

Serie 11 - Externe Pakete und Projekte

Programmieren für Naturwissenschaften FS 2023

Gruppe: Sofia Kessler, Florian Mohaupt, Lukas Batschelet

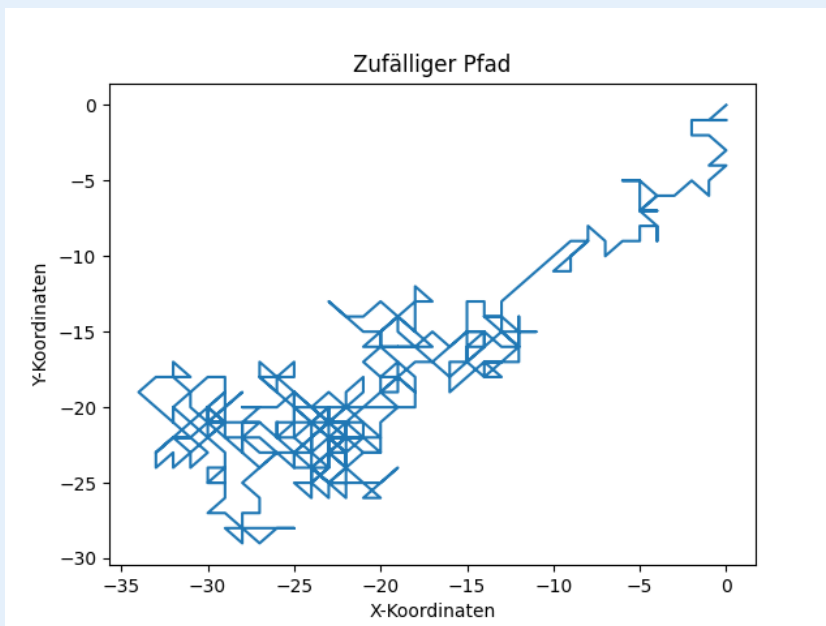
Aufgabe : matplotlib

Aufgabe

Ziel dieser Aufgabe ist es, eine Funktion `random_path` zu schreiben. Diese Funktion erhält als Parameter eine Zahl und generiert dann einen zufälligen Pfad dieser Länge. Die Funktion soll den Pfad dann mit Hilfe des Moduls `matplotlib` anzeigen. Der Pfad soll bei den Koordinaten (0, 0) beginnen und dann in jedem weiteren Schritt die x- und y-Koordinaten so verändern, dass der jeweilige Wert entweder um 1 verkleinert wird, gleich bleibt oder um 1 erhöht wird. Daraus ergeben sich dann 8 verschiedene Richtungen, die zufällig in jedem Schritt generiert werden.

Tipps:

1. Erstellen Sie zwei Listen. Eine enthält alle x-Koordinaten, die andere alle y-Koordinaten.
2. Verwenden Sie die Funktion `randint` des Moduls `random`.
3. Sie können den Pfad anzeigen, indem sie die Funktion `plot` des Moduls `matplotlib.pyplot` verwenden.
4. Die Funktion `plot` erhält als ersten Parameter alle x-Koordinaten und als zweiten Parameter alle y-Koordinaten in Form einer Liste.



Info

Falls das Modul `matplotlib` noch nicht installiert ist, kannst du es mit folgendem Befehl in der Konsole installieren:

```
pip install matplotlib
```

Mögliche Lösung

```
import matplotlib.pyplot as plt
import random

def random_path(length):
    x = [0]
    y = [0]
    for _ in range(length):
```

```
dx = random.randint(-1, 1)
dy = random.randint(-1, 1)
new_x = x[-1] + dx
new_y = y[-1] + dy
x.append(new_x)
y.append(new_y)
plt.plot(x, y)
plt.xlabel('X-Koordinaten')
plt.ylabel('Y-Koordinaten')
plt.title('Zufälliger Pfad')
plt.show()
```

```
# Beispielaufruf der Funktion mit einer Länge von 100
random_path(500)
```

Aufgabe 2: pandas

Aufgabe

Auf Ilias finden Sie die Datei `covid-data.csv` ^[1]. Lesen Sie diese mit Hilfe des Pakets pandas ein und erfüllen Sie folgende Aufgaben:

1. Machen Sie sich zunächst mit der Tabelle vertraut. Welche Daten werden abgebildet? Welche Spalten gibt es?
2. Geben Sie für jede Ortschaft (location) die totalen Fälle (total_cases) sowie die totalen Todesfälle (total_deaths) an.
3. Fügen Sie dem in Schritt 2 berechneten Dataframe eine neue Spalte (death_rate) hinzu. In dieser Spalte soll die Todesrate vom jeweiligen Land berechnet werden (dieser Wert muss selber berechnet werden).

Tipp: Um einem Dataframe `df` eine neue Spalte `address` hinzufügen zu können, kann `df["address"] = values` gebraucht werden, wobei `values` für die Werte in der Spalte stehen.

1. (Zusatz) Geben Sie die 10 Länder mit den höchsten Todesraten zurück.

Tipp: Verwenden Sie die pandas-Funktion `sort_values()`.

Info

Für die Bearbeitung dieser Aufgabe ist es erforderlich den Datensatz `covid-data.csv` herunterzuladen und im gleichen Verzeichnis wie das Python-Skript zu speichern.

Falls du das Paket `pandas` noch nicht installiert hast, kannst du es mit folgendem Befehl in der Konsole installieren:

```
pip install pandas
```

Mögliche Lösung

```
import pandas as pd

# 1. Daten einlesen und Tabelle anzeigen
df = pd.read_csv("covid-data.csv")
print("Erste 5 Zeilen der Tabelle:")
print(df.head())
print("Spalten in der Tabelle:")
print(df.columns)

# 2. Totale Fälle und Todesfälle für jede Ortschaft
grouped_data = df.groupby('location').agg({
    'total_cases': 'last',
    'total_deaths': 'last'
}).reset_index()

print("Totale Fälle und Todesfälle für jede Ortschaft:")
print(grouped_data)

# 3. Todesrate berechnen und als neue Spalte hinzufügen
grouped_data['death_rate'] = (grouped_data['total_deaths'] / grouped_data['total_cases']) * 100

print("Tabelle mit Todesrate:")
print(grouped_data)

# 4. Die 10 Länder mit den höchsten Todesraten
top_death_rates = grouped_data.sort_values(by='death_rate', ascending=False).head(10)
print("Die 10 Länder mit den höchsten Todesraten:")
print(top_death_rates)
```

1. Quelle: <https://github.com/owid/covid-19-data/tree/master/public/data>↵