

--	--	--	--

[illegible]

□ □ □ □

Markdown MDX

□ □ □ □

--	--	--	--

- □□□□□□□□□□□□□□□□□□
- □□□□□□□□
- □□□□□

□□ **Markdown** □□□□ **MDX**

MDX



0000000000 MDX 000

0000

0000000000 Doom 00

00

00

0000

000000

00

MDX

00 MDX 000000000000000000

rspress 00

doom 00

00000000

000

000000000000000000

i18n.json

.ts/.tsx

.mdx

API 00

00 API

CRD

0000

00 openapi 00

000000

props

00

0000

000000

00

000000000000000000000000 ACP 00

000000

0000

000 ACP

📄📄

📄📄

📄📄📄

📄📄📄📄

📄📄📄📄📄

📄📄📄📄📄

📄📄📄

📄📄📄📄📄📄

📄📄📄

📄📄 PDF

📄📄📄📄

📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄

```
mkdir my-docs && cd my-docs
```

📄 `npm init -y` 📄📄📄📄📄📄📄📄📄📄📄 `npm`📄`yarn` 📄 `pnpm` 📄 `doom:`

 `npm`  `yarn`  `pnpm`  `bun`

```
npm install -D @alauda/doom typescript
```

📄📄📄📄📄📄📄📄:

```
# 📄📄 docs 📄📄📄📄📄📄📄  
mkdir docs/en && echo '# Hello World' > docs/en/index.md
```

```
mkdir docs/zh && echo '# 文档' > docs/zh/index.md
```

package.json 配置:

```
{
  "scripts": {
    "dev": "doom dev",
    "build": "doom build",
    "new": "doom new",
    "serve": "doom serve",
    "translate": "doom translate",
    "export": "doom export"
  }
}
```

doom.config.yml:

```
title: My Docs
```

tsconfig.json 配置:

```
{
  "compilerOptions": {
    "jsx": "react-jsx",
    "module": "NodeNext",
    "moduleResolution": "NodeNext",
    "noUnusedLocals": true,
    "noUnusedParameters": true,
    "resolveJsonModule": true,
    "skipLibCheck": true,
    "strict": true,
    "target": "ESNext",
  },
  "mdx": {
    "checkMdx": true,
  },
}
```

global.d.ts

```
/// <reference types="@alauda/doom/runtime" />
```

.mdx doom

```
doom -h
```

```
# output
```

```
Usage: doom [options] [command]
```

```
Doctor Doom making docs.
```

```
Options:
```

```
-V, --version
```

output the version number

```
-c, --config <config>
```

Specify the path to the config file

```
-v <version>
```

Specify the version of the documentation, c

```
-b, --base <base>
```

Override the base of the documentation

```
-p, --prefix <prefix>
```

Specify the prefix of the documentation bas

```
-f, --force [boolean]
```

Force to

1. fetch latest reference remotes or scaffo

2. translate ignore hash equality check and

```
-i, --ignore [boolean]
```

Ignore internal routes (default: false)

```
-d, --download [boolean]
```

Display download pdf link on nav bar (defau

```
-e, --export [boolean]
```

Run or build in exporting PDF mode, `apis/*

```
-I, --include <language...>
```

Include **only** the specific language(s),

```
-E, --exclude <language...>
```

Include all languages except the specific l

```
-o, --out-dir <path>
```

Override the `outDir` defined in the config

```
-r, --redirect <enum>
```

Whether to redirect to the locale closest t

```
-n, --no-open [boolean]
```

Do not open the browser after starting the

```
-h, --help
```

display help for command

```
Commands:
```

```
dev [options] [root]
```

Start the development server

```
build [root]
```

Build the documentation

```
preview|serve [options] [root]
```

Preview the built documentation

<code>new</code> [template]	Generate scaffolding from templates
<code>translate</code> [options] [root]	Translate the documentation
<code>export</code> [options] [root]	Export the documentation as PDF, <code>`apis/**`</code>
<code>help</code> [command]	display help <code>for</code> command

