	7
5.2 Inline styling .....	8
6 Related info .....	8

## Introduction

To create a checklist as a form, for example, the following steps are necessary:

1. **Create and save the checklist form** via the *Special:Form\_editor* special page. The form serves as a template for all checklists that are created with it.
2. **Create and save checklist instances** via the special page *Special:Create\_form\_instance*.

Optionally, the styling of the form can be adjusted.

As an example, we are creating a small checklist for an event where food-related hygiene information has to be tracked.

For each day of the event, a separate form has to be completed! The form has to be completed by 1 p.m.

Event

Event name

Fall fest 2022

Organizer

Hallo Welt!

Date

09/22/2022

Contact

John Doe

Hygiene requirements

Booth (floor, walls, ceiling)

✓

Handwash station (soap dispenser, disposable towels)

✗

Delivery processing

Refrigerated products

☒ Max. temperatur 7° C

☒ Packaging

☒ Expiration date

Baked goods

☒ Packaging

☐ Expiration date / freshness

Storage

Refrigerator 1

clean

Other comments

Event checklist example

## Create a checklist form

1. **Define** the main properties of the form:
  1. **Open** the *Special:Form\_editor* page.
  2. **Enter** a name for the new form, e.g. *EventChecklistHygiene*. This name is used as a unique identifier for the database. Therefore, do not use any special characters (apart from the hyphen "-").



2. **Click** the text box radio button.
  3. **Enter** a name. This is used as the database ID for this field and should therefore not contain any special characters.
  4. **Enter** a label. This is displayed as the label for the text field.
  5. **Add** more form elements.
4. Define which form fields should be used to generate the page name of the form instances created with this form. As an example, we define the page name for the checklists as subpages to the form name with the values from the event name and date fields (both fields must exist in the form: `EventChecklistHygiene/{{ech-eventname}}-{{ech-date}}`)
1. **Mark** *event name* and *day of event* as required so that these values always exist.
  2. **Switch** to the *Target* view in the form properties.
  3. **Select** "Json in a wiki page" as the target type.
  4. **Enter** your title syntax: `EventChecklistHygiene/{{ech-eventname}}-{{ech-date}}`

Form properties

Appearance	Target Type	JSON on wiki page
Behaviour	Title	EventChecklistHygiene/{{ech-eventname}}-{{ech-date}} *
Infrastructure	Assign after action	<input type="checkbox"/>
Target		
Listeners		

Define the page name syntax for form instances

5. **Click** Submit at the bottom of the page to save the form. The form is now stored in the main namespace as *EventChecklistHygiene.form*.

EventChecklistHygiene

1

Event Checklist

Event

Event name

Submit Reset

Saved form

To continue editing the form, switch to the editing mode of the page (1).

## Create a checklist instance

---

To use the form, create your first checklist.

1. **Open** the page *Special:Create\_form\_instance*.
2. **Choose** the form *EventChecklistHygiene*.
3. **Fill out** the form fields.
4. **Click** *Submit*.

Special Create form instance

### Create form instance

Select the form to create

Event Checklist

Event

Event name \*

Day of event \*

EventChecklistHygiene

Eigenkontrolle-veranstaltung

EventChecklistHygiene

Submit Reset

Create a checklist instance

The checklist is now saved in the wiki and can be edited further.

## Send email

---

The form can also send an Email after being saved:

The screenshot shows the 'Target' configuration page in the BlueSpice Forms extension. On the left is a sidebar with icons and labels for 'Appearance', 'Behaviour', 'Infrastructure', 'Target' (selected), and 'Listeners'. The main area is divided into two columns. The left column contains labels for 'Target Type', 'Receiver', 'Subject', 'Content', and 'Assign after action'. The right column contains the corresponding input fields: a dropdown for 'Target Type' (set to 'Email'), a dropdown for 'Receiver' (set to 'Marketing'), a text area for 'Subject' containing 'Dear {{{name}}},', a larger text area for 'Content' containing 'This is the content: {{{content}}}', and a checkbox for 'Assign after action' which is currently unchecked. Below this main configuration area is a section titled 'Form elements' with a sub-section 'Element picker' containing two input fields labeled 'Name' and 'Content'.

- **Target type:** Email
- **Recipient:** Possible recipients must have been configured on the server for security reasons. Example:

```
$wgFormsTargetEmailRecipients = [  
'Marketing' => "marketing@example.com",  
'Administration' => 'WikiSysop'  
];
```

- **Subject:** Existing form fields can be used as variables here. The value is taken from the form.
- **Content:** As in the subject, existing form fields can also be used as variables here. The value is taken from the form.

[Localization messages](#) in combination with form fields can also be used as subject or email body:

```
{{int:Mailsubject| {{{name}}}|{{{surname}}} }}
```

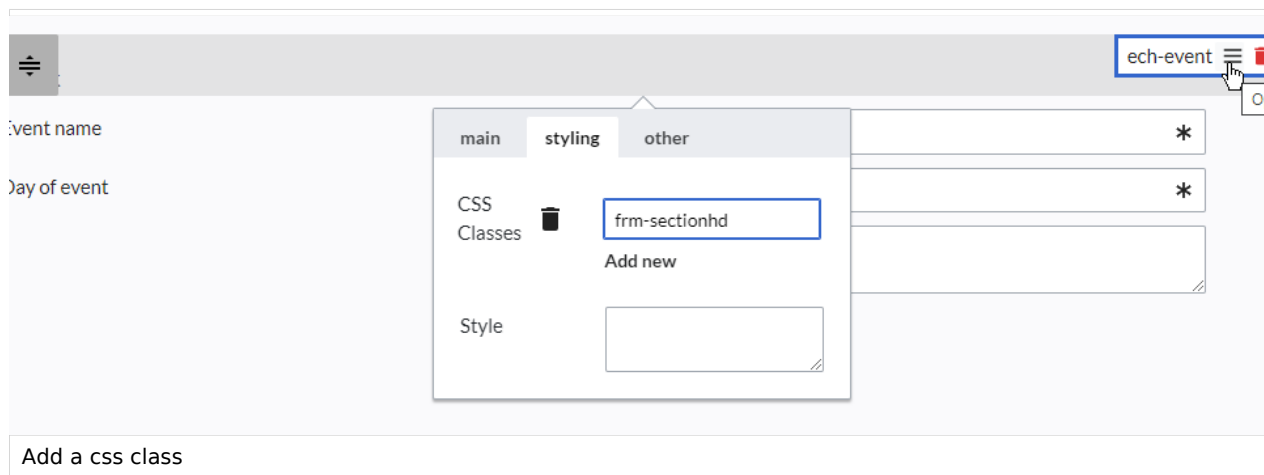
## Adjust the styling

## Common.css

To style individual elements in the form, go to the styling tab in the options dialog. There, you can enter a css class name. The styling is then defined on the *MediaWiki:Common.css* page (admin rights are required).

To style the section heading:

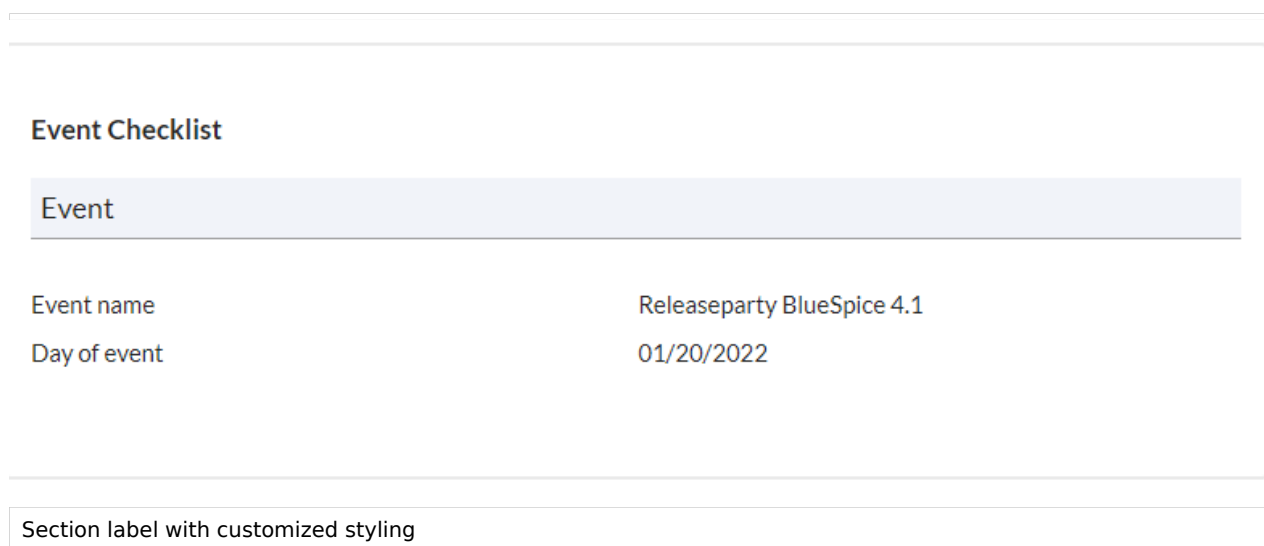
1. **Open** the styling tab in the options menu of the Event heading field.
2. **Add** a new CSS class and enter a selector name (*CSS Classes* field), e.g. *frm-sectionhd*.



