In [51]: ▶|

```python
import pandas as pd

file_path = 'C:/Users/Abhinav/Desktop/Data Analytics/US_Students_Records.xlsx'

# Load the XLSX file
record = pd.read_excel(file_path)

# Display the first few rows to verify the data was loaded
df.head()
```

Out[51]:

	Year	African_Sub_Saharan	East_Africa	Central_Africa	Southern_Africa	West_Africa	Asia	East_Asia	South_and_Central
0	1949/50	537	37	0	0	361	6806	4160	
1	1954/55	862	99	9	9	568	10175	5423	
2	1959/60	1346	457	11	11	639	17808	9268	
3	1964/65	5259	1860	209	209	2318	30640	16222	
4	1969/70	6064	1731	264	264	3133	51033	27589	
5	1974/75	16040	3949	400	400	10600	58460	30720	
6	1979/80	29600	5093	1060	1060	20760	81730	45710	
7	1984/85	33720	6548	1335	1335	22440	143680	72630	
8	1989/90	20370	6446	1766	1766	7950	208110	127320	
9	1994/95	17642	5915	1294	1294	5943	261789	168190	
10	1999/00	26124	9497	1579	1579	9177	280149	180147	
11	2000/01	29399	10651	1511	1511	10346	302058	189371	
12	2001/02	32509	11841	1612	1612	11385	324812	196813	
13	2002/03	35406	12478	1939	1939	13590	332298	199666	
14	2003/04	33942	11418	1889	1889	13821	324006	189874	
15	2004/05	32492	10649	2074	2074	13782	325112	192561	
16	2005/06	32847	10617	2410	2410	13846	327785	197576	
17	2006/07	32423	10514	2815	2815	13344	344495	204023	
18	2007/08	32020	9909	2902	2902	13632	380465	223306	
19	2008/09	32708	10411	2819	2819	13837	415000	240952	
20	2009/10	32121	9872	2872	2872	14064	435667	261125	
21	2010/11	31470	8863	2831	2831	14446	461818	286925	
22	2011/12	30046	7827	2778	2778	14245	489970	319515	
23	2012/13	30585	7761	2861	2861	14452	525849	357596	
24	2013/14	31113	7549	2883	2883	14998	568510	393205	
25	2014/15	33593	7560	3032	3032	16958	627306	417881	
26	2015/16	35364	7690	3311	3311	18100	689525	439702	

	Year	African_Sub_Saharan	East_Africa	Central_Africa	Southern_Africa	West_Africa	Asia	East_Asia	South_and_Central
27	2016/17	37735	8480	3545	3545	19237	734309	459284	2
28	2017/18	39479	9093	3562	3562	20395	758076	468304	2
29	2018/19	40290	9227	3325	3325	21423	768260	472085	2
30	2019/20	41697	9989	3071	3071	22151	758014	472339	2
31	2020/21	39061	9319	2899	2899	20808	645622	395808	2
32	2021/22	42518	10157	2871	2871	22974	658669	372378	2

