In [1]:
```
import numpy as np
import pandas as pd
from matplotlib import pyplot as plt
```

In [ ]:
```
```

Load and concat data

In [2]:
```
df_2023 = pd.read_excel("data/data_raw/HR2023.xlsx")
```

```
c:\Users\sebas\Documents\Data_Science\WS_24_25\Urban_technologie\project\.venv\Lib\site-packages\openpyxl\styles\stylesheet.py:237: UserWarning: Workbook contains no default style, apply openpyxl's default
  warn("Workbook contains no default style, apply openpyxl's default")
```

In [3]:
```
df_id = pd.read_csv("data/HR10y_on_id.csv", index_col="Unnamed: 0")
df_nlp = pd.read_csv("data/HR10y_on_nlp.csv", index_col="Unnamed: 0")
```

In [4]:
```
df_nlp.columns
```

Out[4]:
```
Index(['index', 'Epl.', 'Kap.', 'Tit.', 'Ist 2023', 'Zweckbestimmung', 'id',
       'id_nlp_help', 'id_nlp', '2012 id', '2012 Zweck', 'Ist 2012', '2013 id',
       '2013 Zweck', 'Ist 2013', '2014 id', '2014 Zweck', 'Ist 2014',
       '2015 id', '2015 Zweck', 'Ist 2015', '2016 id', '2016 Zweck',
       'Ist 2016', '2017 id', '2017 Zweck', 'Ist 2017', '2018 id',
       '2018 Zweck', 'Ist 2018', '2019 id', '2019 Zweck', 'Ist 2019',
       '2020 id', '2020 Zweck', 'Ist 2020', '2021 id', '2021 Zweck',
       'Ist 2021', '2022 id', '2022 Zweck', 'Ist 2022'],
      dtype='object')
```

In [5]:
```
df_nlp.rename(columns={"Ist 2023_x" : "Ist 2023"}, inplace=True)
df_nlp.drop(["id_nlp_help", "id_nlp"], axis=1, inplace=True)
```

In [6]:
```
df_nlp.columns
```

Out[6]:
```
Index(['index', 'Epl.', 'Kap.', 'Tit.', 'Ist 2023', 'Zweckbestimmung', 'id',
       '2012 id', '2012 Zweck', 'Ist 2012', '2013 id', '2013 Zweck',
       'Ist 2013', '2014 id', '2014 Zweck', 'Ist 2014', '2015 id',
       '2015 Zweck', 'Ist 2015', '2016 id', '2016 Zweck', 'Ist 2016',
       '2017 id', '2017 Zweck', 'Ist 2017', '2018 id', '2018 Zweck',
       'Ist 2018', '2019 id', '2019 Zweck', 'Ist 2019', '2020 id',
       '2020 Zweck', 'Ist 2020', '2021 id', '2021 Zweck', 'Ist 2021',
       '2022 id', '2022 Zweck', 'Ist 2022'],
      dtype='object')
```

In [7]:
```
df_10y = pd.concat([df_id, df_nlp], axis=0, ignore_index=True)
```

In [8]:
```
df_10y
```

Out[8]:

	id	Zweckbestimmung	Ist 2023	2012 Zweck	Ist 2012	20
0	232010101	Beteiligung der Länder an der Deutschen Künstl...	1085500.00	Beteiligung der Länder an der Deutschen Künstl...	1083796.31	Beteil Länd Deutsche
1	529010101	Außergewöhnlicher Aufwand aus dienstlicher Ver...	1070598.98	Außergewöhnlicher Aufwand aus dienstlicher Ver...	584973.18	Außergew Auf dienstli
2	681010101	Übernahme von Patenschaften, Ausgaben aus beso...	1341480.99	Übernahme von Patenschaften, Ausgaben aus beso...	1347209.82	Übern Pater Ausg
3	684010101	Deutsche Künstlerhilfe	3123801.79	Deutsche Künstlerhilfe	3283896.31	Kü
4	421010101	Bezüge des Bundespräsidenten	259498.56	Bezüge des Bundespräsidenten	198840.92	Be Bundespr
...	...	...	...	...	...	
2279	526020431	Sachverständige, Ausgaben für Mitglieder von F...	1451.80	Sachverständige, Ausgaben für Mitglieder von F...	86596.98	Sachve Aus Mitglied
2280	519010712	Unterhaltung der Grundstücke und baulichen Anl...	1417.77	Unterhaltung der Grundstücke und baulichen Anl...	176229.55	Unterha Grunds bauli
2281	119011016	Einnahmen aus Veröffentlichungen	1271.07	Einnahmen aus Veröffentlichungen	13368712.69	Einna Veröffent
2282	514211219	Verbrauchsmittel, Haltung von Fahrzeugen und dgl.	1178.76	Verbrauchsmittel, Haltung von Fahrzeugen und dgl.	21139.55	Verbrau Ha Fahrze
2283	119011218	Einnahmen aus Veröffentlichungen	1079.00	Einnahmen aus Veröffentlichungen	38198.87	Einna Veröffent

2284 rows × 40 columns

Analyse data set

In [9]:
```
# Check if id column is still unique == we have an unique index
df_10y.value_counts(id)
```

Out[9]:
```
140713794966280    1
1750993632144      1
1751291200112      1
1751291200336      1
1751291200400      1
                  ..
1751291202640      1
1751291201520      1
1751291201488      1
1751291200848      1
1751291200816      1
Name: count, Length: 2284, dtype: int64
```

In [10]:
```
# Check how much volumn of booking 2023 are covert
money_mapped = round(df_10y["Ist 2023"].sum() / df_2023["Ist 2023"].sum(), 3)
print(f"Persentage of Budget 23_mapped / 23_all: {money_mapped}")
print(f"Rufly {round(money_mapped*100)}% of the money is mapped")
```

```
Persentage of Budget 23_mapped / 23_all: 0.792
Rufly 79% of the money is mapped
```