📄📄📄📄📄📄

📄 `yarn dev` 📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄

```
doom dev -h

# output
Usage: doom dev [options] [root]

Start the development server

Arguments:
  root                Root directory of the documentation

Options:
  -H, --host [host]   Dev server host name
  -P, --port [port]   Dev server port number
  -l, --lazy [boolean] Whether to enable `lazyCompilation` which could i
  -h, --help          display help for command
```

📄📄📄📄📄📄

📄 `yarn build` 📄📄📄📄📄📄📄📄📄📄 `dist` 📄📄📄📄📄📄

📄📄📄📄

📄 `yarn serve` 📄📄📄📄📄📄📄📄📄📄📄 `-b` , `-p` 📄📄📄📄📄📄📄📄 `-b` , `-p` 📄
📄

📄📄📄📄📄📄📄

📄 `yarn new` 📄📄📄📄📄📄📄📄📄📄📄📄📄

□□□□

`doom translate -h`

output

`Usage: doom translate [options] [root]``Translate the documentation``Arguments:``root Root directory of the documentation``Options:`

`-s, --source <language>` Document source language, one of en, zh, ru (default)

`-t, --target <language>` Document target language, one of en, zh, ru (default)

`-g, --glob <path...>` Glob patterns for source dirs/files

`-C, --copy [boolean]` Whether to copy relative assets to the target directory

`-h, --help` display help for command

- `-g, --glob` 指定源文件的路径模式 `glob` 模式，支持通配符

1. `yarn translate -g abc xyz` 将 `<root>/<source>/abc` 目录下的文件复制到 `<root>/<target>/xyz` 目录

2. `yarn translate -g '*'` 将 `<root>/<source>` 目录下的所有文件复制到 `<root>/<target>` 目录

- `-C, --copy` 是否将源文件复制到目标目录，默认为 `false`

- 示例

1. `<root>/public/<source>/abc.jpg` 复制到 `<root>/public/<target>/abc.jpg`

2. `<root>/<source>/assets/xyz.jpg` 复制到 `<root>/<target>/assets/xyz.jpg`

3. `<root>/<source>/assets/<source>/xyz.jpg` 复制到 `<root>/<source>/assets/<source>/xyz.jpg`

```
<root>/<target>/assets/<target>/xyz.jpg
./assets/<target>/xyz.jpg
```

- `source` 和 `target` 的相对路径
- 1. `/<source>/abc.jpg` 和 `<root>/public/<target>/abc.jpg` 是相同的
因此 `/<target>/abc.jpg` 是 `<root>/public/<target>/abc.jpg` 的相对路径
- 2. `<root>/<source>/abc.mdx` 和 `./assets/<source>/xyz.jpg` 是相同的
因此 `<root>/<target>/assets/<target>/xyz.jpg` 是 `<root>/<source>/abc.mdx` 的相对路径
因此 `./assets/<target>/xyz.jpg` 是 `<root>/<source>/assets/<target>/xyz.jpg` 的相对路径

WARNING

在 `source` 和 `target` 中，`internalRoutes` 是 `target` 的相对路径

TIP

在 `source` 中，`AZURE_OPENAI_API_KEY` 是 `Leader` 的相对路径

在 `source` 中，`internalRoutes` 是 `target` 的相对路径

```
i18n:
  title:
    en: DevOps Connectors
  additionalPrompts: 'Azure Connectors'
  disableAutoTranslation: false
title: DevOps
```

