

Arbeiten mit REST- und HTTP-APIs in TypeScript

Senden, Empfangen und Verarbeiten von REST-Calls

Die `fetch` API ist die native Methode in JavaScript und TypeScript, um HTTP-Anfragen auszuführen.

Um Daten an eine API zu senden, nutzen wir die `fetch`-Methode mit der HTTP-Methode `POST`.

APIs erlauben oft das Filtern oder Paginieren über Query-Parameter.

Umgang mit HTTP-Clients (Fetch, Axios)

`fetch` ist gut für einfache und native Lösungen.

Axios bietet eine elegantere Syntax, bessere Fehlerbehandlung und Features wie Interceptors.

In Node.js:

```
npm install axios
```

In Deno:

```
import axios from "npm:axios";
```

Asynchronität und Error Handling in API-Aufrufen

Es ist wichtig, asynchrone API-Aufrufe sauber zu verwalten und Fehlerhandling effektiv zu implementieren.

Umgang mit Exceptions und Fehlerprotokollierung

Eine zentrale Protokollierung von Fehlern (Konsole, Datei, Remote-Logging) hilft, Fehler schnell zu identifizieren und zu beheben.

`process.on('unhandledRejection')` ermöglicht die globale Fehlerprotokollierung (nur in Node.js).