In [82]:

```python
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# Create a line plot to show the trend over the years.
plt.figure(figsize=(12, 6))
ax = sns.lineplot(data=record, x='Year', y='Total', marker='o', markersize=5, color='b', label='Total Stud
plt.title('Trend of Total International Students from All Continents Going to the USA')
plt.xlabel('Year')
plt.ylabel('Total Number of Students')

# Set x-axis ticks at 5-year intervals (adjust based on your dataset)
xticks = range(0, len(record['Year']), 5)
xlabels = record['Year'].iloc[xticks].apply(lambda x: x.replace('/', '-'))
ax.set_xticks(xticks)
ax.set_xticklabels(xlabels, rotation=45)

plt.legend(loc='upper left')
plt.grid(True)
plt.show()
```
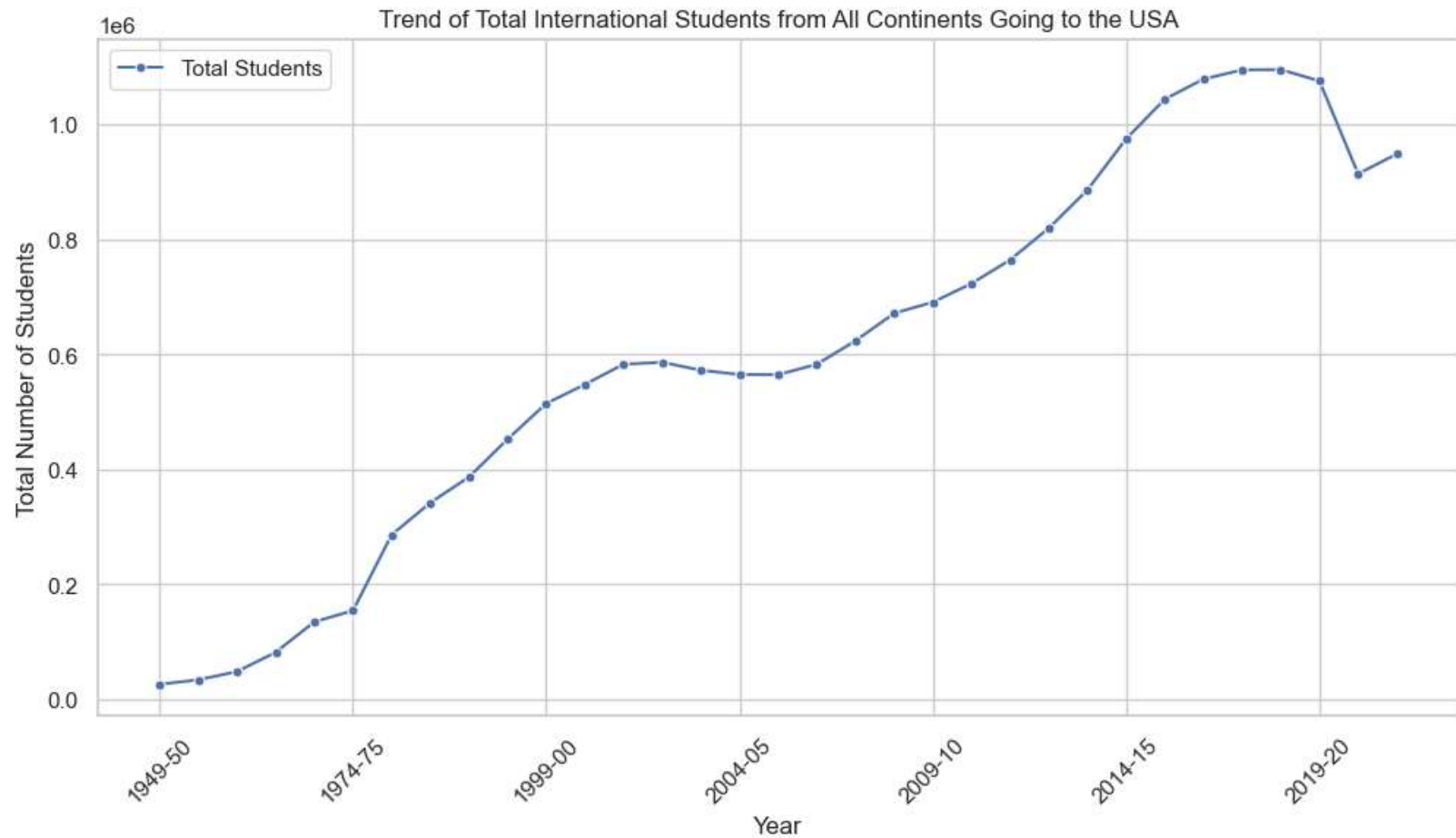
Trend of Total International Students from All Continents Going to the USA