PDF

WARNING

在 `source` 中，`yarn build` 是 `target` 的相对路径

```
doom export -h
```

```
# output
```

```
Usage: doom export [options] [root]
```

```
Export the documentation as PDF, `apis/**` and `*/apis/**` routes will be ign
```

```
Arguments:
```

```
  root                Root directory of the documentation
```

```
Options:
```

```
-H, --host [host]  Serve host name
-P, --port [port]  Serve port number (default: "4173")
-h, --help         display help for command
```

```
00 yarn export 00000 PDF 000000000000 -b , -p 0000000000000000 -b , -p
00
```

```
000000 playwright 00000000 build-harbor.alauda.cn/frontend/playwright-
runner:doom 000000000000000000 00000000000000000000
```

```
.env.yarn
```

```
PLAYWRIGHT_DOWNLOAD_HOST="https://cdn.npmmirror.com/binaries/playwright"
```



📄

📄 `doom` 📄📄

📄📄

📄📄

API 📄📄

📄📄📄📄📄

📄📄📄

📄📄📄

📄📄📄

📄📄📄

📄📄📄📄📄

`sites.yaml` 📄

📄📄

📄

📄“📄📄📄📄”📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄

📄📄

📄📄

📄

📄

Markdown

Callouts

Mermaid

MDX

📄 MDX 📄📄📄📄📄📄📄📄📄📄📄

rspress 📄

doom 📄

📄📄📄📄📄

📄📄📄

📄📄📄📄📄📄📄📄📄📄📄

i18n.json

.ts/.tsx

.mdx

API 📄

📄 API

CRD

📄📄📄

📄 openapi 📄

📄📄📄📄📄

props

📄

📄📄📄

📄📄📄📄📄

00

00000000000000000000 ACP 00

00000

0000

000 ACP

- `lang` 指定生成文档的语言 `en` 指定生成文档的语言
`public` 指定生成文档是否公开
- `title` 指定生成文档的标题
- `logo` 指定生成文档的 logo `public` 指定生成文档是否公开
`doom` 指定生成文档的 logo 为 doom
- `logoText` 指定生成文档的 logo 文本
- `icon` 指定生成文档的 icon `logo`
- `base` 指定生成文档的 base 路径 `product-docs` 指定生成文档的 base 路径 `/`
- `outDir` 指定生成文档的输出目录 `dist/{base}/{version}` 指定生成文档的输出目录
`dist/{outDir}/{version}` 指定生成文档的输出目录 `version` 指定生成文档的版本

```
api:
  # CRD  doom.config.*  glob  json/yaml
  crds:
    - docs/shared/crds/*.yaml
  # OpenAPI  doom.config.*  glob  json/yaml
  openapis:
    - docs/shared/openapis/*.json
  #  openapi
  #  https://doom.alauda.cn/apis/references/CodeQuality.html#v1alpha1.CodeQ
  references:
    v1alpha1.CodeQualityBranch: /apis/references/CodeQualityBranch#v1alpha1.C
  #  API  gateway
  pathPrefix: /apis
```

API

```
# doom.config.* glob json/yaml
permission:
  functionresources:
    # `kubectl get functionresources`
    - docs/shared/functionresources/*.yaml
  roletemplates:
    # `kubectl get roletemplates -l auth.cpaas.io/roletemplate.official=true`
    - docs/shared/roletemplates/*.yaml
```

```
reference:
- repo: alauda-public/product-doc-guide #
  branch: # [string]
  publicBase: # [string] /images/xx.png do
sources:
- name: anchor #
  path: docs/index.mdx# # doom.conf
  ignoreHeading: # [boolean] true
  processors: #
    - type: ejsTemplate
      data: # ejs `<%= data.xx %>`
  frontmatterMode: merge # frontmatter ignore ign
```

frontmatterMode

- ignore frontmatter frontmatter

- `merge` 🐼🐼🐼🐼🐼 `frontmatter`🐼🐼🐼🐼🐼 `key`🐼🐼🐼🐼🐼🐼🐼🐼🐼🐼🐼
- `replace` 🐼🐼🐼🐼🐼🐼 `frontmatter` 🐼🐼🐼🐼🐼 `frontmatter`
- `remove` 🐼🐼🐼🐼🐼🐼 `frontmatter`