In [11]:
```
df_2023["Ist 2023"].sum()
```

Out[11]:
```
np.float64(915326789706.8201)
```

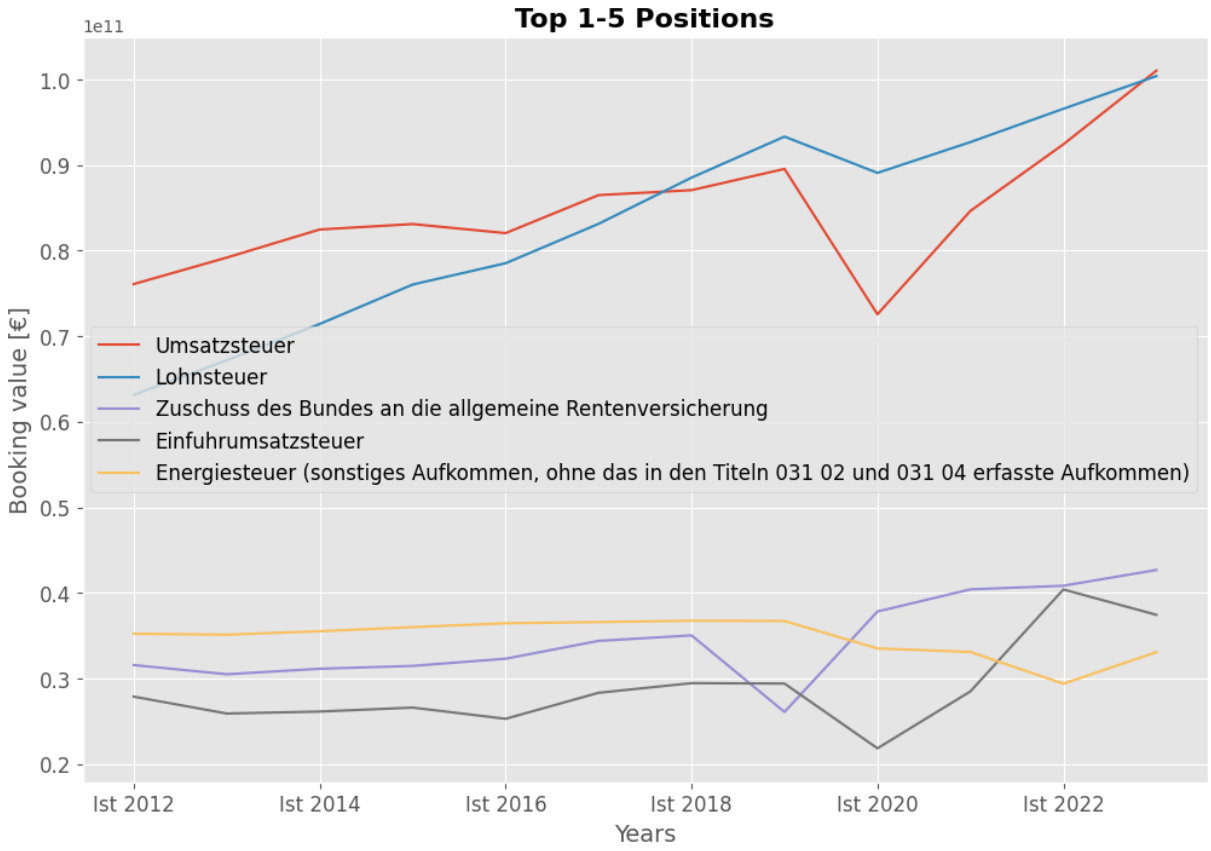
In [ ]:
```

```

Plots and visualisation

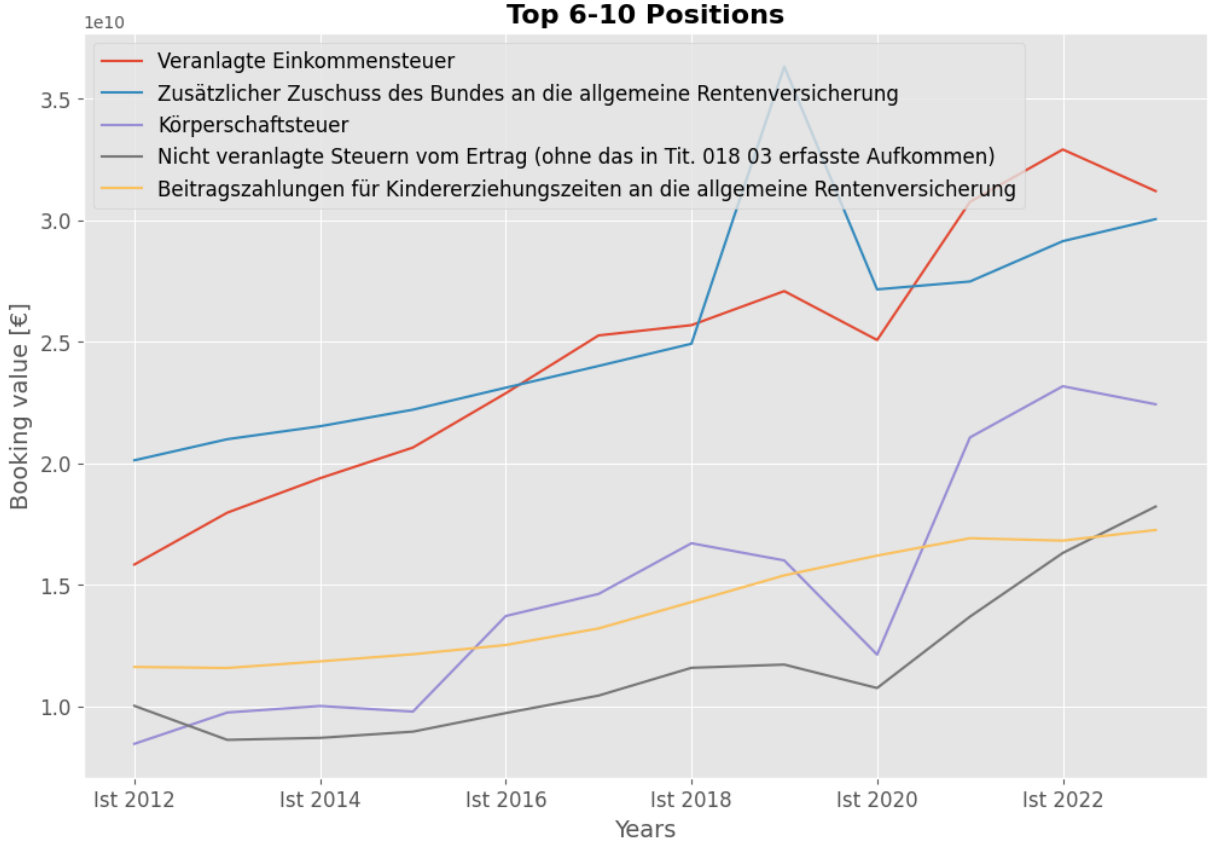
In [12]:
```
def plot_5_positions(df, position_range=(0,5)):
    df_plot = df.iloc[position_range[0]:position_range[1]].set_index("Zweckbestimmu
    # Setting a larger figure size and applying a style
    plt.figure(figsize=(12, 8))  # Adjust the width and height as needed
    plt.style.use('ggplot')  # You can choose other styles like 'ggplot', 'fivethir

    # Plotting the DataFrame
    df_plot.plot(ax=plt.gca())  # Use the current Axes to apply size and style
    plt.title(f"Top {position_range[0]+1}-{position_range[1]} Positions", fontsize=
    plt.xlabel("Years", fontsize=14)  # Customizing the x-axis label
    plt.ylabel("Booking value [€]", fontsize=14)  # Customizing the y-axis label

    # Customizing ticks and legend
    plt.xticks(fontsize=12)
    plt.yticks(fontsize=12)
    plt.legend(fontsize=12, loc="best")  # Position the legend automatically

    # Show the plot
    plt.show()
```

In [13]:
```
df_10y.sort_values("Ist 2023", ascending=False, inplace=True)
```

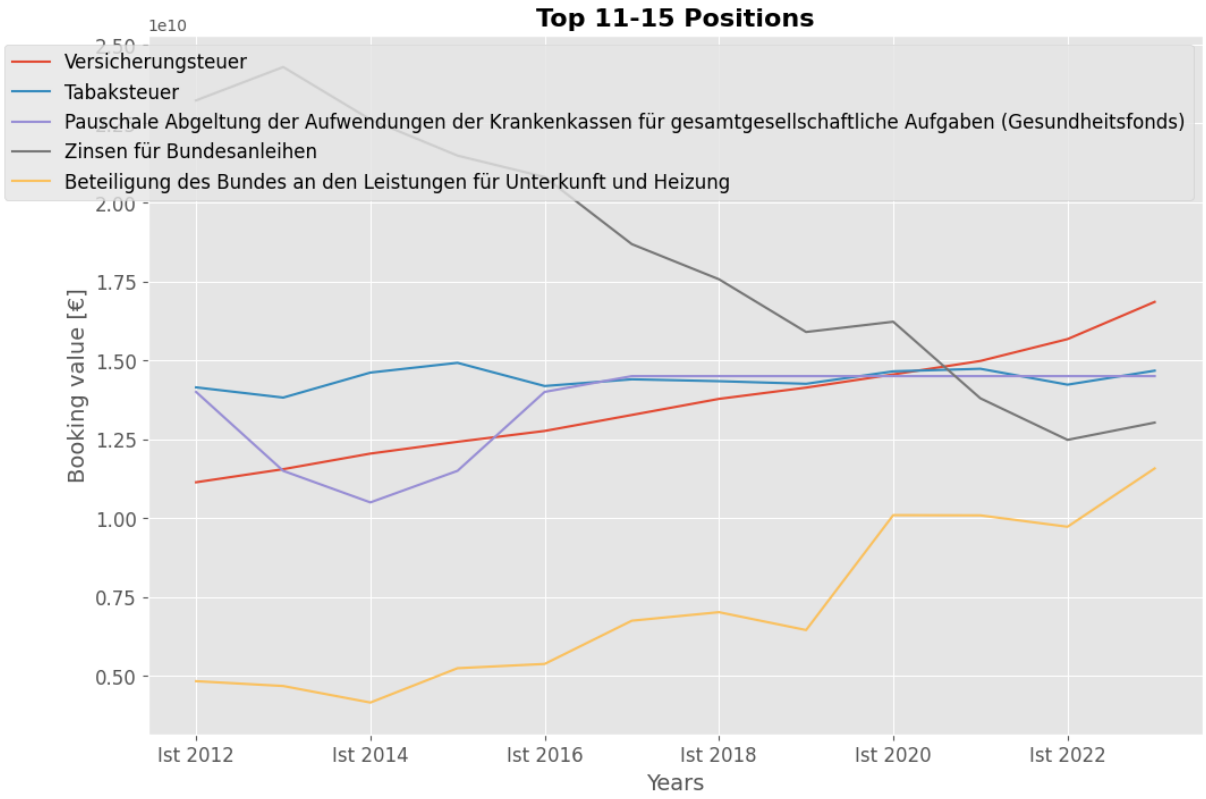
In [14]:
```
plot_5_positions(df_10y)
```



