Javascript - Beispielhafte Anfrage

Beschreibung:

Diese JavaScript-Funktion ruft Sensordaten von der SmartDog API v2 ab.

Die API liefert Sensordaten für einen bestimmten Zeitraum im JSON-Format zurück.

Programmiersprache: JavaScript (async/await)

API-URL: `https://apiv2.smart-dog.eu/index.php`

Code:

```
async function getSensorData() {
    const url = "https://apiv2.smart-dog.eu/index.php";

    const requestData = {
        action: "getSensorData",
        apikey: "6641d282073d76b625987af5141d3e2a",
        SensorID: "551941",
        UTC_TIMESTAMP_FROM: "1739867939",
        UTC_TIMESTAMP_TO: "1739877939"
    };

    try {
        const response = await fetch(url, {
            method: "POST",
            headers: {
                "Content-Type": "application/json"
            },
            body: JSON.stringify(requestData)
        });
```

```
        if (!response.ok) {
            throw new Error(`HTTP-Fehler! Status: ${response.status}`);
        }

        const data = await response.json();
        document.getElementById("sensorOutput").textContent = JSON.stringify(data, null, 2);
    } catch (error) {
        console.error("Fehler beim Abrufen der Sensordaten:", error);
        document.getElementById("sensorOutput").textContent = "Fehler beim Abrufen der Daten.";
    }
}
```

Beschreibung der Funktion:

1. API-Request vorbereiten:
    - Erstellt eine Anfrage an die API mit den Parametern:
        - `action`: `"getSensorData"`
        - `apikey`: `"6641d282073d76b625987af5141d3e2a"`
        - `SensorID`: `"551941"`
        - `UTC_TIMESTAMP_FROM`: `"1739867939"`
        - `UTC_TIMESTAMP_TO`: `"1739877939"`
2. Daten abrufen mit `fetch()`:
    - Sendet die Anfrage per `POST`-Methode an die API.
    - Setzt den `Content-Type`-Header auf `"application/json"`.
    - Wandelt das `requestData`-Objekt in JSON um und sendet es als `body`.
3. Fehlermanagement:

- Falls der Server keine gültige Antwort sendet (`!response.ok`), wird eine `Error`-Exception geworfen.
- Falls ein Fehler auftritt, wird dieser in der Konsole (`console.error`) ausgegeben und eine Fehlermeldung im HTML-Element mit der ID `"sensorOutput"` angezeigt.

Beispiel für eine erfolgreiche API-Antwort:


```
{
  "sensor_id": 551941,
  "valid": 1,
  "datasets": {
    "1739868002": {
      "DATA": "251",
      "TIMESTAMP_UTC": 1739868002,
      "TIMESTAMP_LOCAL": 1739871602
    },
    "1739868302": {
      "DATA": "257",
      "TIMESTAMP_UTC": 1739868302,
      "TIMESTAMP_LOCAL": 1739871902
    }
  }
}
```