```
In [1]:    import pandas_datareader as pdr
           import pandas as pd
           import datetime
           import plotly.express as px
```

```
In [2]:    %%capture
           # Start and End in yyyy, mm, dd format
           start = datetime.datetime (1971, 1, 1)
           end = datetime.datetime (2020, 12, 31)

           ### GET MARKET CAP DATA ###
           data_series = "WILL5000PRFC"
           market_cap = pdr.DataReader(data_series, 'fred', start, end)

           # Drop NaN values from market_cap dataframe
           market_cap = market_cap.dropna()

           # Conver dataframe index to datetime object
           market_cap.index = pd.to_datetime(market_cap.index)

           ### GET GDP DATA ###
           data_series = "GDP"
           gdp = pdr.DataReader(data_series, 'fred', start, end)
           # Drop NaN values from gdp dataframe
           gdp = gdp.dropna()

           # Conver dataframe index to datetime object
           gdp.index = pd.to_datetime(gdp.index)

           # Create quarterly market_cap
           market_cap_quarterly = pd.DataFrame()

           market_cap_quarterly = market_cap
           market_cap_quarterly.resample("3M").last()

           # Merge the two dataframes using pandas' merge() on their indices
           market_cap_to_gdp = pd.merge(market_cap_quarterly, gdp, left_index=True, right_index=True)

           # Create Date column so we can use it as the lable in chart
           market_cap_to_gdp["Date"] = market_cap_to_gdp.index

           # Multiply the Wilshire 5000 Full Cap index by $1.19 billion to set it to 1980's USD per Wilshire's notes.
           market_cap_to_gdp["WILL5000PRFC"] = market_cap_to_gdp["WILL5000PRFC"] * 1190000000

           # Calculate the indicator as market cap / GDP
           market_cap_to_gdp["Buffett Indicator"] = market_cap_to_gdp["WILL5000PRFC"] / market_cap_to_gdp["GDP"]
```
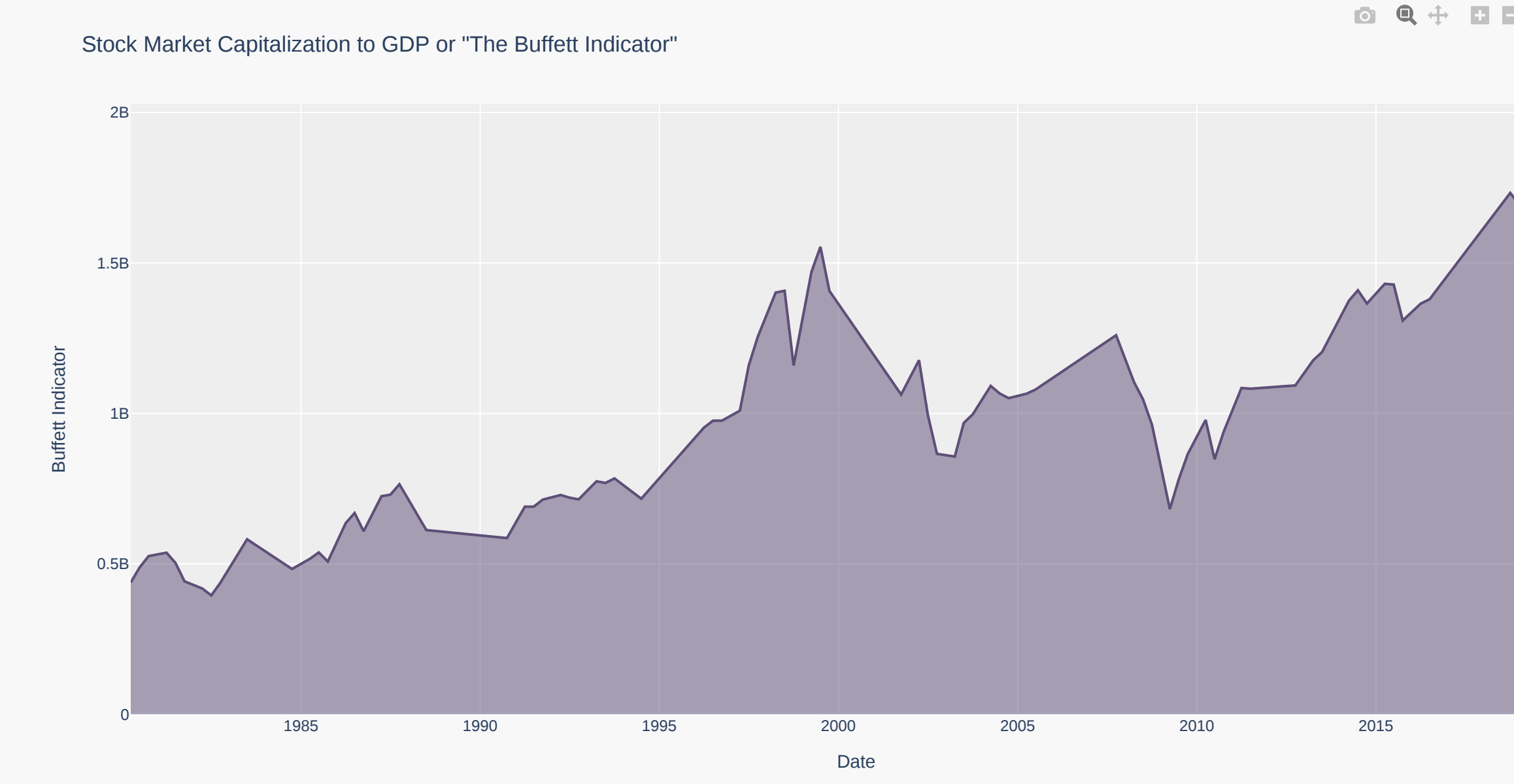
```
In [3]:    indicator_chart = px.area(market_cap_to_gdp, x='Date', y='Buffett Indicator',
               title='Stock Market Capitalization to GDP or "The Buffett Indicator"',
               color_discrete_sequence=["#5e4f78"],
               labels="",
               width=1250,
               height=600)

           indicator_chart.update_layout(margin=dict(l=100,r=40,t=80, b=40),
               plot_bgcolor = "#eeeeee",
               paper_bgcolor="#f8f8f8")

           indicator_chart.show()
```



```
In [ ]:
```