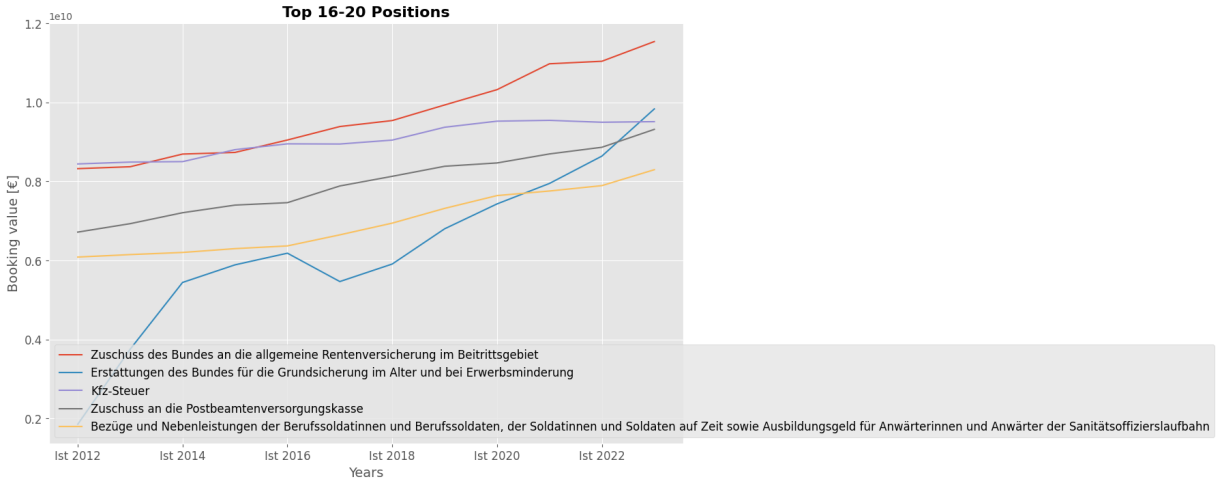
In [15]:
```
plot_5_positions(df_10y, (5,10))
```



In [16]:
```
plot_5_positions(df_10y, (10,15))
```



In [17]:
```
plot_5_positions(df_10y, (15,20))
```



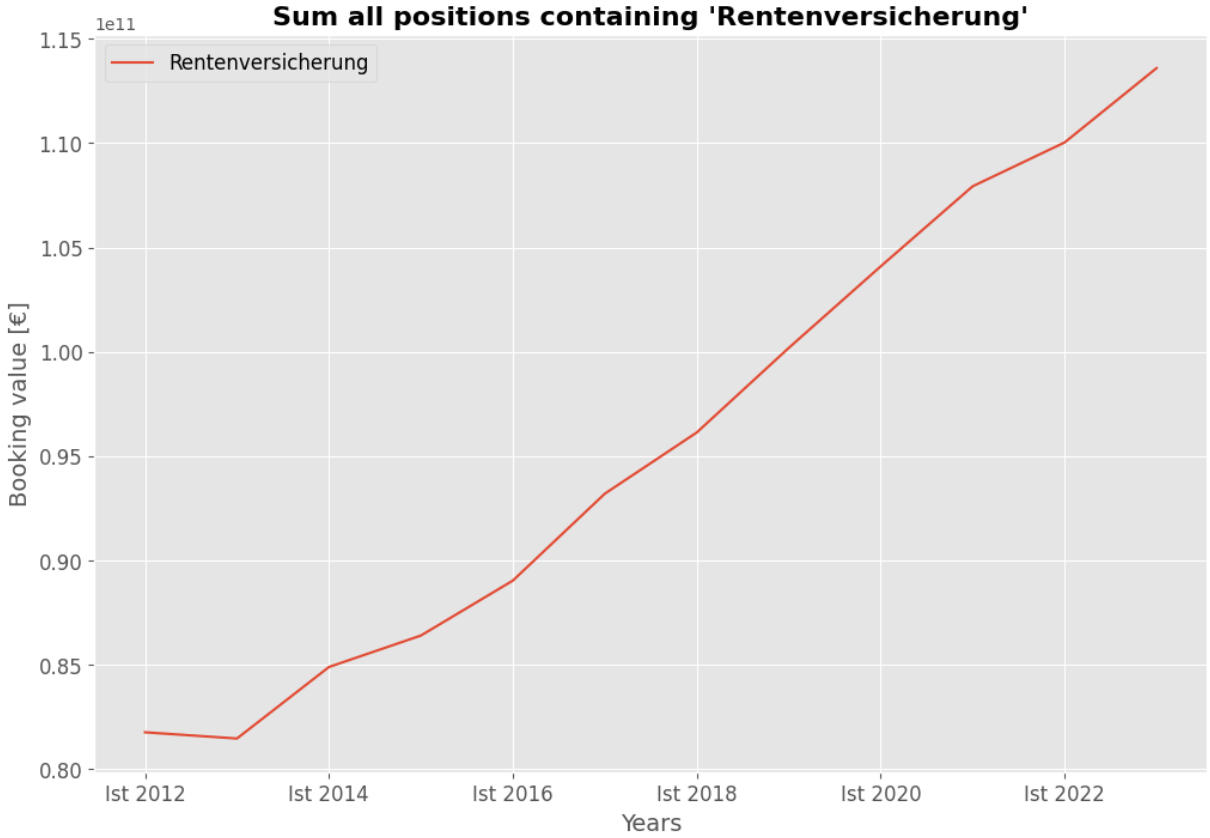
In [18]:
```
# Make sum of all positions containing
```