In [93]:

```python
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# Assuming your dataset is already loaded and named as 'record'.

# Create a line plot for the selected continents.
selected_continents = [
    'Asia', 'East_Asia', 'South_and_Central_Asia', 'Southeast_Asia'
]

# Define the years for custom x-axis labels.
custom_years = ['1949-50', '1974-75', '1999-00', '2004-05', '2009-10', '2014-15', '2019-20']

# Create a mapping from custom years to the closest available data points.
custom_years_to_data_index = {
    '1949-50': 0,
    '1974-75': 5,
    '1999-00': 10,
    '2004-05': 15,
    '2009-10': 20,
    '2014-15': 25,
    '2019-20': 30
}

plt.figure(figsize=(12, 6))
for column in selected_continents:
    sns.lineplot(data=record, x='Year', y=column, label=column)

plt.title('Number of Students from Selected Asian Regions Going to the USA')
plt.xlabel('Year')
plt.ylabel('Number of Students')
plt.legend(loc='upper left')
plt.grid(True)

# Expand the vertical axis.
plt.ylim(bottom=0)

# Set custom x-axis ticks and add labels at the end of each line.
plt.xticks([custom_years_to_data_index[year] for year in custom_years], custom_years, rotation=45)

for column in selected_continents:
    x_pos = custom_years_to_data_index['2019-20']
```
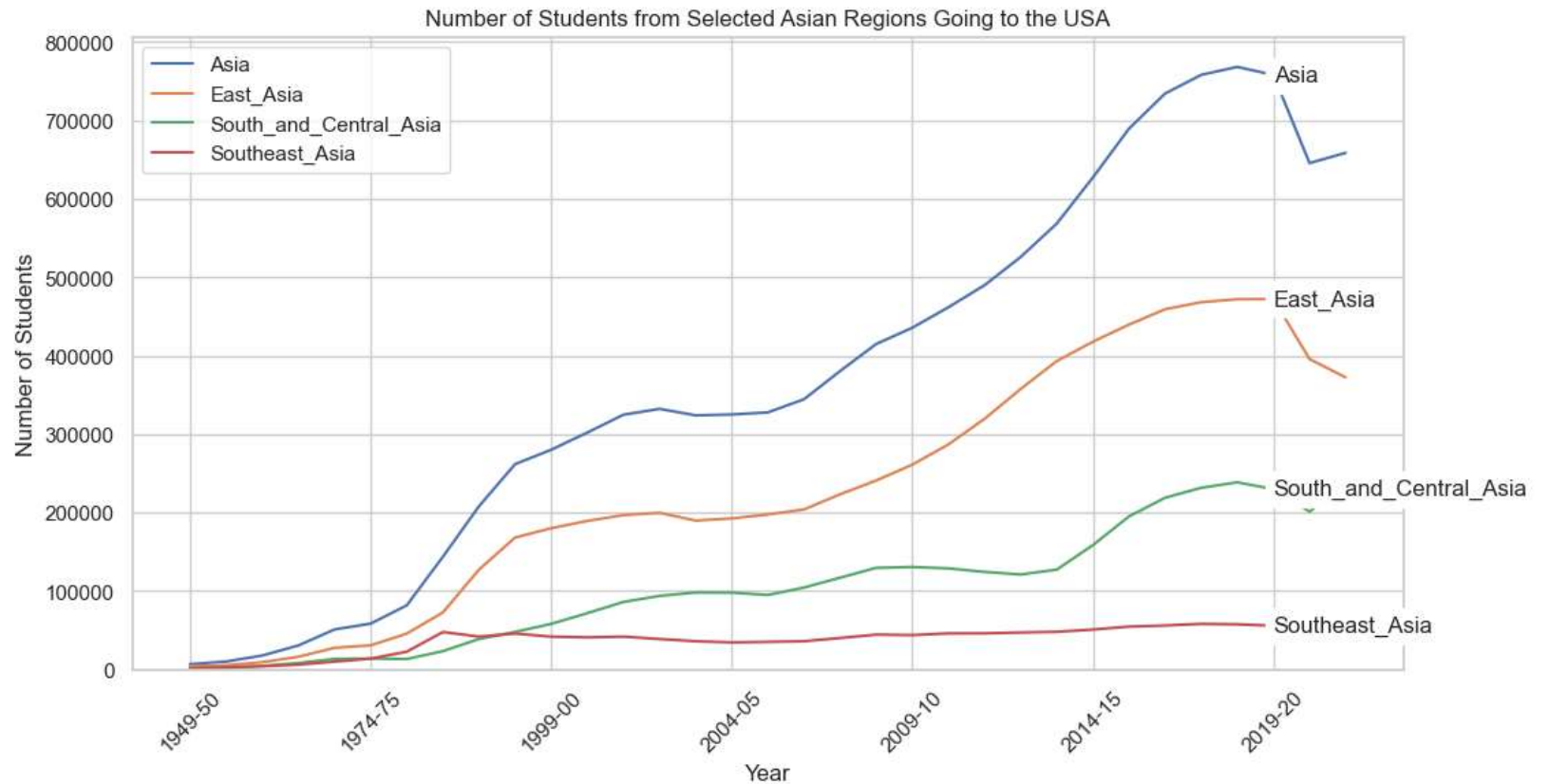
```
    y_pos = record[column].iloc[x_pos]
    plt.text(x_pos, y_pos, column, va='center', ha='left', backgroundcolor='w')

plt.show()
```


