

Syntax for discussion category forms

In this article

About YAML syntax for discussion category forms

Top-level syntax

You can use YAML syntax to define the fields in your discussion category forms.

About YAML syntax for discussion category forms

You can create custom discussion category forms by adding a YAML form definition file to the `/.github/DISCUSSION_TEMPLATE/` folder in your repository. If you're new to YAML and want to learn more, see "[Learn YAML in Y minutes](#)."

The name must correspond with the slug for one of your discussion categories. For example, the template for the "Announcements" category should be

```
.github/DISCUSSION_TEMPLATE/announcements.yml
```

For each field, you can define the input type, validation, and a default label.

When a community member fills out a discussion form, their responses for each input are converted to markdown and added to the body of a discussion. Community members can edit their discussions that were created with a discussion form and other people can interact with the discussion like a discussion created through other methods.

This example YAML configuration file defines a general discussion category form.

YAML



```
title: "[General] "
labels: ["General Introduction"]
body:
  - type: markdown
    attributes:
      value: |
        This is text that will show up in the template!
  - type: textarea
    id: improvements
    attributes:
      label: Top 3 improvements
      description: "What are the top 3 improvements we could make to this
project?"
    value: |
      1.
      2.
      3.
      ...
    render: bash
  validations:
    required: true
  - type: markdown
    attributes:
      value: |
```

```

    ## Markdown header
    And some more markdown
- type: input
  id: has-id
  attributes:
    label: Suggestions
    description: A description about suggestions to help you
  validations:
    required: true
- type: dropdown
  id: download
  attributes:
    label: Which area of this project could be most improved?
    options:
      - Documentation
      - Pull request review time
      - Bug fix time
      - Release cadence
  validations:
    required: true
- type: checkboxes
  attributes:
    label: Check that box!
    options:
      - label: This one!
        required: true
      - label: I won't stop you if you check this one, too
- type: markdown
  attributes:
    value: |
      ### The thrilling conclusion
      _to our template_

```

Top-level syntax [↗](#)

The configuration file for a discussion category form must contain a `body` key, and the `body` must contain at least 1 non-Markdown field.

YAML



```

body:
- type: input
  id: suggestion
  attributes:
    label: Suggestion
    description: "How might we make this project better?"
    placeholder: "Adding a CODE_OF_CONDUCT.md file would be a great idea."
  validations:
    required: true

```

You can set the following top-level keys for each issue form.

Key	Description	Required	Type
<code>body</code>	Definition of the input types in the discussion form.	Required	Array
<code>labels</code>	Labels that will automatically be added to discussions created with this template.	Optional	Array or comma-delimited string

title

A default title that will
be pre-populated in
the discussion
submission form.

Optional

String

To add fields to your form, include an array of form elements in the `body` key. For a list of available elements and their syntaxes, see "[Syntax for GitHub's form schema](#)."

Legal

© 2023 GitHub, Inc. [Terms](#) [Privacy](#) [Status](#) [Pricing](#) [Expert services](#) [Blog](#)