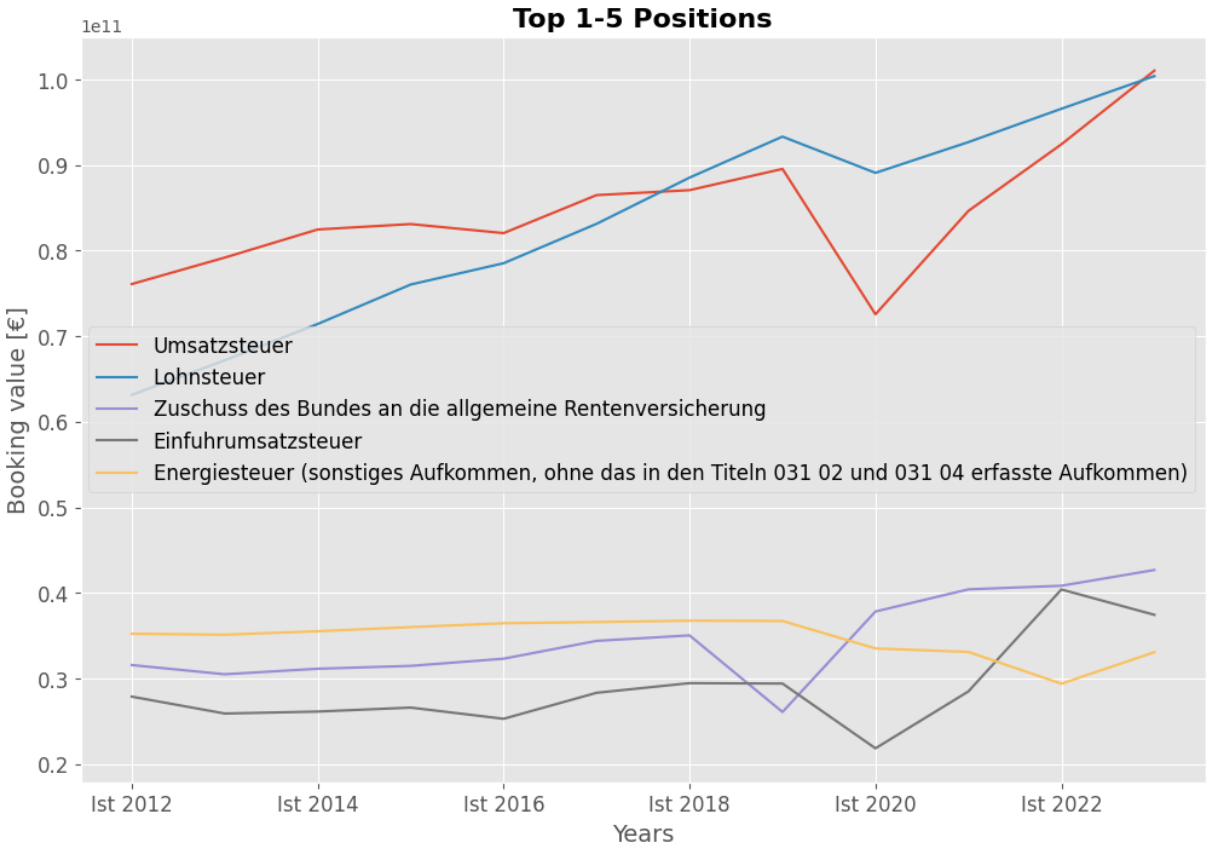
In [61]:
```
def plot_kategory(df, str_kategory):
    print(f"df_sub has {len(df[df['Zweckbestimmung'].str.contains(str_kategory)])}
    print(df[df["Zweckbestimmung"].str.contains(str_kategory)]["Zweckbestimmung"].h
    df_plot = df[df["Zweckbestimmung"].str.contains(str_kategory)][[f"Ist 20{year}"

    # Setting a larger figure size and applying a style
    plt.figure(figsize=(12, 8))  # Adjust the width and height as needed
    plt.style.use('ggplot')  # You can choose other styles like 'ggplot', 'fivethir

    # Plotting the DataFrame
    df_plot.plot(ax=plt.gca())  # Use the current Axes to apply size and style
    plt.title(f"Sum all positions containing \'{str_kategory}\'", fontsize=16, font
```

```
    plt.xlabel("Years", fontsize=14)  # Customizing the x-axis label
    plt.ylabel("Booking value [€]", fontsize=14)  # Customizing the y-axis label

    # Customizing ticks and legend
    plt.xticks(fontsize=12)
    plt.yticks(fontsize=12)
    plt.legend(fontsize=12, loc="best")  # Position the legend automatically

    # Show the plot
    plt.show()
```

In [62]:
```
plot_kategory(df_10y, "Rentenversicherung")
plot_5_positions(df_10y)
```

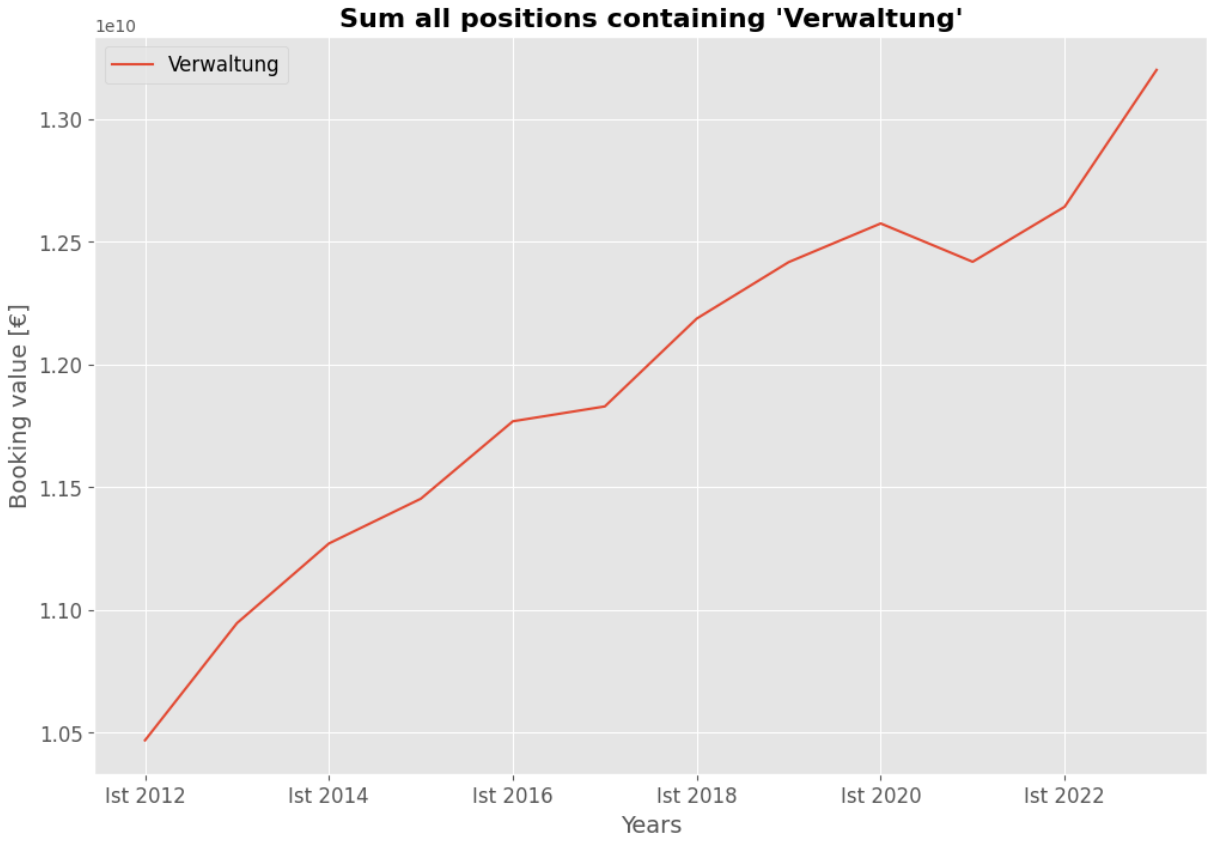
```
df_sub has 15 rows containing str: Rentenversicherung
903    Zuschuss des Bundes an die allgemeine Rentenve...
904    Zusätzlicher Zuschuss des Bundes an die allgem...
905    Beitragszahlungen für Kindererziehungszeiten a...
908    Zuschuss des Bundes an die allgemeine Rentenve...
913    Beteiligung des Bundes in der knappschaftliche...
Name: Zweckbestimmung, dtype: object
```