🐼🐼🐼🐼🐼🐼🐼🐼

🐼🐼🐼🐼🐼🐼

```
releaseNotes:
  queryTemplates:
    fixed: # 🐼🐼🐼 ejss 🐼🐼🐼 jql 🐼🐼
    unfixed:
```

release-notes.md

```
<!-- release-notes-for-bugs?template=fixed&project=DevOps -->
```

release-notes.mdx

```
{/* release-notes-for-bugs?template=fixed&project=DevOps */}
```

🐼🐼 `template=fixed&project=DevOps` 🐼🐼 `fixed` 🐼 `queryTemplates` 🐼🐼🐼🐼🐼🐼🐼
🐼🐼 `query` 🐼🐼 `project=DevOps` 🐼🐼 `ejs` 🐼🐼🐼🐼🐼 `fixed` 🐼🐼🐼🐼🐼 `jira` `jql` 🐼🐼
`https://jira.alauda.cn/rest/api/2/search?jql=<jql>` 🐼🐼🐼 `API` 🐼🐼🐼🐼🐼🐼
`JIRA_USERNAME` 🐼 `JIRA_PASSWORD` 🐼🐼🐼🐼🐼🐼🐼

🐼🐼🐼🐼🐼

sidebar:

```
collapsed: false #  false
```

--	--	--	--	--	--

```
internalRoutes: # glob docs cli -i, --ignore`
- '*/internal/**/*'
```

--	--	--	--	--	--	--	--

```
shiki:
  theme: # optional, https://shiki.style/themes
  langs: # optional, https://shiki.style/languages
  transformers: # optional, only available in js/ts config, https://shiki.sty
```

WARNING

plaintext

`sites.yaml` ☐ ☐

[illegible]

```
- name: connectors #   
base: /devops-connectors #   
version: v1.1 # ExternalSite/ExternalSiteLink   
  
displayName: # name
```

en: DevOps Connectors

zh: DevOps 连接器

连接器

连接器

repo: <https://github.com/AlaudaDevops/connectors-operator> # 连接器

image: devops/connectors-docs # 连接器

🌑🌑🌑🌑

translate:

```
# 连接器 `sourceLang`, `targetLang`, `userPrompt` 或 `additionalPrompts`
# 或 `sourceLang` 或 `targetLang` 或 `sourceLang` 或 `targetLang`
# `userPrompt` 连接器
# `additionalPrompts` 连接器 `frontmatter.i18n` 连接器 `additionalPrompts` 连接器
# 连接器
```

systemPrompt: |

连接器

连接器<%= targetLang %>连接器<%= sourceLang %>连接器<%=

连接器

```
- 连接器<%= sourceLang %>连接器<%= targetLang %>连接器
- 连接器 MDX 连接器 MDX 连接器 jsx 连接器 <Overview />连接器
- 连接器
- MDX 连接器MDX 连接器 MDX 连接器
  - <Overview /> 连接器 Overview 连接器
  - <Tab label="value">连接器</Tab>连接器 label 连接器 key 连接器"value" 连接器
<%= terms %>
- 连接器
  - {/* release-notes-for-bugs */}
  - <!-- release-notes-for-bugs -->
- 连接器
  - {/* reference-start */}
  - {/* reference-end */}
  - <!-- reference-start -->
  - <!-- reference-end -->
- 连接器 \< 连接器 \{ 连接器
- 连接器 Markdown 连接器 frontmatter, 连接器 frontmatter.i18n 连接器
```

```
##  🌑

🌑🌑🌑🌑🌑🌑🌑🌑

1.  🌑🌑<%= sourceLang %>🌑🌑🌑🌑🌑<%= targetLang %>🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑
2.  🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑
    - 🌑🌑🌑<%= targetLang %>🌑🌑🌑🌑🌑🌑🌑🌑
    - 🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑
    - 🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑
3.  🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑<%= targetLang %>🌑🌑🌑🌑
4.  🌑🌑🌑🌑🌑🌑🌑<%= targetLang %>🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑<%= targetLang %>🌑🌑🌑🌑🌑🌑🌑

🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑🌑

<%= userPrompt %>

<%= additionalPrompts %>
  userPrompt: # 🌑🌑🌑🌑🌑🌑 `systemPrompt` 🌑🌑 `ejs` 🌑🌑🌑🌑
```

