

# dataframe

Ein **DataFrame** ist eine zentrale Datenstruktur in der **Pandas-Bibliothek** von Python.

Konkret: Eine 2 dimensionale Tabelle mit Zeilen und Spalten.

Vgl: Excell-Tabelle bzw. SQL-Tabelle

- ➔ zB Gute Interaktion mit CSV Dateien
- ➔ Verschiedene Datentypen möglich!
- eine **Tabelle** mit **beschrifteten Zeilen und Spalten (vgl wieder Excell oder SQL-Tabelle)**
- sehr leistungsfähig für **Datenanalyse, Filtern, Aggregation, Visualisierung**
- Teil der pandas Bibliothek (import pandas as pd): Daher muss die pandas Bibliothek installiert werden bevor Sie es verwenden können.

## Beispielcode

```
import pandas as pd

# Beispiel-Daten
daten = {
    'Bestellnummer': [1001, 1002, 1003, 1004],
    'Datum': ['2025-03-01', '2025-03-02', '2025-03-02', '2025-03-03'],
    'Produkt': ['T-Shirt', 'Hose', 'T-Shirt', 'Schuhe'],
    'Kategorie': ['Kleidung', 'Kleidung', 'Kleidung', 'Schuhe'],
    'Preis': [19.99, 39.99, 19.99, 59.99],
    'Anzahl': [2, 1, 3, 1]
}

# DataFrame erstellen
df = pd.DataFrame(daten)
print(df)
```

## Zugriff auf Daten

```
# Erste Zeilen anzeigen
print(df.head())

# Nur bestimmte Spalten anzeigen
print(df[['Produkt', 'Preis']])

# Filter: Nur T-Shirts anzeigen
print(df[df['Produkt'] == 'T-Shirt'])

# Neue Spalte: Gesamtpreis
df['Gesamtpreis'] = df['Preis'] * df['Anzahl']
print(df)
```

## Plot erstellen

```
import matplotlib.pyplot as plt

# Gesamtumsatz pro Kategorie
umsatz_kat = df.groupby('Kategorie')['Gesamtpreis'].sum()

# Balkendiagramm
umsatz_kat.plot(kind='bar', title='Umsatz pro Kategorie')
plt.xlabel('Kategorie')
plt.ylabel('Umsatz in €')
plt.tight_layout()
plt.show()
```

## Speichern als CSV

```
# Als CSV speichern
df.to_csv('verkaeufe.csv', index=False)

# Einlesen einer CSV-Datei
df_neu = pd.read_csv('verkaeufe.csv')
print(df_neu)
```

## Weitere “einfache” Beispiele

```
# Nach Datum sortieren
df.sort_values(by='Datum', ascending=True)

# Summe aller Verkäufe
gesamtumsatz = df['Gesamtpreis'].sum()
print(f"Gesamtumsatz: {gesamtumsatz:.2f} €")
```

```
# Nur Bestellungen nach dem 01.03.2025
df['Datum'] = pd.to_datetime(df['Datum'])
print(df[df['Datum'] > '2025-03-01'])
```