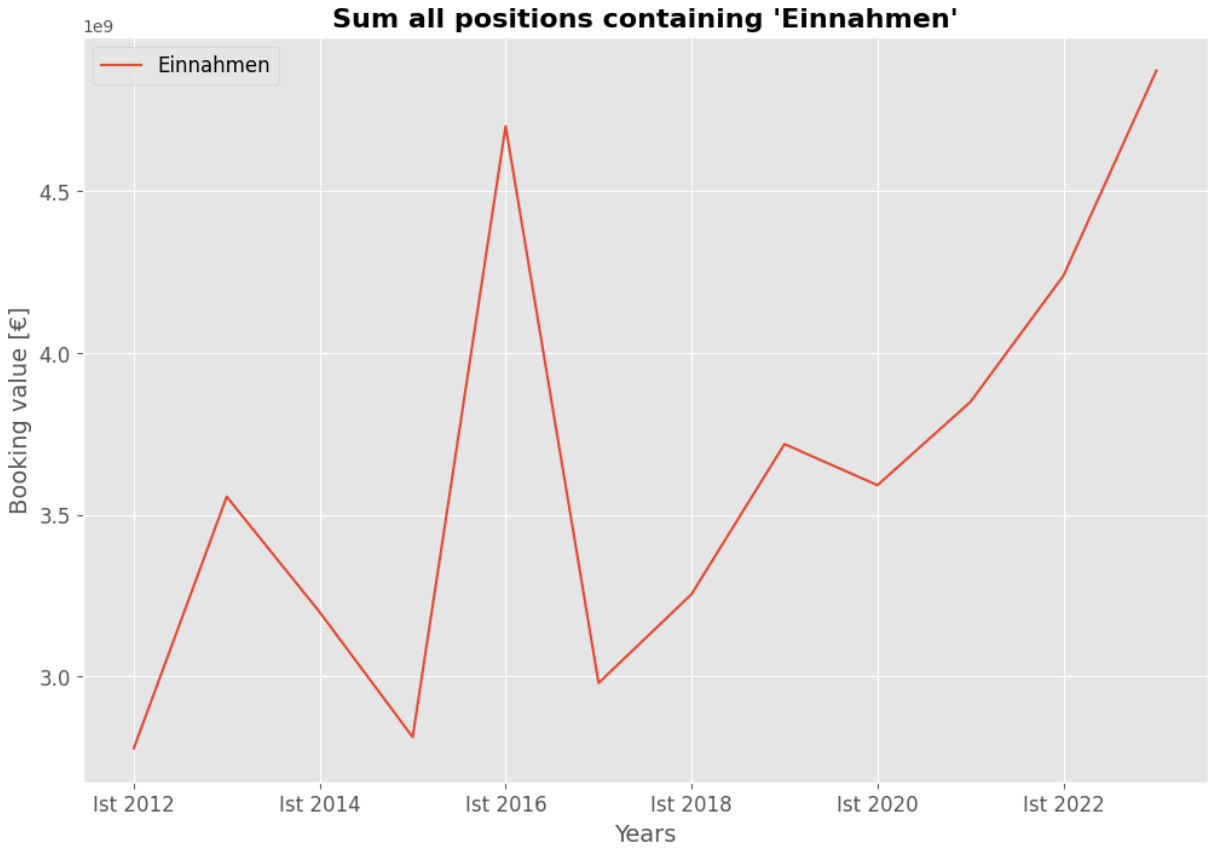
In [63]:
```
plot_kategory(df_10y, "Verwaltung")
```

```
df_sub has 169 rows containing str: Verwaltung
910    Verwaltungskosten für die Durchführung der Gru...
912    Erstattungen von Verwaltungsausgaben des Bunde...
950    Erstattung von Verwaltungskosten an die Bundes...
984    Zuwendungen an die Lausitzer und Mitteldeutsch...
992    Erstattung der Verwaltungskosten an die Bundes...
Name: Zweckbestimmung, dtype: object
```



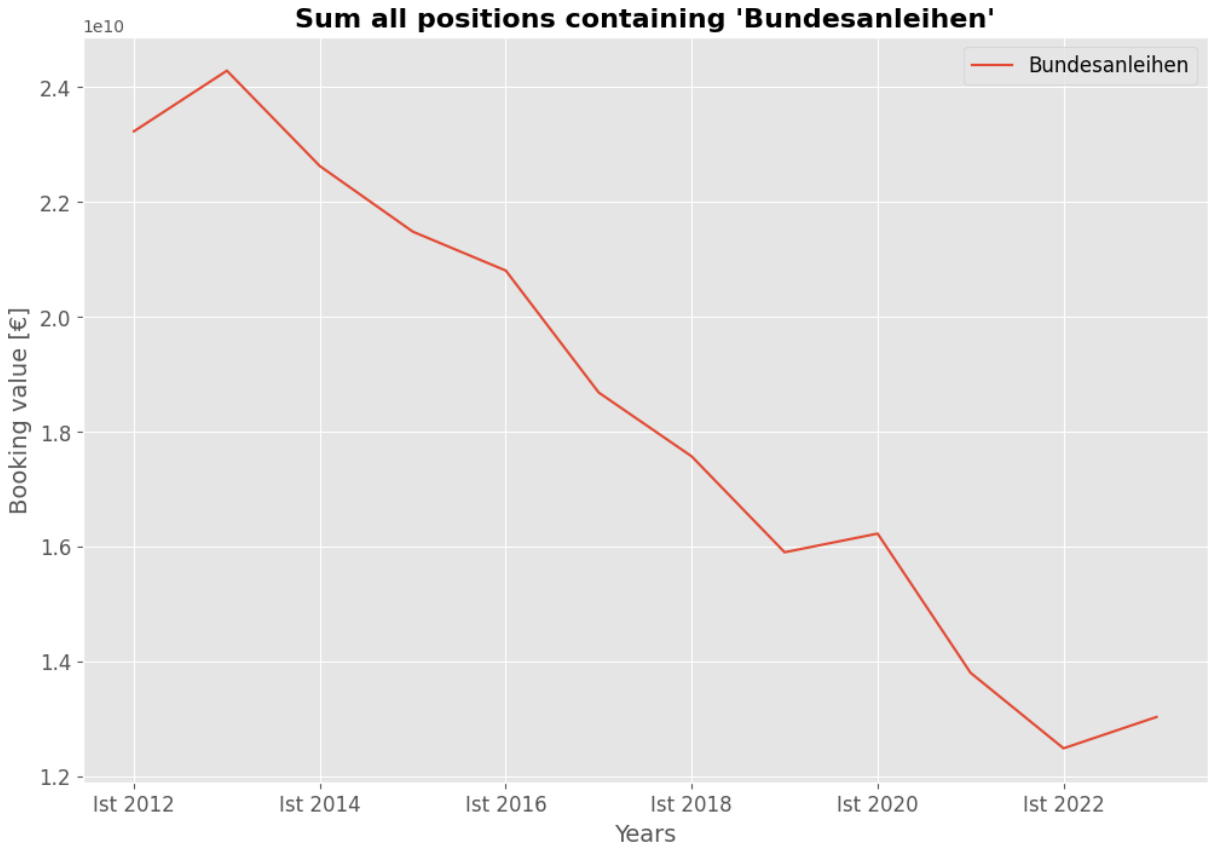
In [64]:
```
plot_kategory(df_10y, "Einnahmen")
```

```
df_sub has 152 rows containing str: Einnahmen
800    Entgelte und sonstige Einnahmen aus Gewährleis...
832                                Vermischte Einnahmen
940    Einnahmen aus Zuschüssen des Europäischen Sozi...
698                                Vermischte Einnahmen
944    Einnahmen für die Endlagerung radioaktiver Abf...
Name: Zweckbestimmung, dtype: object
```



