☰   △ **Nuxt** ▾   Q Search CTRL K   🐦 👾 ⌾

createError

You can use this function to create an error object with additional metadata. It is usable in both the Vue and Nitro portions of your app, and is meant to be thrown.

Parameters:

- err: { cause, data, message, name, stack, statusCode, statusMessage, fatal }

Throwing Errors in Your Vue App

If you throw an error created with `createError`:

- on server-side, it will trigger a full-screen error page which you can clear with `clearError`.

- on client-side, it will throw a non-fatal error for you to handle. If you need to trigger a full-screen error page, then you can do this by setting `fatal: true`.

Example

```
                                                                pages/movies/[slug].vue
<script setup lang="ts">
const route = useRoute()
const { data } = await useFetch(`/api/movies/${route.params.slug}`)
if (!data.value) {
  throw createError({ statusCode: 404, statusMessage: 'Page Not Found' })
}
</script>
```

Throwing Errors in API Routes

Use `createError` to trigger error handling in server API routes.

Example

```
export default eventHandler(() => {
  throw createError({
    statusCode: 404,
    statusMessage: 'Page Not Found'
  })
})
```

👉 Read more in Docs > Getting Started > Error Handling.

✎ Edit on Github

Enterprise

Design Kit   NuxtLabs

Nuxt Studio