useLazyAsyncData

`useLazyAsyncData` provides a wrapper around `useAsyncData` that triggers navigation before the handler is resolved by setting the `lazy` option to `true`.

Description

By default, useAsyncData blocks navigation until its async handler is resolved.

> `useLazyAsyncData` has the same signature as `useAsyncData`.

> 👉 Read more in Docs > API > Composables > Use Async Data.

Example

```ts
<script setup lang="ts">
/* Navigation will occur before fetching is complete.
  Handle pending and error states directly within your component's template
*/
const { pending, data: count } = await useLazyAsyncData('count', () => $fetch('/api/count'))

watch(count, (newCount) => {
  // Because count might start out null, you won't have access
  // to its contents immediately, but you can watch it.
})
</script>
```

```
<template>
  <div>
    {{ pending ? 'Loading' : count }}
  </div>
</template>
```

`useLazyAsyncData` is a reserved function name transformed by the compiler, so you should not name your own function `useLazyAsyncData`.

👉 Read more in Docs > Getting Started > Data Fetching #uselazyasyncdata.

✎ Edit on Github

Enterprise

Design Kit

NuxtLabs

Nuxt Studio