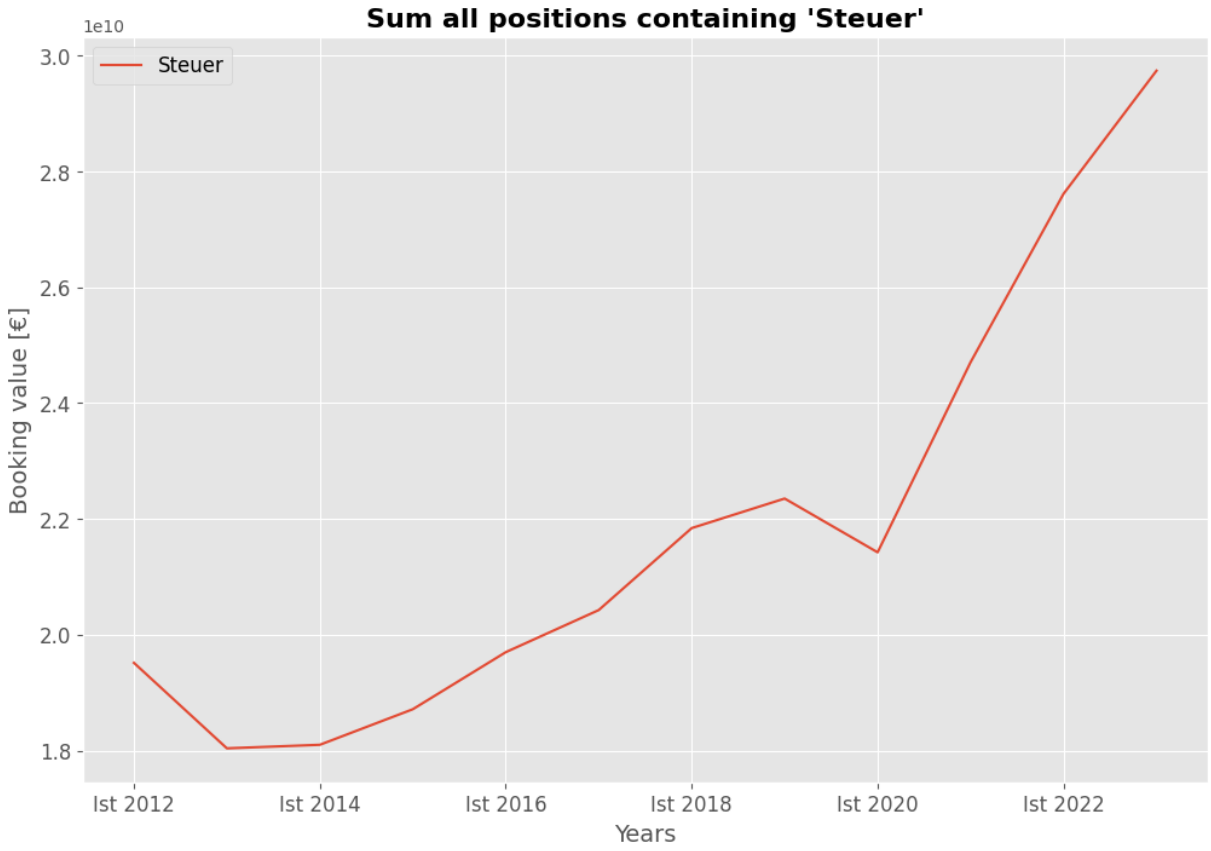
In [65]:
```
plot_kategory(df_10y, "Bundesanleihen")
```

```
df_sub has 1 rows containing str: Bundesanleihen
797    Zinsen für Bundesanleihen
Name: Zweckbestimmung, dtype: object
```



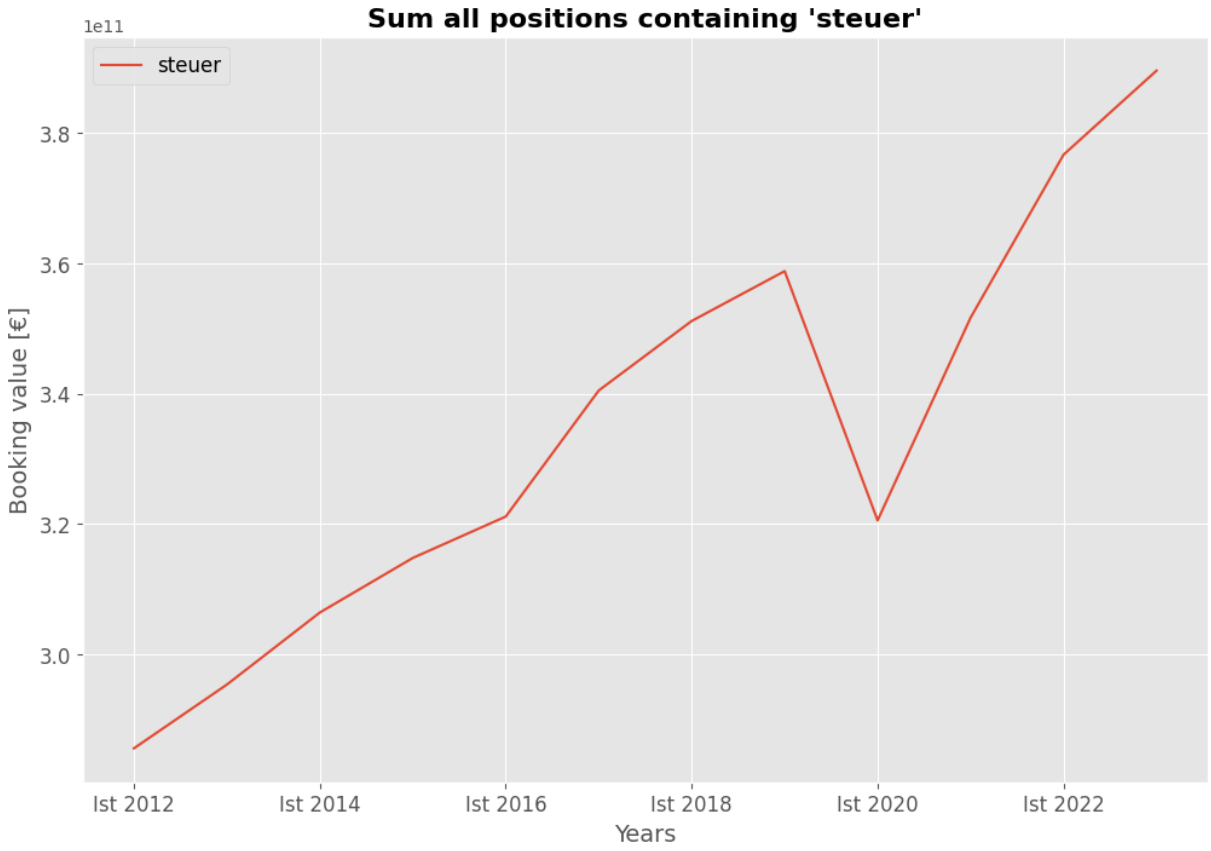
In [68]:
```
plot_kategory(df_10y, "Steuer")
```

```
df_sub has 6 rows containing str: Steuer
806     Nicht veranlagte Steuern vom Ertrag (ohne das ...
822                                             Kfz-Steuer
826     Solidaritätszuschlag zu den nicht veranlagten ...
1195    Bestandserfassung der Bundesfernstraßen, Koord...
37      Ausgaben für die Gemeinsame IT des Bundes, IT-...
Name: Zweckbestimmung, dtype: object
```