11

`index`

`index.md` `index.mdx`

```
├─ index.md
├─ start.mdx
└─ usage
    ├─ index.mdx
    └─ convention.md
```

```

□□□□□□
1. public □□□□□□□□□□□□□□□□
2. public/_remotes □□□□□□□□□□□□□□□□□□□□□□□□
   */public/_remotes □□ .gitignore □□□□□□□□
3. shared □□□□□□□□□□□□□□□□□□□□□□□□□□

```

1. `public` □□□□□□□□□□□□□□□□
2. `public/_remotes` □□□□□□□□□□□□□□□□□□□□□□□□□□□□
`*/public/_remotes` □□ `.gitignore` □□□□□□□□□□
3. `shared` □□□□□□□□□□□□□□□□□□□□□□□□□□□□

🐱🐱

🐱🐱🐱🐱🐱🐱 `frontmatter` 🐱🐱🐱🐱🐱🐱🐱🐱🐱🐱🐱🐱🐱🐱

```
---
title: 🐱
description: 🐱
author: 🐱
category: 🐱
---
```

🐱🐱🐱🐱🐱🐱 `MDX` 🐱 `.mdx` 🐱🐱🐱🐱🐱 `frontmatter` 🐱🐱🐱🐱🐱🐱

🐱🐱

🐱 `index.md` 🐱 `index.mdx` 🐱🐱🐱🐱🐱🐱🐱🐱🐱🐱🐱🐱🐱🐱 `frontmatter` 🐱
`weight` 🐱🐱🐱🐱🐱🐱🐱🐱🐱🐱 `weight` 🐱🐱🐱🐱🐱🐱

```
---
weight: 1
---
```

WARNING

🐱🐱🐱🐱🐱🐱🐱🐱🐱🐱🐱🐱🐱🐱🐱🐱🐱🐱🐱🐱

🐱🐱

index.mdx Overview

```
#  
  
<Overview />
```

Markdown

関数 [gfm](#) 関数Doom 関数関数関数 Markdown 関数関数

関数

Callouts

Mermaid

Callouts

関数関数関数

NOTE

- 関数関数関数関数関数関数関数 ; , % , # , // , /** */ , -- 関数 <!-- --> 関数
- 関数関数関数関数関数関数関数 [\ ! code callout] 関数関数
- 関数 関数 ::callouts 関数関数関数関数関数関数関数関数 <div class="doom-callouts"> 関数
<Callouts> 関数関数

```
```sh
Memory overhead per virtual machine ≈ (1.002 × requested memory) \
 + 218 MiB \ ①
 + 8 MiB × (number of vCPUs) \ ②
 + 16 MiB × (number of graphics devices) \ ③
 + (additional memory overhead) ④
...

