

## Thema des Projekts

Das Projekt behandelt ein aktuelles Thema: „Kohlendioxid (CO<sub>2</sub>)-Emissionen aus fossilen Brennstoffen und der Industrie“.

Die Originaldaten stammen von der öffentlichen Webseite *ourworldindata.org* und wurden als CSV-Datei heruntergeladen:

<https://ourworldindata.org/explorers/co2>

Die Datei enthält rund 26.000 statistische Datensätze, die von verschiedenen Ländern vom 18. Jahrhundert bis heute erhoben wurden.

## Beschreibung des Projekts

Das im Rahmen des Projekts entwickelte Programm ermöglicht die Auswahl verschiedener Parameter für die Diagrammdarstellung sowie die Erstellung von Plots auf Basis der Bibliothek **Matplotlib**. Die benutzerfreundliche Oberfläche wurde mit **Tkinter** erstellt.

Die einstellbaren Darstellungsparameter sind:

- Der Titel des Plots.
- Die Auswahl eines oder mehrerer Länder aus der Liste (China, Deutschland, Indien, Großbritannien, USA)<sup>1</sup>.
- Die Farbe der Kurve für jedes ausgewählte Land<sup>2</sup>.
- Der Plot-Typ (Line Plot, Scatter Plot oder Stack Plot).

Alle ausgewählten Informationen werden im Hauptfenster angezeigt, bevor das Diagramm erstellt wird, sodass der Nutzer jederzeit den Überblick behält.

Zusätzlich bietet das Programm eine Funktion zur **Konvertierung zwischen CSV- und JSON-Format** (CSV → JSON und JSON → CSV) sowie die Möglichkeit, mit beiden Formaten weiterzuarbeiten.

## Technische Beschreibung des Programms

Für den Betrieb benötigt das Programm folgende Dateien:

- **Projekt\_python.py** – enthält den Code der Hauptklasse MainGUI;
- **projekt\_add\_functions.py** – enthält zusätzliche Klassen und Funktionen;
- **co2.csv** – die Originaldatendatei;
- **python\_icon.png** – ein Bild, das in einem Toplevel-Fenster verwendet wird.

Das Programm wird aus der Datei **Projekt\_python.py** mit dem Befehl **MainGUI()** gestartet.

Die Benutzeroberfläche kann mit Maus und Tastatur bedient werden. Bei fehlerhaften oder unvollständigen Eingaben wird eine entsprechende Fehlermeldung ausgegeben.

---

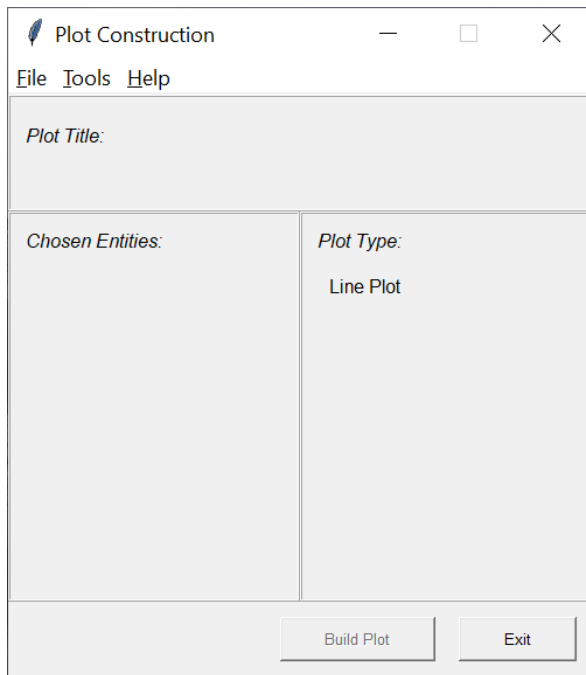
<sup>1</sup> Die Auswahl der Länder basiert auf den Quellen, für die die meisten Informationen gesammelt wurden.

<sup>2</sup> Die Farbe kann entweder aus den Standardfarben oder aus einer Farbpalette ausgewählt werden.