Number of Students from Selected Asian Regions Going to the USA

In [102]:

```python
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# Assuming your dataset is already loaded and named as 'record'.

# Set a Seaborn style with a dark grey background
sns.set(style="darkgrid")

# Create a color palette
colors = sns.color_palette("husl")

# Create a line plot for the selected continents.
selected_continents = [
    'Asia', 'East_Asia', 'South_and_Central_Asia', 'Southeast_Asia'
]

# Define the years for custom x-axis labels.
custom_years = ['1949-50', '1974-75', '1999-00', '2004-05', '2009-10', '2014-15', '2019-20']

# Create a mapping from custom years to the closest available data points.
custom_years_to_data_index = {
    '1949-50': 0,
    '1974-75': 5,
    '1999-00': 10,
    '2004-05': 15,
    '2009-10': 20,
    '2014-15': 25,
    '2019-20': 30
}

# Create a dark grey background
plt.figure(figsize=(12, 6))
ax = plt.gca()
ax.set_facecolor('#333333')  # Dark grey background

for i, column in enumerate(selected_continents):
    sns.lineplot(data=record, x='Year', y=column, label=column, color=colors[i], linestyle='-', linewidth=

plt.title('Number of Students from Selected Asian Regions Going to the USA')
plt.xlabel('Year')
plt.ylabel('Number of Students')
plt.legend(loc='upper left')
```
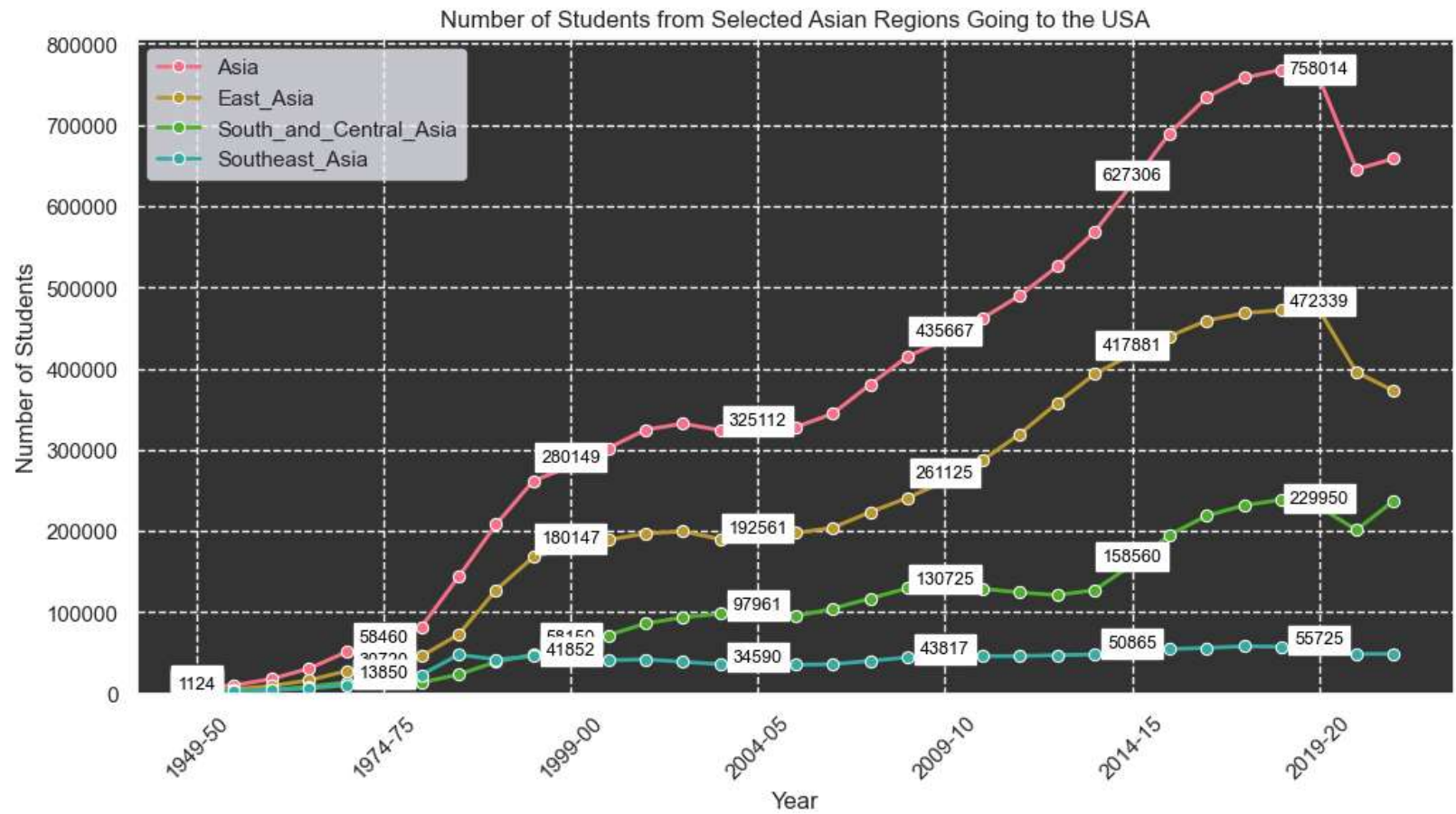
```python
plt.grid(True, color='white', linestyle='--')

# Expand the vertical axis.
plt.ylim(bottom=0)

# Set custom x-axis ticks and add labels at the end of each line.
plt.xticks([custom_years_to_data_index[year] for year in custom_years], custom_years, rotation=45)

# Add values to the x-axis labels
for year in custom_years:
    x_pos = custom_years_to_data_index[year]
    for column in selected_continents:
        y_pos = record[column].iloc[x_pos]
        plt.text(x_pos, y_pos, f'{y_pos}', va='bottom', ha='center', backgroundcolor='w', color='black', f

plt.show()
```

Number of Students from Selected Asian Regions Going to the USA

In [ ]: