

Datenverarbeitung und maschinelles Lernen WS 2024/25 Übung 1 Einleitung

Aufgabe 1: Extrahieren von Kontaktinformationen aus E-Mails

Schreiben Sie eine Funktion zur Verarbeitung von E-Mail-Adressen und zum Extrahieren des Vor- und Nachnamens sowie des Landes eines jeden Kontakts auf der Grundlage der E-Mail-Domäne. Speichern Sie diese Informationen in "Kontakte_info.csv" mit Spalten: Vorname, Nachname, Land.

Beispiel-E-Mails:

john.doe@unige.ch
maria.rossi@uniroma.it
carlos.lopez@ub.es
anne.schmidt@tum.de
sofia.kowalska@uj.pl

Tipp: Verwenden Sie `'split()'`, um Namen und Domänen zu trennen, und ein Wörterbuch, um Domänensuffixe auf Ländernamen abzubilden.

Aufgabe 2: Wöchentliche Wetterdatenanalyse

Schreiben Sie ein Programm zur Analyse der täglichen Temperaturdaten für jede Stunde über eine Woche.

1. Definieren Sie eine Funktion zur Analyse der Daten eines einzelnen Tages (Maximum, Minimum, Durchschnitt, Bereich).
2. Definieren Sie eine weitere Funktion, um die Daten der gesamten Woche zu analysieren und die Tages- und Wochenstatistiken zusammenzufassen.

Tipp: Verwenden Sie `'max()'`, `'min()'` und einfache Arithmetik für die Tagesstatistiken. Schleife durch jeden Tag, um wöchentliche Zusammenfassungen zu erhalten.

Aufgabe 3: Schüler-Gradebook-System

Erstellen Sie ein Notenbuchsystem, um:

1. Fügen Sie einen Schüler hinzu und vergeben Sie Noten.
2. Berechnen Sie die Durchschnittsnote jedes Schülers.
3. Ermitteln Sie die höchsten und niedrigsten Punktzahlen für ein bestimmtes Fach.

Tipp: Verwenden Sie ein Wörterbuch, um Schüler und Noten zu speichern. Für Durchschnittswerte verwenden Sie `'sum()'` und `'len()'` für die Noten der Schüler.

Aufgabe 4: Was sind die Hauptmerkmale von Python, die es für verschiedene Programmieraufgaben beliebt machen?

Aufgabe 5: Wie geht Python mit dynamischer und starker Typisierung um und was sind die Vorteile und möglichen Fallstricke dieses Ansatzes? Nennen Sie ein Beispiel, bei dem die dynamische Typisierung sowohl hilfreich als auch problematisch sein kann.

Aufgabe 6: Erkläre, wie unterschiedlich die for- und while-Schleifen in Python funktionieren. Beschreiben Sie ein Szenario, in dem die Verwendung einer for-Schleife sinnvoller ist als die einer while-Schleife und andersherum.