:::callouts
```

- 1. Required for the processes that run in the ``virt-launcher`` pod.
- 2. Number of virtual CPUs requested by the virtual machine.
- 3. Number of virtual graphics cards requested by the virtual machine.
- 4. Additional memory overhead:
  - If your environment includes a Single Root I/O Virtualization (SR-IOV) n
  - If Secure Encrypted Virtualization (SEV) is enabled, add 256 MiB.
  - If Trusted Platform Module (TPM) is enabled, add 53 MiB.

...

Memory overhead per virtual machine  $\approx (1.002 \times \text{requested memory}) \setminus$

+ 218 MiB  $\setminus$  ①

+ 8 MiB  $\times$  (number of vCPUs)  $\setminus$  ②

+ 16 MiB  $\times$  (number of graphics devices)  $\setminus$  ③

+ (additional memory overhead) ④

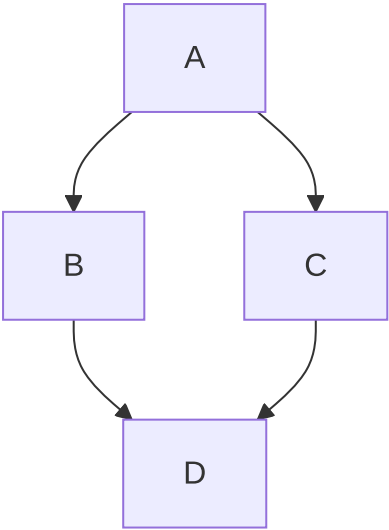
- ① Required for the processes that run in the `virt-launcher` pod.
- ② Number of virtual CPUs requested by the virtual machine.
- ③ Number of virtual graphics cards requested by the virtual machine.
- ④ Additional memory overhead:
  - If your environment includes a Single Root I/O Virtualization (SR-IOV) network device or a Graphics Processing Unit (GPU), allocate 1 GiB additional memory overhead for each device.
  - If Secure Encrypted Virtualization (SEV) is enabled, add 256 MiB.
  - If Trusted Platform Module (TPM) is enabled, add 53 MiB.

Shiki Transformers

# Mermaid

```
mermaid
graph TD;
 A-->B;
 A-->C;
 B-->D;
 C-->D;

```



# MDX

MDX Markdown Markdown JSX [rspress MDX](#)

rspress

doom

Overview

Directive

ExternalSite

ExternalSiteLink

AcpApisOverview ExternalApisOverview

Term

props

TermsTable

props

JsonViewer

## rspress

rspress .mdx

- Badge
- Card
- LinkCard

- [PackageManagerTabs](#)
- [Steps](#)
- [Tab/Tabs](#)
- [Toc](#)

在 `preview.mdx` 文件中，从 `@rspress/core/theme` 包中引入 `SourceCode` 组件。

```
preview.mdx

import { SourceCode } from '@rspress/core/theme'

<SourceCode href="/" />
```

# doom 文档

`doom` 是一个用于生成文档的工具，它支持 MDX 格式，并提供了丰富的组件和 API 来帮助你快速搭建文档网站。

## Overview

本文将介绍 `doom` 的基本用法和核心概念。

## Directive

在 MDX 文件中，你可以使用 `Directive` 组件来定义文档的元数据，如标题、描述等。

```
- 在 `doc/en` 目录下创建 `doc/zh` 目录，用于存放中文文档。

<Directive type="danger" title="警告">
 这是一个警告信息，请仔细阅读。
</Directive>
```

- ( doc/en ) □□□□□□□□ doc/zh □□□□□□□□□□□□□□□□□□□□□□  
□□□□

11

doc/zh

11

## ExternalSite

--	--	--	--	--	--	--	--

```
<ExternalSite name="connectors" />
```

### Note


 DevOps
 


 DevOps

DevOps ☐ ☐ ☐ ☒

## ExternalSiteLink

□ □ □ □ □ □ □ □ □ □

```
<ExternalSiteLink name="connectors" href="link.mdx#hash" children="Content" /
```

[Content ↗](#)

**TIP**

```
mdx <ExternalSiteLink name="connectors" href="link" children="Content" />
```

```
<ExternalSiteLink name="connectors" href="link">
 Content {/* □□□□ `p` □□ */}
```

```
</ExternalSiteLink>
```

このコンポーネントは `p` という子コンポーネント `children` を受け取る。

## AcpApisOverview ☐ ExternalApisOverview

このコンポーネントは API の一覧を表示する。

```
<AcpApisOverview />
{/* same as following */}
<ExternalApisOverview name="acp" />