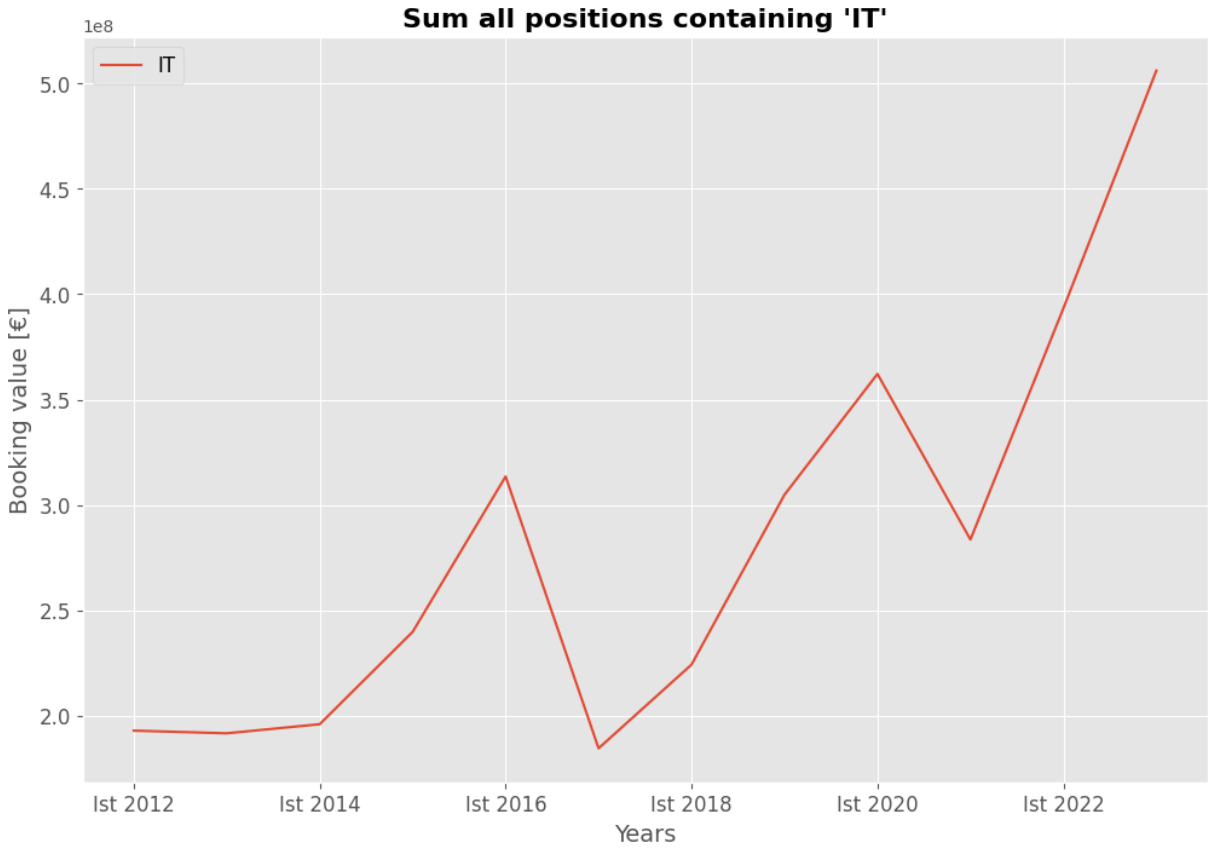
In [69]:
```
plot_kategory(df_10y, "steuer")
```

```
df_sub has 24 rows containing str: steuer
808                                          Umsatzsteuer
804                                            Lohnsteuer
809                                     Einfuhrumsatzsteuer
813    Energiesteuer (sonstiges Aufkommen, ohne das i...
805                            Veranlagte Einkommensteuer
Name: Zweckbestimmung, dtype: object
```



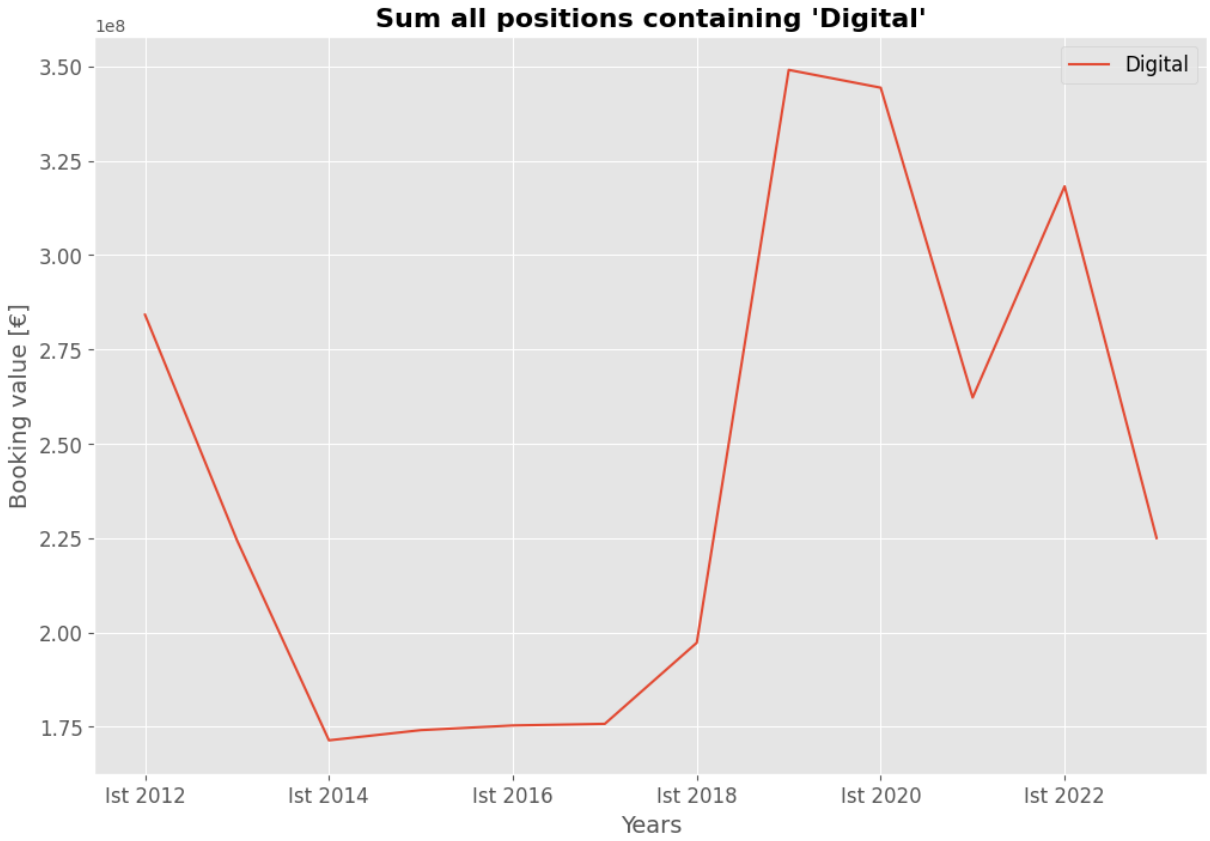
In [70]:
```
plot_kategory(df_10y, "IT")
```

```
df_sub has 47 rows containing str: IT
772                 Kommunikationssysteme, IT-Sicherheit
217    Erwerb von Geräten, Ausstattungs- und Ausrüstu...
171    Erwerb von Geräten, Ausstattungs- und Ausrüstu...
246    Behördenspezifische fachbezogene Verwaltungsau...
106    Erwerb von Geräten, Ausstattungs- und Ausrüstu...
Name: Zweckbestimmung, dtype: object
```



