



# Backend Challenge

## Was dich erwartet

Du bist Teil des Entwicklungsteams, dass an einer Anwendung zur Erstellung einer CO2-Bilanz arbeitet. Zwischen dir, dem Frontend Developer und dem Product Owner hat bereits ein erstes Grooming stattgefunden.

Das Frontend wird eine UI implementieren, in dem ein Unternehmen seinen jährlichen Energieverbrauch aufgeteilt auf Energieträger erfassen kann. Diese Daten werden verwendet, um im Backend die CO2-Bilanz zu generieren.

## Aufgabe 1

Implementiere einen API-Endpunkt, der es dem Client ermöglicht, eine Liste von beliebigen Energieträgern samt Verbrauch zu schicken. Je Energieträger wird dir der Client folgende Daten schicken:

Feld	Datentyp	Validierung
Beschreibung	Text	• Pflichtfeld
ID des Energieträgers	Text	• Pflichtfeld
Energieverbrauch (in kg, Liter, km etc.)	Dezimalzahl	• maximal 5 Nachkommastellen • Pflichtfeld
Individueller Emissionsfaktor	Dezimalzahl	• maximal 5 Nachkommastellen • optional

Basierend auf den Energieträgerdaten musst du nun die CO2-Bilanz generieren und dem Client als Response zurücksenden.

Zur Erstellung der CO2-Bilanz stehen dir folgende Endpunkte anderer Systeme zur Verfügung:

- **Scope-Hierarchie**
  - Eine CO2-Bilanz ist hierarchisch nach vorgegebenen Scopes gegliedert.
  - Erfasste Energieträger werden mit ihrem Verbrauch einem festen Scope zugeordnet.
- **Energieträger-Stammdaten**
  - Übersicht aller möglichen Energieträger und welchem Scope sie zugeordnet sind.

Gehe davon aus, dass die API immer erreichbar ist und schnell antwortet. Caching und explizite Fehlerbehandlung sind nicht notwendig.

Mit dem Frontend Developer hast du eine **Response in dieser Form** abgestimmt. Es handelt sich hier lediglich um ein Beispiel, dass du zur Orientierung heranziehen kannst.

Vom Produkt Owner hast du folgende Hinweise erhalten:

- Das Label (→ **label** ) einer Position auf dritter Ebene ist immer nach folgendem Muster aufgebaut: *Name des Energieträgers (Beschreibung)*, bspw. *Heizöl leicht (Standort München)*
- Der Energieverbrauch wird abhängig vom Energieträger in kg, Liter, km etc. erfasst. Zur Berechnung der Energie in kWh (→ **energy** ) muss der Energieverbrauch mit dem angegebenen Konversionsfaktor (siehe **Energieträger-Stammdaten**) multipliziert werden.
- Die berechnete Energie in kWh wird mit dem angegebenen Emissionsfaktor (siehe **Energieträger-Stammdaten**) multipliziert (→ **co2** ). Ist ein individueller Emissionsfaktor vorhanden, so wird dieser für die Berechnung verwendet. Der so errechnete Wert ist abschließend durch 1000 zu dividieren, da die Angabe in Tonnen erfolgt.

- Zu jedem Scope wird je eine Summe der Energie sowie der Emissionen der darunter liegenden Positionen angezeigt.

## Aufgabe 2

Der implementierte Endpunkt darf nur mit einem gültigen JWT aufgerufen werden. Berücksichtige folgende Anforderungen:

- der Client sendet das JWT als Bearer Token im Authorization Header
- das Backend soll das Token validieren (Signatur und Gültigkeit prüfen)
- Ein eigener Login-Endpoint ist nicht notwendig. Es ist ausreichend, wenn du ein JWT mit einer Gültigkeit von bspw. 15 Minuten zu Testzwecken so einfach wie möglich erzeugst (z. B. beim Starten auf der Konsole ausgeben)

## Allgemeine Hinweise

- Verwende eine Programmiersprache deiner Wahl.
- Es geht hier nicht um richtig oder falsch. Wir wollen vor allem deine Herangehensweise verstehen und dir bei dieser Gelegenheit auch einen Einblick in unsere Projektarbeit bieten.
- Triff gerne Annahmen an Stellen, wo die Aufgabenstellung dir keine konkreten Vorgaben macht.