<ExternalApisOverview name="connectors" />
```

### Note

この ACP APIs の一覧は [ACP APIs](#) を参照してください。

### Note

この DevOps の API の一覧は [DevOps の API](#) を参照してください。

## Term

このコンポーネントは Term の一覧を表示する。

```
<Term name="company" textCase="capitalize" />
<Term name="product" textCase="lower" />
<Term name="productShort" textCase="upper" />
```

このコンポーネントは ACP

### props

- `name` : 表示する Term の名前
- `textCase` : 表示する Term の文字列のケース。 `lower` , `upper` , `capitalize`



TermsTable

```
<TermsTable />
```

Key	Value	Label	Category	Icon	Color
company	Alauda	-	Alauda	-	Blue
product	Alauda Container Platform	-	Alauda Container Platform	-	Blue
productShort	ACP	-	ACP	-	Blue

props

- terms : NormalizedTermItem[]

JsonViewer

```
<JsonViewer value={{ key: 'value' }} />
```

```
yaml json
key: value
```

shared

```
import CommonContent from './shared/CommonContent.mdx'
```

```
<CommonContent />
```

runtime API .jsx/.tsx .mdx

```
// shared/CommonContent.tsx
```

```
export const CommonContent = () => {
 const { page } = usePageData()
 return <div>{page.title}</div>
}
```

```
// showcase/content.mdx
```

```
import { CommonContent } from './shared/CommonContent'
<CommonContent />
```

## WARNING

.mdx props issue props  
 .jsx/.tsx

--	--	--

alauda en / zh  
public en / zh

11

i18n.json

```
.ts/.tsx
```

.mdx

## i18n.json

```

// docs
// i18n.json
// useI18n

```

docs/i18n.json

```
{
 "title": {
 "zh": "标题",
 "en": "Title"
 },
 "description": {
 "zh": "描述",
 "en": "description"
 }
}
```

## .ts/.tsx

```
import { useI18n } from '@rspress/runtime'

export const CommonContent = () => {
 const t = useI18n()
 return <h1>{t('title')}</h1>
}
```

## .mdx

```
import { useI18n } from '@rspress/runtime'

{useI18n()('title')}

{useI18n()('description')}
```

# API 文档

本页面提供 API 文档 API 文档 CRD (Custom Resource Definition) 文档文档文档文档文档文档文档文档

```
├─ apis
│ └─ advanced-apis # 高级 API
│ └─ crds # CRDs
│ └─ references # 参考
```

## 文档

- API 文档
  - props
- CRD
  - props
- 文档文档
  - props
- openapi 文档

## API 文档

```
advanced-apis/codeQualityTaskSummary.mdx

CodeQualityTaskSummary

<OpenAPIPath path="/plugins/v1alpha1/template/codeQuality/task/{task-id}/summ
```

CodeQualityTaskSummary

props

- path : OpenAPI schema paths
- pathPrefix : api.pathPrefix
- openapiPath : openapi

CRD

```
crds/ArtifactCleanupRun.mdx

ArtifactCleanupRun

<K8sCrd name="artifactcleanupruns.artifacts.katanomi.dev" />
```

ArtifactCleanupRun

props

- name : CRD metadata.name
- crdPath : openapi CRD

```
references/CodeQuality.mdx

CodeQuality

<OpenAPIRef schema="v1alpha1.CodeQuality" />
```

props

- schema : OpenAPI schema definitions (v2) or component/schemas (v3)
- openapiPath : openapi

openapi

OpenAPIPath OpenAPIRef openapi  
openapi openapiPath

```
<OpenAPIPath
 path="/plugins/v1alpha1/template/codeQuality/task/{task-id}/summary"
 openapiPath="shared/openapis/katanomi.json"
/>
```

权限管理

```
<K8sPermissionTable functions={['devops-testplans', 'devops-testmodules']} />
```

权限

props

权限

props

- functions: string[] - 权限资源 FunctionResource 列表

权限

权限	权限	权限 资源	权限 资源	权限 资源	权限 资源	权限 资源	权限 资源
权限 devops- testplans	权限	✓	✓	✓	✓	✓	✗
	权限	✓	✗	✓	✓	✓	✗
	权限	✓	✗	✓	✓	✓	✗
	权限	✓	✗	✓	✓	✓	✗
权限 devops-	权限	✓	✓	✓	✓	✓	✗



模块	测试	测试用例 ID	测试用例 描述	测试用例 预期结果	测试用例 实际结果	测试用例 通过/失败	测试用例 备注
testmodules	测试	✓	×	✓	✓	✓	×
	测试	✓	×	✓	✓	✓	×
	测试	✓	×	✓	✓	✓	×



```
publicBase: # [string] /images/xx.png do
sources:
- name: anchor #
 path: docs/index.mdx# # doom.conf
 ignoreHeading: # [boolean] true
 processors: #
 - type: ejstemplate
 data: # ejs `<%= data.xx %>`
 frontmatterMode: merge # frontmatter ignore ign
```

## frontmatterMode

- `ignore` frontmatter frontmatter
- `merge` frontmatter key
- `replace` frontmatter frontmatter
- `remove` frontmatter

□□□ ACP

--	--	--	--	--	--

□ □ □ □ □ □ □ □

--	--	--	--	--	--	--	--

--	--	--	--	--	--	--	--

[illegible]

doom build # □□□□□□

`doom` `serve` # □□□□□□□□□□

[ci.yaml](#) [Dockerfile](#)

```
FROM build-harbor.alauda.cn/ops/alpine:latest
```

```
WORKDIR /docs
```

```
COPY . dist
```

## ACP

`doom build` `dist` `-v`

```
release-4.0 4.0
doom build -v 4.0 # 4.0 dist/4.0 {base}/4.0
doom build -v master # master dist/master {base}/master
doom build -v {other} # dist/{other} {base}/{other}

unversioned unversioned-x.y
doom build -v unversioned # dist/unversioned {base}
doom build -v unversioned-4.0 # 4.0 dist/unversioned-4.0
```

```
├─ console-platform
│ ├── 4.0
│ ├── 4.1
│ ├── index.html
│ ├── overrides.yaml
│ └─ versions.yaml
├─ console-devops-docs
│ ├── 4.0
│ ├── 4.1
│ ├── index.html
│ ├── overrides.yaml
│ └─ versions.yaml
├─ console-tekton-docs
│ └─ 1.0
```

```

| ├── 1.1
| ├── index.html
| ├── overrides.yaml
| └── versions.yaml

```

## index.html

```

<!DOCTYPE html>
<html>
 <head>
 <title>Redirecting...</title>
 <meta http-equiv="refresh" content="0; url=/console-docs/4.1" />
 </head>
 <body>
 <p>Redirecting to /console-docs/4.1</p>
 </body>
</html>

```

□□□□□□□□

## overrides.yaml

```

console-platform
https://gitlab-ce.alauda.cn/idp/Doom/-/blob/master/src/terms.ts#L11
terms:
 company:
 en: Alauda
 zh: 奥达
 product:
 en: Alauda Container Platform
 zh: 奥达容器平台
 productShort:
 en: ACP

#
title:
 en: Doom - Alauda
 zh: Doom - 奥达
logoText:

```

en: Doom - Alauda  
zh: Doom - 文档

versions.yaml

- '4.1'
- '4.0'

📄📄📄📄📄📄📄

📄📄📄📄📄 [chart-frontend](#) 📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄 [alauda-docs](#) 📄📄📄📄📄  
📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄  
📄📄

📄📄📄📄📄📄📄

📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄  
📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄📄

INFO

📄📄📄📄 PR 📄📄📄gitops 📄📄📄📄