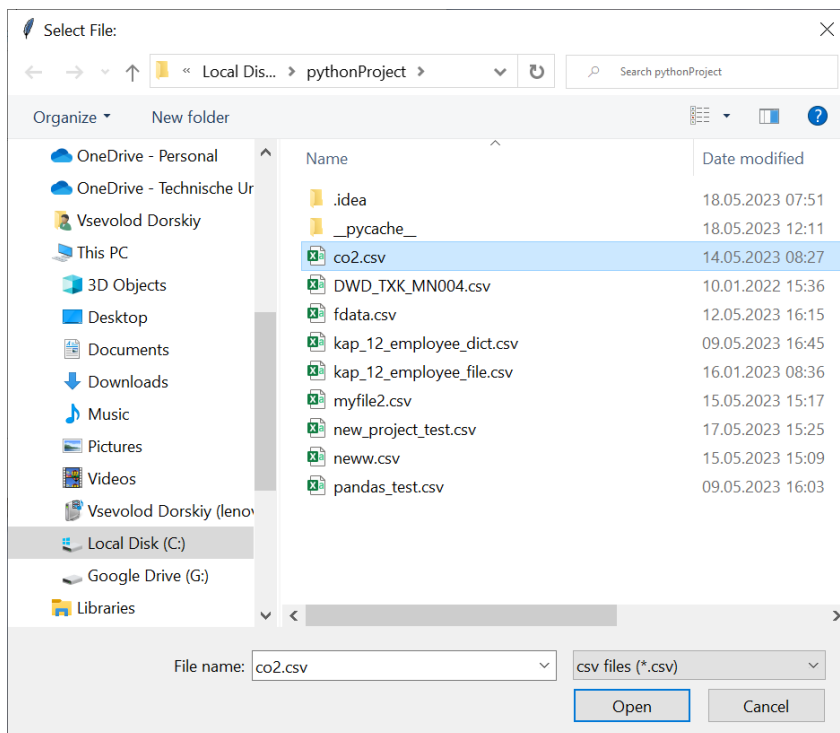
# Funktionsablauf des Programms

## Auswahl der Datei und Initialisierung

Nach dem Programmstart erscheint ein Hauptfenster:

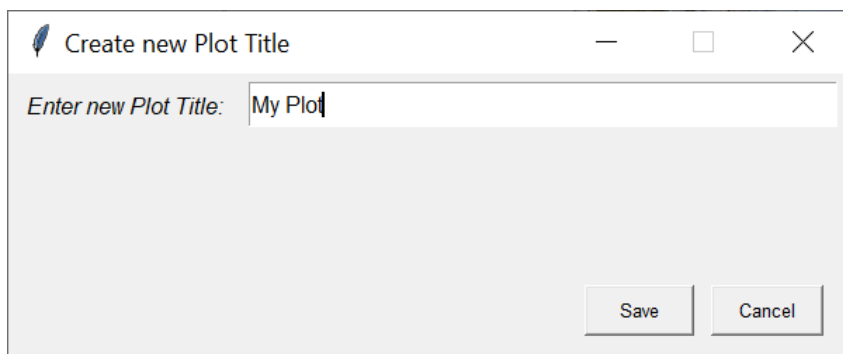


Zunächst muss im Dialogfenster die Datei **co2.csv** ausgewählt und eingelesen werden. Nach erfolgreichem Laden erscheint eine Bestätigungsmeldung, und die Optionen zur Parametereinstellung sowie die Schaltfläche *Build Plot* werden freigeschaltet.

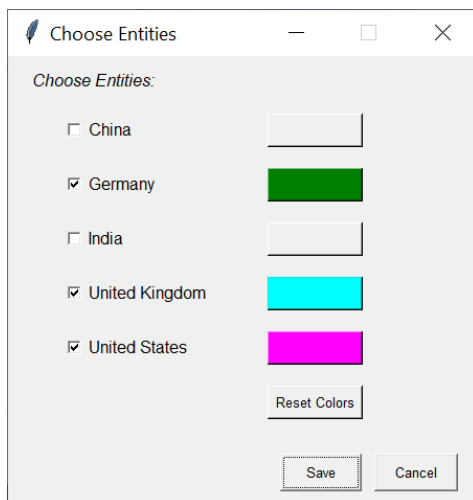


## Festlegung der Plot-Parameter

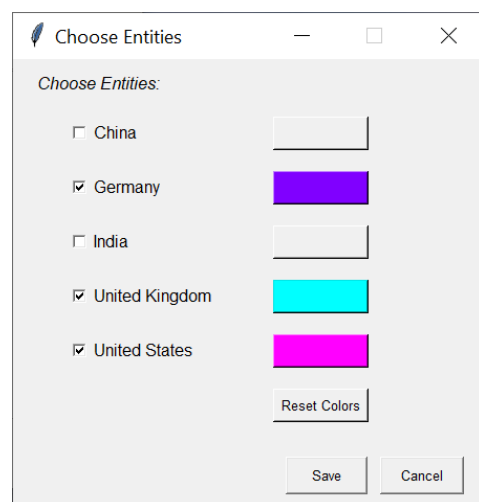
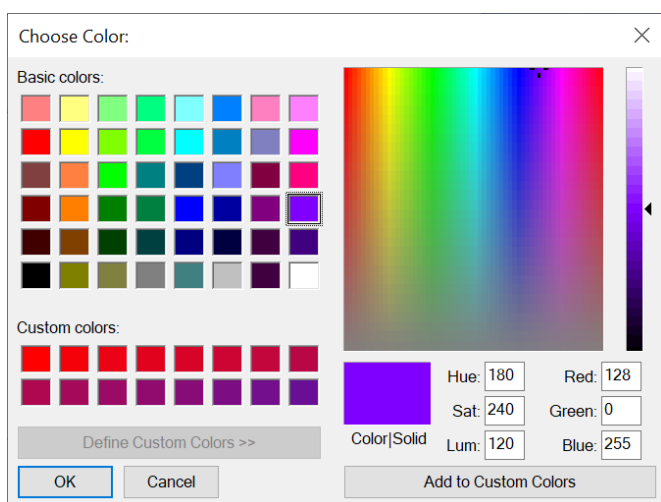
- Der Titel wird in einem Toplevel-Fenster (*Create new Plot Title*) eingegeben (zum Beispiel: My Plot):



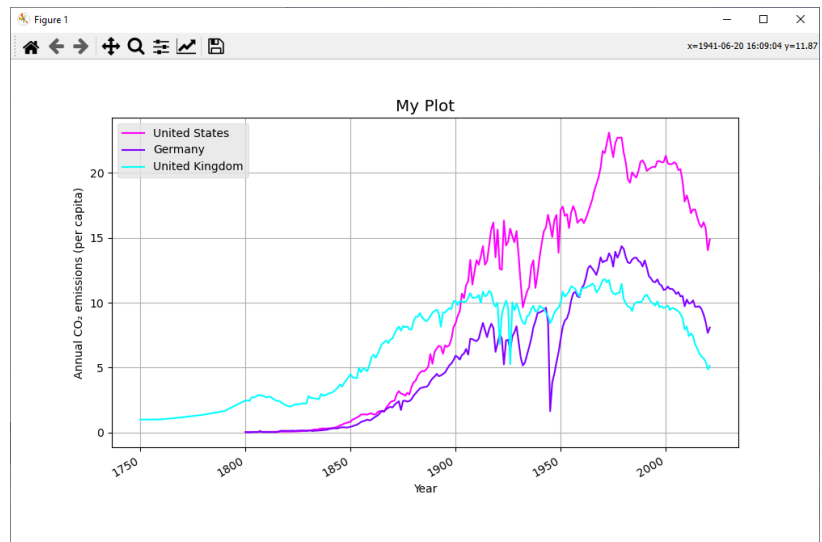
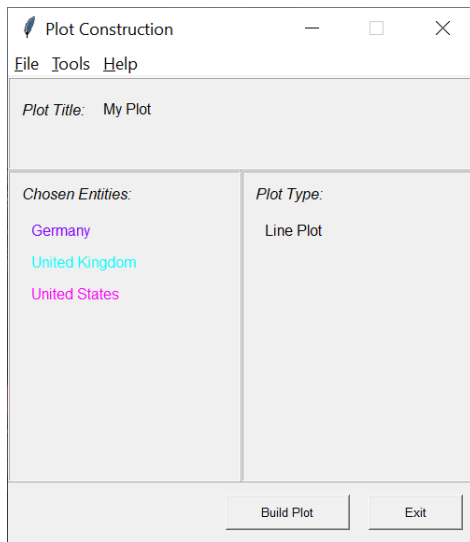
- Die gewünschten Länder werden ausgewählt und erhalten jeweils eine Darstellungsfarbe:



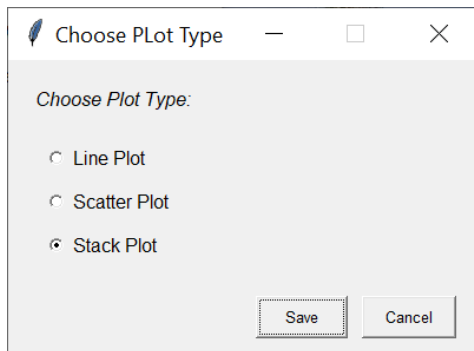
- Farben können frei aus einer Palette gewählt werden (z.B. neue Farbe für Germany):



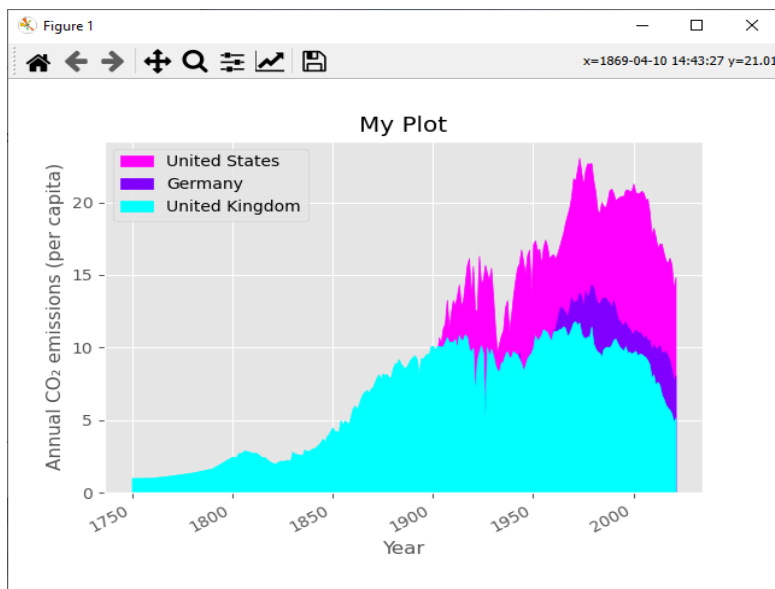
Alle ausgewählten Parameter werden im Hauptfenster zusammengefasst angezeigt:



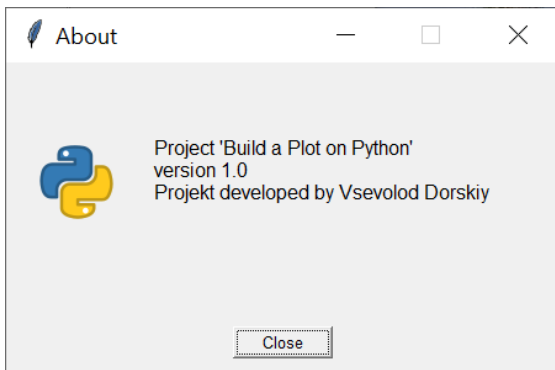
- Der Standardwert für *Plot Type* ist *Line Plot*. Der Plot-Typ kann geändert werden (z. B. *Line Plot* → *Stack Plot*):



Beim Erstellen des Plots werden die Daten so sortiert, dass die Darstellung übersichtlich bleibt. Zusätzlich wird intern der Datensatz mit dem höchsten Wert ermittelt:



Das Fenster *About* enthält einen Animationstext sowie ein eingebundenes Bild aus der Datei *python\_icon.png*.



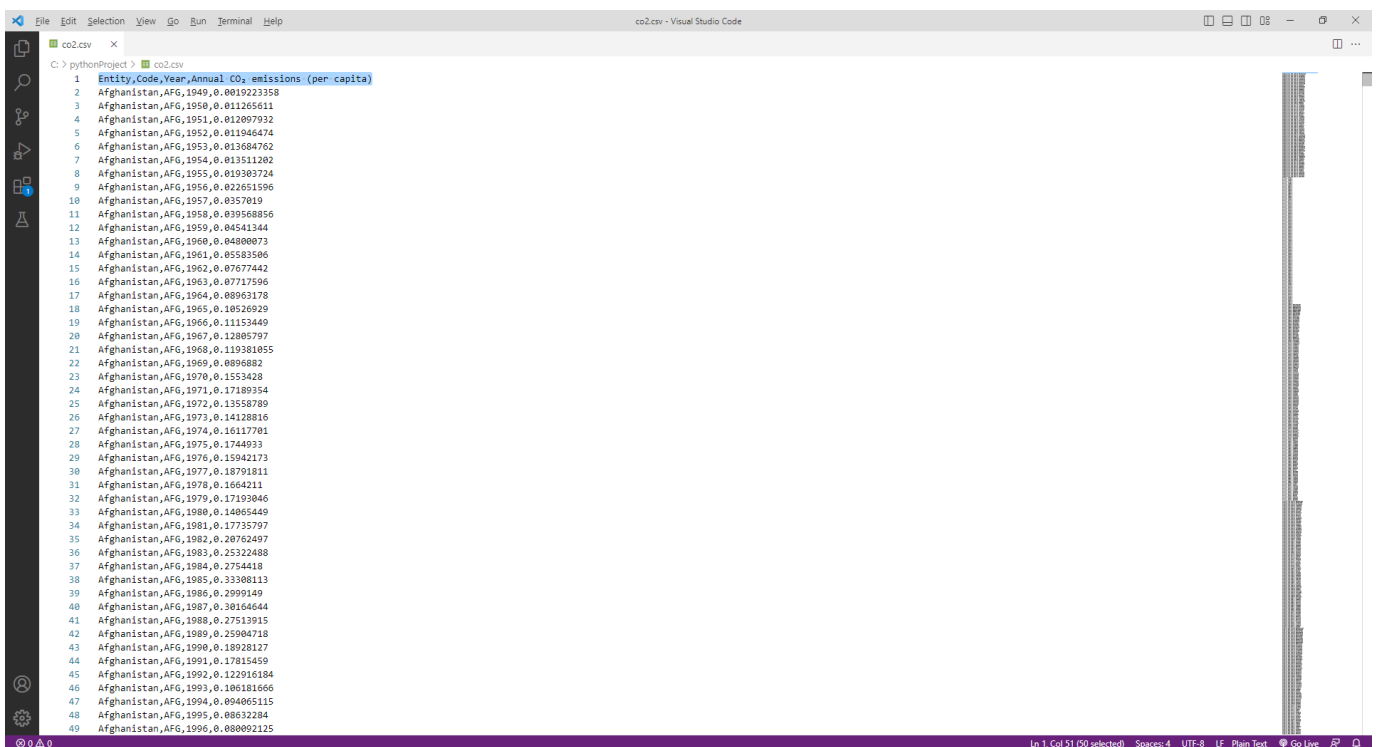
## Speichern und Dateikonvertierung

Das Programm unterstützt folgende Funktionen:

- **CSV → JSON**
- **JSON → CSV.**

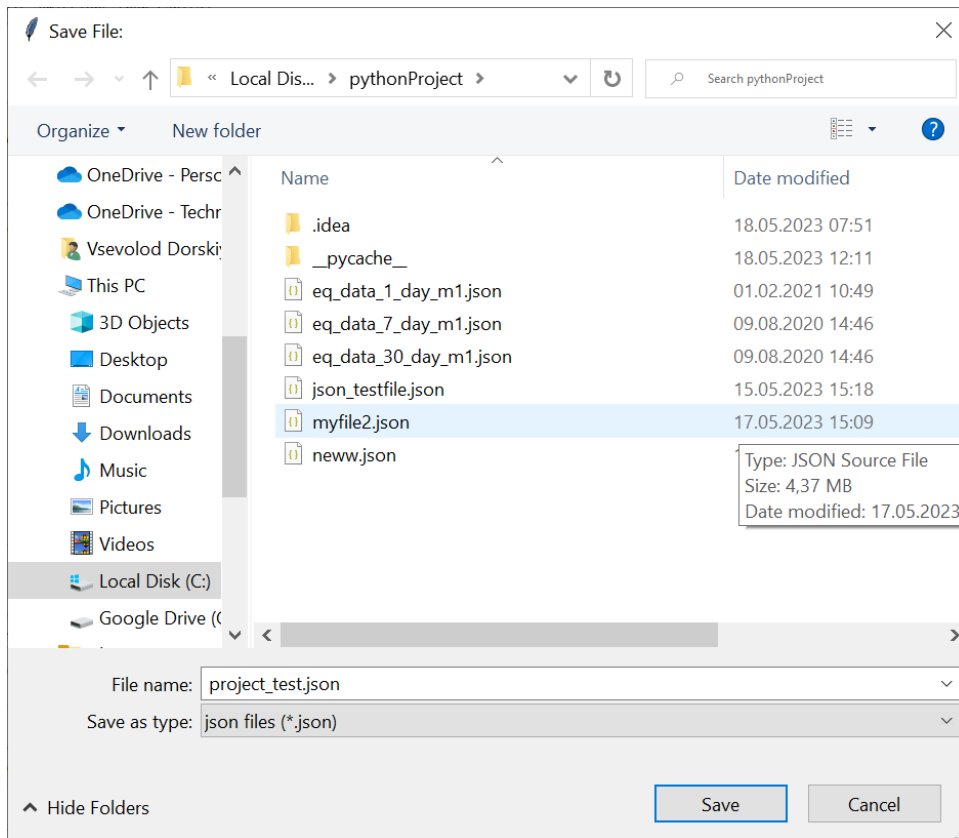
Beispielablauf:

1. Die ursprüngliche CSV-Datei ist:

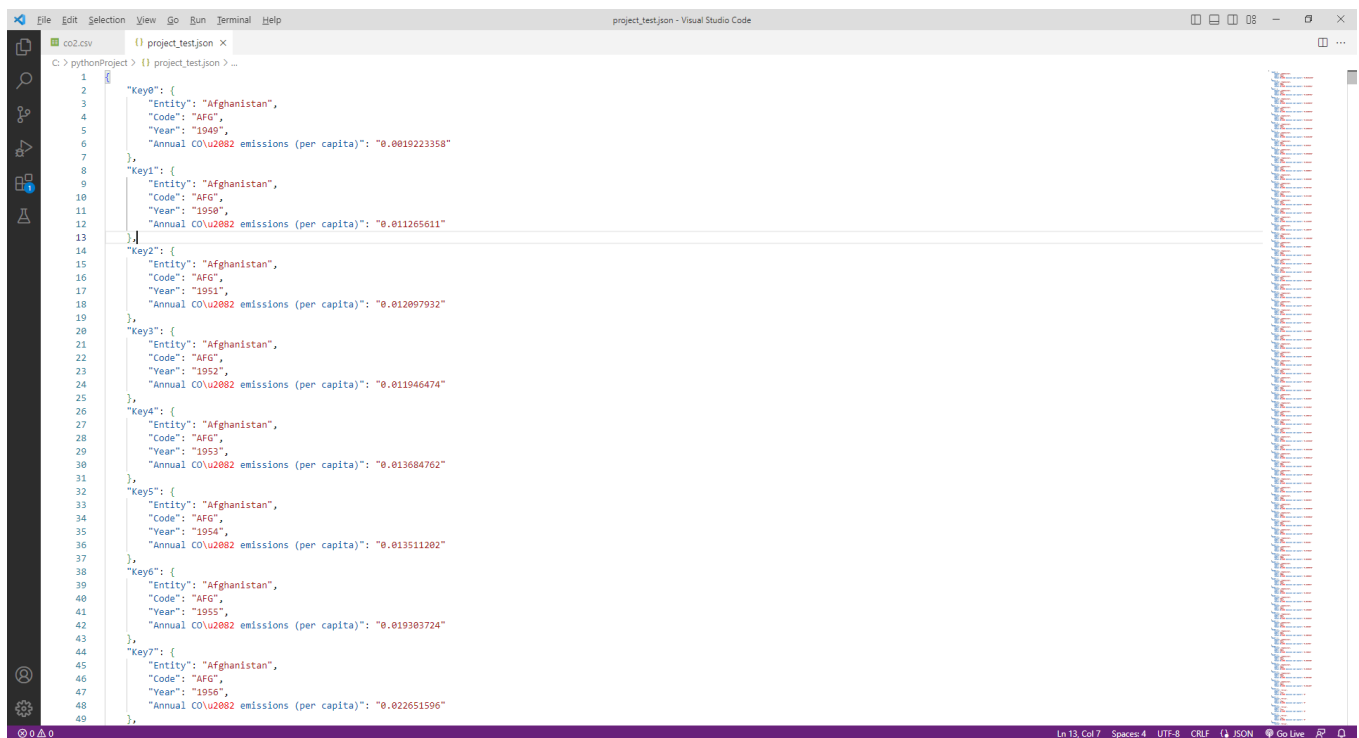


```
1 Entity,Code,Year,Annual CO2 emissions (per capita)
2 Afghanistan,AFG,1949,0.0019223358
3 Afghanistan,AFG,1950,0.011265611
4 Afghanistan,AFG,1951,0.012097032
5 Afghanistan,AFG,1952,0.011946474
6 Afghanistan,AFG,1953,0.013684762
7 Afghanistan,AFG,1954,0.013511202
8 Afghanistan,AFG,1955,0.019303724
9 Afghanistan,AFG,1956,0.022651596
10 Afghanistan,AFG,1957,0.0357019
11 Afghanistan,AFG,1958,0.039568856
12 Afghanistan,AFG,1959,0.04541344
13 Afghanistan,AFG,1960,0.04800073
14 Afghanistan,AFG,1961,0.05583596
15 Afghanistan,AFG,1962,0.07677442
16 Afghanistan,AFG,1963,0.07717596
17 Afghanistan,AFG,1964,0.08963178
18 Afghanistan,AFG,1965,0.10526929
19 Afghanistan,AFG,1966,0.11153449
20 Afghanistan,AFG,1967,0.12085797
21 Afghanistan,AFG,1968,0.119381055
22 Afghanistan,AFG,1969,0.0896882
23 Afghanistan,AFG,1970,0.1553428
24 Afghanistan,AFG,1971,0.17189354
25 Afghanistan,AFG,1972,0.13558789
26 Afghanistan,AFG,1973,0.14128836
27 Afghanistan,AFG,1974,0.16117701
28 Afghanistan,AFG,1975,0.1744933
29 Afghanistan,AFG,1976,0.15942173
30 Afghanistan,AFG,1977,0.18791811
31 Afghanistan,AFG,1978,0.1664211
32 Afghanistan,AFG,1979,0.17193846
33 Afghanistan,AFG,1980,0.14065449
34 Afghanistan,AFG,1981,0.17735797
35 Afghanistan,AFG,1982,0.20762497
36 Afghanistan,AFG,1983,0.25322488
37 Afghanistan,AFG,1984,0.2754418
38 Afghanistan,AFG,1985,0.33308113
39 Afghanistan,AFG,1986,0.2999149
40 Afghanistan,AFG,1987,0.30164644
41 Afghanistan,AFG,1988,0.27513915
42 Afghanistan,AFG,1989,0.25904718
43 Afghanistan,AFG,1990,0.18928127
44 Afghanistan,AFG,1991,0.17815459
45 Afghanistan,AFG,1992,0.122916184
46 Afghanistan,AFG,1993,0.100181666
47 Afghanistan,AFG,1994,0.094065115
48 Afghanistan,AFG,1995,0.08632284
49 Afghanistan,AFG,1996,0.080092125
```

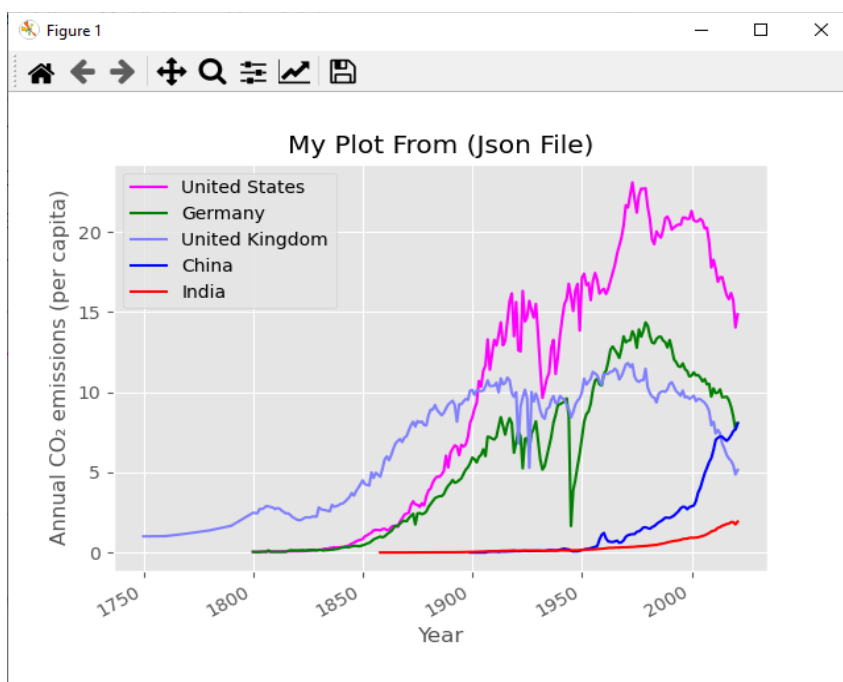
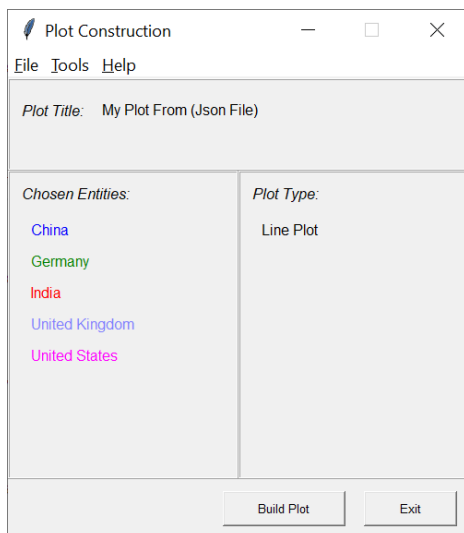
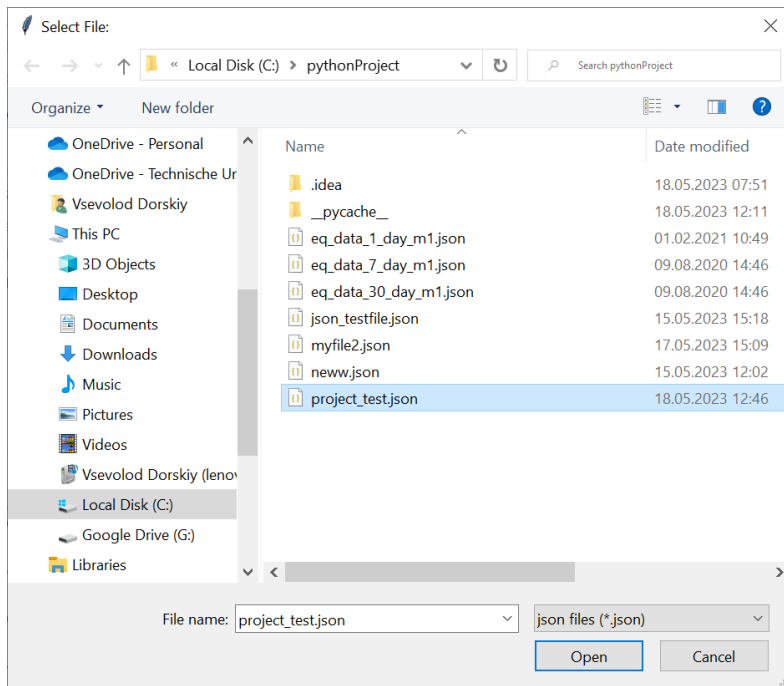
Diese CSV-Datei kann über den Dialog im JSON-Format gespeichert werden (z. B. *projekt\_test.json*):



2. Die Datei wird automatisch konvertiert:



### 3. JSON-Dateien können anschließend wie CSV-Dateien eingelesen und geplottet werden:



#### 4. Die JSON-Datei kann auf Wunsch wieder als CSV gespeichert werden (z. B. *projekt\_test.csv*):

