

Analyse und Visualisierung von Wetterdaten

- **Kernidee:** Abruf, Analyse und Darstellung von Wetterdaten für einen bestimmten Ort.
- **Datenquelle:** Kostenlose Wetter-APIs wie [OpenWeatherMap](#) oder [Meteostat](#).
- **Mögliche Umsetzungen:**
 - **Grundversion:** Ein Skript, das für eine feste Stadt die aktuelle Temperatur, Luftfeuchtigkeit und Wetterbeschreibung ausgibt.
 - **Grafische Version:** Eine Desktop-Anwendung (mit **Tkinter/PyQt**) oder eine kleine Webseite, auf der ein Benutzer einen Ort eingeben kann und die aktuellen Wetterdaten sowie eine Vorhersage für die nächsten Tage erhält.
- **Optionale Erweiterungen & Vertiefungen:**
 - **Historischer Vergleich:** Visualisierung von Temperatur- oder Niederschlagsverläufen für den aktuellen Monat im Vergleich zum gleichen Monat der Vorjahre.
 - **Interaktive Karte:** Nutzung von **Folium**, um Wetterdaten (z.B. Temperaturen oder Windgeschwindigkeiten) für mehrere Orte gleichzeitig auf einer Weltkarte darzustellen.
 - **Agrar-Dashboard:** Spezialisierte Ansicht, die für Landwirte relevante Daten wie die Niederschlagsmenge der letzten 30 Tage oder die Anzahl der Sonnenstunden anzeigt.
 - **KI-Anwendung (Mustererkennung):** Analyse historischer Daten, um Korrelationen zu finden (z.B. "Wie hängt die Windgeschwindigkeit mit schnellen Temperaturänderungen zusammen?"). Anwendung von Clustering, um typische "Wetterprofile" für eine Jahreszeit zu identifizieren.
- **Hilfreiche Python-Bibliotheken:**
 - **Datenbeschaffung:** requests (für die Kommunikation mit der Wetter-API).
 - **Datenverarbeitung:** pandas (zur Handhabung der Zeitreihendaten).
 - **UI:** Tkinter (in Python enthalten), PyQt (umfangreicher), Flask (für eine Weboberfläche).
 - **Visualisierung:** Matplotlib, Seaborn (für ansprechendere Graphen), Folium (für interaktive Karten).
 - **KI & Statistik:** Scikit-learn (für Clustering und Korrelationsanalyse)