



Ich habe eine CSV Datei im Format:

```
29-04-2023 18:10:26; 19.25
29-04-2023 18:10:27; 19.25
29-04-2023 18:10:28; 19.25
29-04-2023 18:10:29; 19.25
29-04-2023 18:10:31; 19.25
29-04-2023 18:10:32; 20.5
29-04-2023 18:10:33; 21.5
29-04-2023 18:10:34; 22
29-04-2023 18:10:35; 22.5
29-04-2023 18:10:37; 22.75
29-04-2023 18:10:38; 23
29-04-2023 18:10:39; 22.75
29-04-2023 18:10:40; 22.5
29-04-2023 18:10:41; 22.25
29-04-2023 18:10:43; 22
29-04-2023 18:10:44; 21.75
29-04-2023 18:10:45; 21.5
29-04-2023 18:10:46; 21.5
29-04-2023 18:10:48; 21.25
29-04-2023 18:10:49; 21.25
29-04-2023 18:10:50; 21.75
29-04-2023 18:10:51; 22.25
29-04-2023 18:10:52; 22.75
29-04-2023 18:10:54; 23
29-04-2023 18:10:55; 23.25
29-04-2023 18:10:56; 23.5
29-04-2023 18:10:57; 23
29-04-2023 18:10:58; 22.75
29-04-2023 18:11:00; 22.5
29-04-2023 18:11:01; 22.5
29-04-2023 18:11:02; 22.25
29-04-2023 18:11:03; 22
29-04-2023 18:11:05; 22
29-04-2023 18:11:06; 22.5
29-04-2023 18:11:07; 23
29-04-2023 18:11:08; 23.25
29-04-2023 18:11:09; 23.5
29-04-2023 18:11:11; 23.5
29-04-2023 18:11:12; 23.25
29-04-2023 18:11:13; 23
29-04-2023 18:11:14; 22.75
29-04-2023 18:11:15; 22.5
29-04-2023 18:11:17; 22.25
29-04-2023 18:11:18; 22.25
29-04-2023 18:11:19; 22
29-04-2023 18:11:20; 21.75
29-04-2023 18:11:21; 21.75
29-04-2023 18:11:23; 21.5
29-04-2023 18:11:24; 21.5
29-04-2023 18:11:25; 21.25
29-04-2023 18:11:26; 21.25
29-04-2023 18:11:28; 21
29-04-2023 18:11:29; 21
29-04-2023 18:11:30; 20.75
29-04-2023 18:11:31; 20.75
29-04-2023 18:11:32; 20.75
29-04-2023 18:11:34; 20.5
29-04-2023 18:11:35; 20.5
29-04-2023 18:11:36; 20.5
29-04-2023 18:11:37; 20.5
29-04-2023 18:11:38; 20.25
```



Die gegebene CSV-Datei enthält eine Zeitstempelspalte und eine Wertspalte, wobei der Wert in Dezimalzahlen angegeben wird. Die Trennung zwischen den beiden Spalten erfolgt durch ein Semikolon.



Um diese Datei zu lesen und weiterzuverarbeiten, können Sie eine Programmiersprache wie Python verwenden. Hier ist ein Beispielcode, der die Datei öffnet, die Zeitstempel und Werte extrahiert und sie in separate Listen speichert:

```
python Copy code

import csv
from datetime import datetime

timestamps = []
values = []

with open('beispiel.csv') as csvfile:
    reader = csv.reader(csvfile, delimiter=';')
    for row in reader:
        timestamp = datetime.strptime(row[0], '%d-%m-%Y %H:%M:%S')
        value = float(row[1])
        timestamps.append(timestamp)
        values.append(value)
```

Dieser Code verwendet die CSV-Module in Python, um die Datei zu öffnen und Zeilen zu lesen. Der `datetime`-Typ wird verwendet, um die Zeitstempel in ein verwendbares Format zu konvertieren, und `float` wird verwendet, um den Wert in eine Dezimalzahl zu konvertieren. Schließlich werden die Zeitstempel und Werte in separate Listen gespeichert.

Nachdem die Daten extrahiert wurden, können Sie sie auf verschiedene Weise analysieren oder visualisieren, je nach Ihren Bedürfnissen. Zum Beispiel können Sie die Werte als Zeitreihenplot mit matplotlib plotten:

```
python Copy code

import matplotlib.pyplot as plt

plt.plot(timestamps, values)
plt.xlabel('Zeit')
plt.ylabel('Werte')
plt.show()
```

Dieses Code-Segment zeichnet ein einfaches Liniendiagramm mit den Zeitstempeln auf der x-Achse und den Werten auf der y-Achse. Es kann je nach Bedarf angepasst werden, um den Plot zu optimieren oder um andere Arten von Visualisierungen zu erstellen.