The screenshot shows the 'ech-event' options dialog for the 'Event name' field. The 'styling' tab is active, displaying the 'CSS Classes' field with the value 'frm-sectionhd' and an 'Add new' button. The 'Style' field is empty. The dialog is titled 'ech-event' and has a close button.

3. **Paste** the following style declarations into the *MediaWiki:Common.css* page, e.g.:

```
border-bottom: 1px solid #a6a6a7; background:#f1f3f9; padding:6px;margin:30px 0;
```



The screenshot shows the 'Event Checklist' section of the form. It contains a table with two columns: 'Event name' and 'Day of event'. The 'Event name' row is highlighted. Below the table, there is a section label 'Section label with customized styling'.

Event name	Day of event
Releaseparty BlueSpice 4.1	01/20/2022

Section label with customized styling

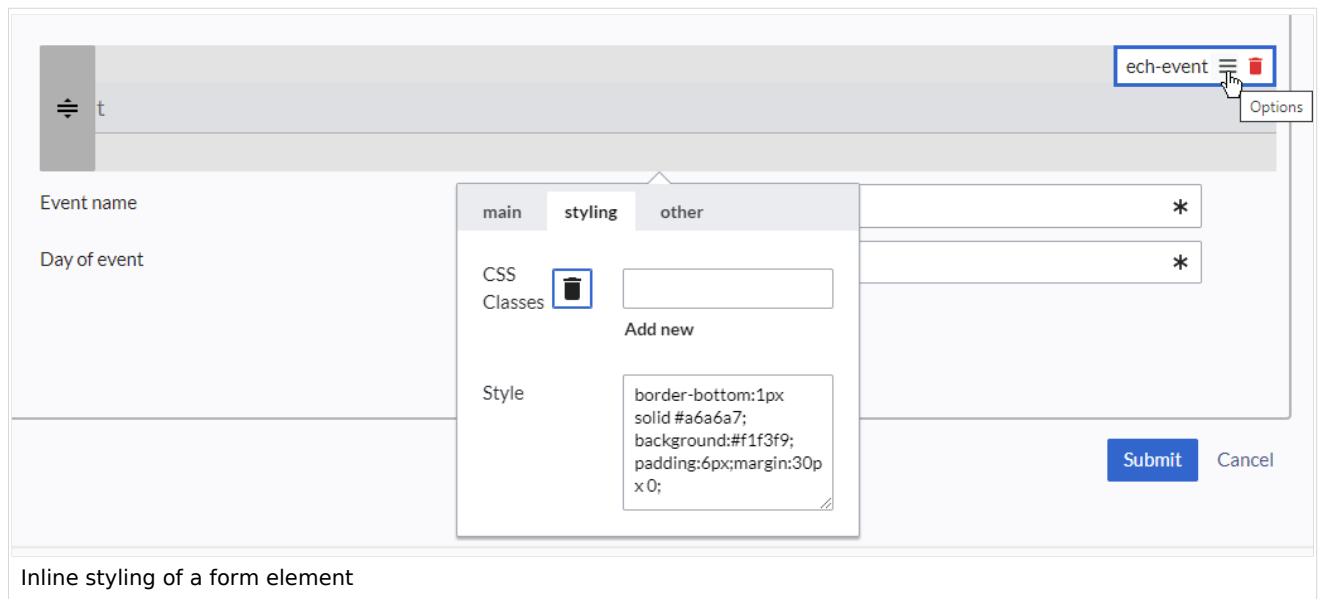
If you do not have permission to view this page, you can enter style information directly for each element in the form via inline styling.

## Inline styling

Without access to *MediaWiki:Common.css*, the style information can be entered directly in the form. However, with many recurring style declarations this is not efficient in contrast to the definition of CSS classes in *MediaWiki:Common.css*.

As an example, let's apply the previous section heading example directly as inline styling. To do this, open the options menu of the *ech-event* form element again. Enter the following statement in the *Styling* field and save the change:

```
border-bottom: 1px solid #a6a6a7; background:#f1f3f9; padding:6px;margin:30px 0;
```



The screenshot shows a form element titled 'ech-event' with an options menu open. The menu has three tabs: 'main', 'styling', and 'other'. The 'styling' tab is active, showing a 'CSS Classes' section with a trash icon and an 'Add new' button, and a 'Style' section with a text area containing the CSS code: `border-bottom:1px solid #a6a6a7; background:#f1f3f9; padding:6px;margin:30px 0;`. The form itself has two input fields: 'Event name' and 'Day of event'. A 'Submit' button and a 'Cancel' link are at the bottom right of the form.

Inline styling of a form element

## Related info

- [ <https://www.mediawiki.org/wiki/HTMLForm> Full documentation on mediawiki.org]

➔ [Technical Reference: Forms](#)