

Hausübung 01: Wetter-App

In dieser Übung erstellen Sie eine einfache Wetter-App die es Nutzer*innen ermöglicht, die aktuellen Wetterdaten für eine beliebige Postleitzahl/Stadt abzurufen.

Funktionsweise

Ablauf

1. **Eingabe:** Der*Die Nutzer*in gibt den Namen einer Stadt und/oder eine Postleitzahl in die Eingabefelder ein.
2. **Suchanfrage:** Nach Betätigung des "Suchen"-Buttons werden die eingegebenen Daten an eine Geocoding-API (Nominatim) gesendet, um die geografischen Koordinaten (Breiten- und Längengrad) des Ortes zu ermitteln.
 - Falls die API keine Ergebnisse liefert, wird eine Fehlermeldung angezeigt.
 - Falls die API mehrere Ergebnisse liefert, überlegen Sie, welches Ergebnis Sie verwenden möchten, z.B., könnten Sie österreichische Orte bevorzugen, oder immer das erste Ergebnis verwenden.
 - Verwenden Sie zum Speichern der Koordinaten ein Objekt der Klasse `LocationData`, das zumindest die Attribute `latitude` und `longitude` enthält. Optional können Sie zusätzlich den Namen des Ortes (im JSON response unter `display_name` verfügbar) für die Anzeige speichern.
3. **Wetterdaten abrufen:** Mit den ermittelten Koordinaten wird eine Anfrage an eine Wetter-API (Open-Meteo) gesendet. Diese liefert die aktuellen Wetterdaten für den angegebenen Standort, wie Temperatur und Wettercode.
 - Die API-Antwort wird verarbeitet und die relevanten Daten werden in einem Objekt der Klasse `WeatherData` gespeichert. Dieses Objekt enthält den Namen der Stadt, den Wettercode und die Temperatur.
4. **Anzeige:** Die Wetterdaten werden auf der Webseite angezeigt. Dazu gehören:
 - Der Name des gewählten Ortes
 - Ein Icon, das die Wetterlage repräsentiert (basierend auf dem Wettercode)
 - Die aktuelle Temperatur in Grad Celsius
 - Eine textuelle Beschreibung der Wetterlage (basierend auf dem Wettercode)

Wettercodes

Die Wetter-API verwendet numerische Wettercodes, um die Wetterlage zu beschreiben. Diese Codes werden in der App in textuelle Beschreibungen und Icons übersetzt. Beispielsweise steht der Code `0` für "Klarer Himmel" und `61` für "Leichter Regen". Die Übersetzung von Wettercodes in textuelle Beschreibungen und Icons kann über die bereitgestellten Funktionen `getWeatherDescription` und `getWeatherIcon` erfolgen, die einen Wettercode als Parameter akzeptieren und die entsprechende Beschreibung bzw. das entsprechende Icon zurückgeben.

Gestaltung der Webseite

Verwenden Sie CSS um die Webseite zu gestalten. Für die Eingabe des Ortes verwenden Sie zwei Textfelder, zum Absenden der Anfrage einen Button. Fehlermeldungen sollen ebenfalls auf der Seite angezeigt werden - natürlich nur bei Bedarf.

Vorlage

Im Moodle finden Sie eine Vorlage als Zip-Datei, die das Grundgerüst der Wetter-App enthält. Sie können die Entwicklung mit dieser Vorlage starten und sie einfach erweitern. Alternativ kann die Vorlage aber auch beliebig verändert oder ganz ignoriert werden.

Beispiel

So könnte die Wetter-App beispielsweise aussehen:

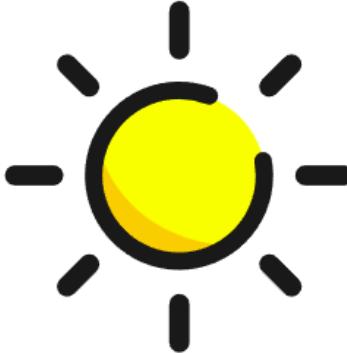
The screenshot shows a web browser window titled "Wetter App". The URL in the address bar is "ue01/hue01/index.html?_jt=t6677og8u055pdq5n657b9oovt". The page content is as follows:

Wetter App

Bitte geben Sie PLZ und/oder Ort ein, um das Wetter abzufragen.

PLZ: Stadt: Suchen

Linz, Oberösterreich, Österreich



18.1 °C

Klarer Himmel

Sie dürfen Ihre Webseite gerne ansehnlicher gestalten als ich...

Technologien

- **HTML:** Für die Struktur der Webseite.
- **CSS:** Für die Gestaltung der Webseite.
- **JavaScript:** Für die Logik der App, einschließlich:
 - Abrufen der Geokoordinaten/Wetterdaten mittels fetch API.
 - Verarbeitung der API-Antworten mittels asynchronem JavaScript und JSON.
 - Darstellung der Wetterdaten auf der Webseite mittels DOM-Manipulation.
 - Objektorientierung mit Klassen für Wetterdaten und Ortsdaten.

APIs

Nominatim

Eine Geocoding-API von OpenStreetMap, die kostenlos und ohne Registrierung verwendet werden kann. Eine Beispiel URL für eine Suchanfrage könnte so aussehen:

<https://nominatim.openstreetmap.org/search?format=json&city=Berlin&postalcode=12345>

- `format=json` : legt fest, dass die Antwort im JSON-Format zurückgegeben wird.
- `city=Berlin` : gibt den gesuchten Ort an (hier: Berlin).
- `postalcode=12345` : gibt die Postleitzahl des gesuchten Ortes an (hier: 12345).

Die API gibt eine **Liste** von Orten zurück, die zur Suchanfrage passen. Jeder Ort enthält Informationen wie Breiten- und Längengrad, Postleitzahl, Anzeigename des Ortes etc. Wird der Ort nicht gefunden, so wird ein leeres Array zurückgegeben.

Es genügt, wenn Sie für Ihre Anfrage Postleitzahl oder Ort verwenden. Sie können auch beide verwenden, um die Suche zu präzisieren.

Open-Meteo

Eine Wetter-API, die ebenfalls kostenlos und ohne Registrierung verwendet werden kann. Eine Beispiel URL für eine Wetteranfrage könnte so aussehen:

https://api.open-meteo.com/weather?current_weather=true&latitude=52.52&longitude=13.405

- `current_weather=true` : gibt an, dass die aktuellen Wetterdaten abgefragt werden.
- `latitude=52.52` : gibt den Breitengrad des Ortes an (hier: Berlin).
- `longitude=13.405` : gibt den Längengrad des Ortes an (hier: Berlin).

Die API gibt die aktuellen Wetterdaten für den angegebenen Ort zurück, einschließlich Temperatur und Wettercode.