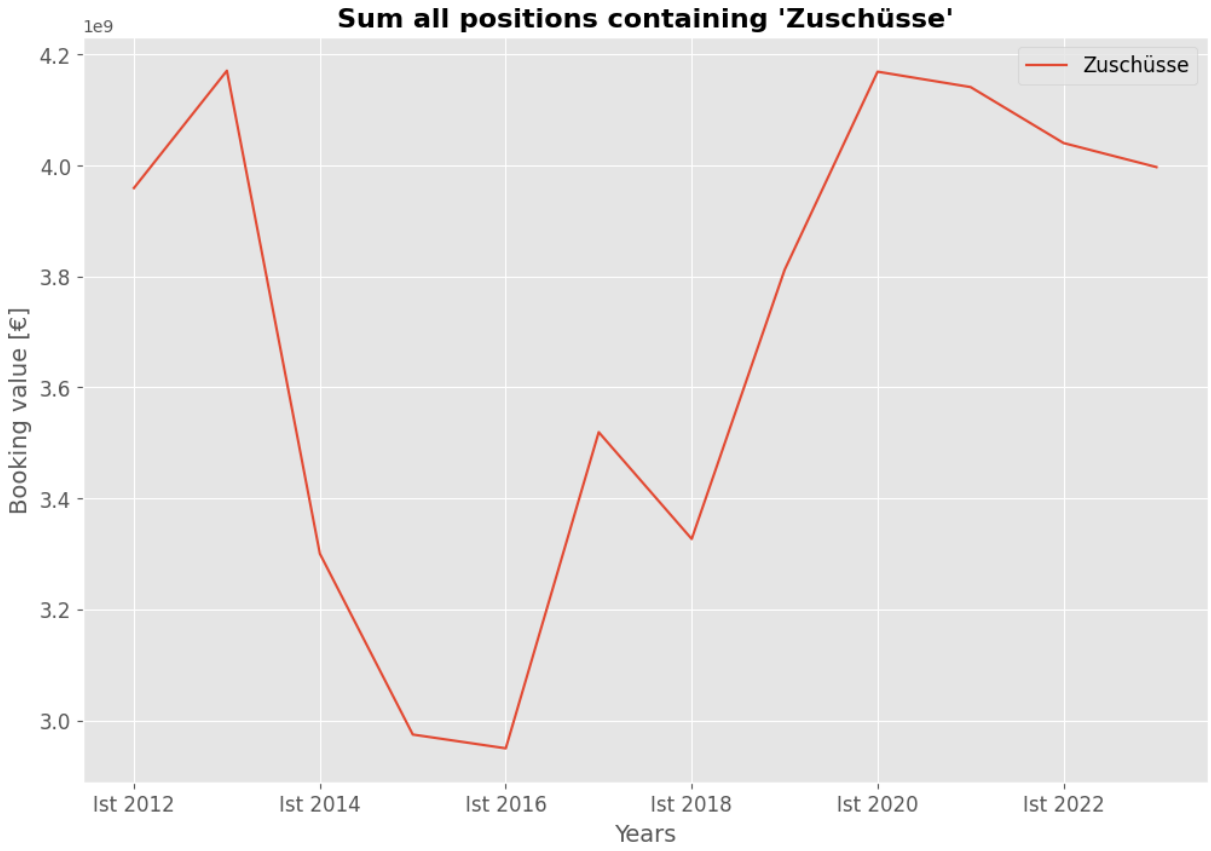
In [ ]:
```
plot_kategory(df_10y, "Digital")
```

```
df_sub has 4 rows containing str: Digital
1012    Zuschüsse an die Bundesanstalt für den Digital...
766     Analysen, Planung und Datenerhebung für Grunds...
1042    Zuschüsse an die Bundesanstalt für den Digital...
36                 Digitale Gesellschaft und Datenpolitik
Name: Zweckbestimmung, dtype: object
```



In [73]:
```
plot_kategory(df_10y, "Zuschüsse")
```

```
df_sub has 74 rows containing str: Zuschüsse
922    Zuschüsse zu den Beiträgen zur Rentenversicher...
940    Einnahmen aus Zuschüssen des Europäischen Sozi...
728                  Zuschüsse an Begabtenförderungswerke
964    Zuschüsse zur Förderung von Umwelt und Sicherh...
968    Verwendung von Zuschüssen des Europäischen Soz...
Name: Zweckbestimmung, dtype: object
```



In [ ]:

Make own dataset

In [53]:
```
df_all = pd.read_csv("data/mapped_on_nlp/HR2023_nlp_reference_year.csv", index_col=
```

In [54]:
```
df_all.drop(columns=["index", "id_nlp_help", "id_nlp"], axis=1, inplace=True)
```

In [55]:
```
for year in range(12,16):
    df_i = pd.read_csv(f"data/mapped_on_nlp/HR20{year}_nlp_mapped_to_HR2023.csv", i
    df_i.drop(columns=["index","Epl.","Kap.","Tit.", "Zweckbestimmung", "id_nlp_hel
    df_all = pd.merge(df_all, df_i, on='id', how="inner")

# Check if id column is still unique == we have an unique index
df_all.value_counts(id)[:3]
```

Out[55]:
```
140720861188872    1
1884871576016      1
1884877989712      1
Name: count, dtype: int64
```

In [56]:
```
df_all.sort_values("Ist 2023", ascending=False).head(15)
```

Out[56]:

	Epl.	Kap.	Tit.	Ist 2023	Zweckbestimmung	id	Ist 2012	20
0	11	2	63681	4.267868e+10	Zuschuss des Bundes an die allgemeine Rentenve...	636811102	3.156989e+10	6368
1	11	2	63683	3.003697e+10	Zusätzlicher Zuschuss des Bundes an die allgem...	636831102	2.012281e+10	6368
2	11	1	68112	2.580758e+10	Arbeitslosengeld II	681121101	1.895134e+10	6811
3	11	2	63684	1.725755e+10	Beitragszahlungen für Kindererziehungszeiten a...	636841102	1.162751e+10	6368
4	15	1	63606	1.450000e+10	Pauschale Abgeltung der Aufwendungen der Krank...	636061501	1.400000e+10	6360
5	11	1	63211	1.157631e+10	Beteiligung des Bundes an den Leistungen für U...	632111101	4.838414e+09	6321
6	11	2	63682	1.154278e+10	Zuschuss des Bundes an die allgemeine Rentenve...	636821102	8.323487e+09	6368
7	11	2	63201	9.835967e+09	Erstattungen des Bundes für die Grundsicherung...	632011102	1.850003e+09	6320
8	11	1	63613	6.318378e+09	Verwaltungskosten für die Durchführung der Gru...	636131101	4.209093e+09	6361
9	12	2	89111	5.364604e+09	Baukostenzuschüsse für einen Infrastrukturbeit...	891111202	2.500000e+09	8911
10	12	16	63401	5.218166e+09	Erstattungen von Verwaltungsausgaben des Bunde...	634011216	5.104600e+09	6340
11	11	2	63616	4.948454e+09	Beteiligung des Bundes in der knappschaftliche...	636161102	5.546283e+09	6362
12	11	1	68511	3.814147e+09	Leistungen zur Eingliederung in Arbeit	685111101	3.751175e+09	6851
13	11	2	63612	3.542804e+09	Erstattung von Aufwendungen der Deutschen Rent...	636121102	2.908909e+09	6362
14	14	6	55311	2.952729e+09	Erhaltung von Flugzeugen, Flugkörpern, Flugzeu...	553111406	1.234554e+09	5531

In [57]:
```
df_all[df_all["Zweckbestimmung"] == "Arbeitslosengeld II"]
```

Out[57]:

	Epl.	Kap.	Tit.	Ist 2023	Zweckbestimmung	id	Ist 2012	2012 i
2	11	1	68112	2.580758e+10	Arbeitslosengeld II	681121101	1.895134e+10	68112111

In [ ]: