Exploring the Gapminder Dataset with Plotly Express

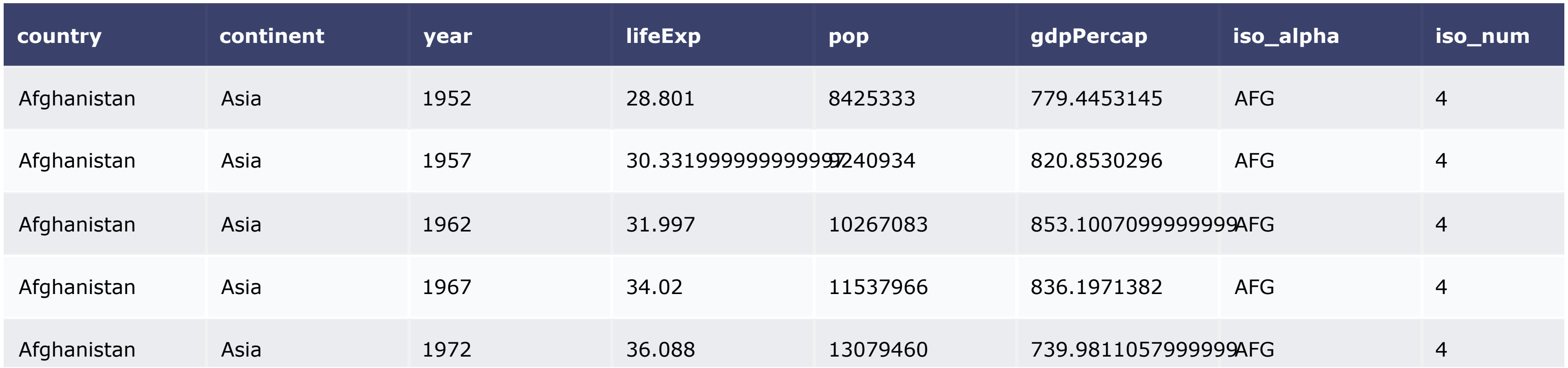
Loading the Data

In [1]:
```
import plotly.offline as py
py.init_notebook_mode(connected=True)
import plotly.graph_objs as go
import pandas as pd
import numpy as np
```

In [2]:
```
from plotly.figure_factory import create_table
import plotly.express as px

gapminder = px.data.gapminder()

table = create_table(gapminder.head())
py.iplot(table)
```



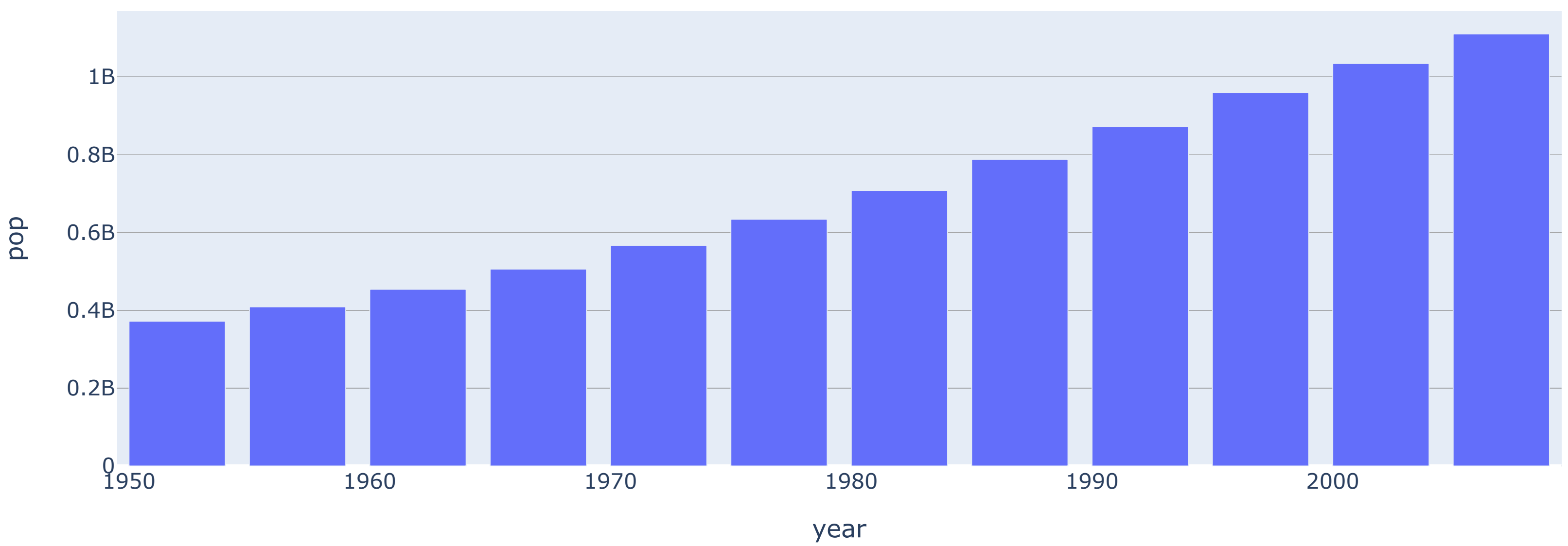
country	continent	year	lifeExp	pop	gdpPercap	iso_alpha	iso_num
Afghanistan	Asia	1952	28.801	8425333	779.4453145	AFG	4
Afghanistan	Asia	1957	30.331999999999997	9240934	820.8530296	AFG	4
Afghanistan	Asia	1962	31.997	10267083	853.1007099999999	AFG	4
Afghanistan	Asia	1967	34.02	11537966	836.1971382	AFG	4
Afghanistan	Asia	1972	36.088	13079460	739.9811057999999	AFG	4

In [3]:
```
type(gapminder)
```

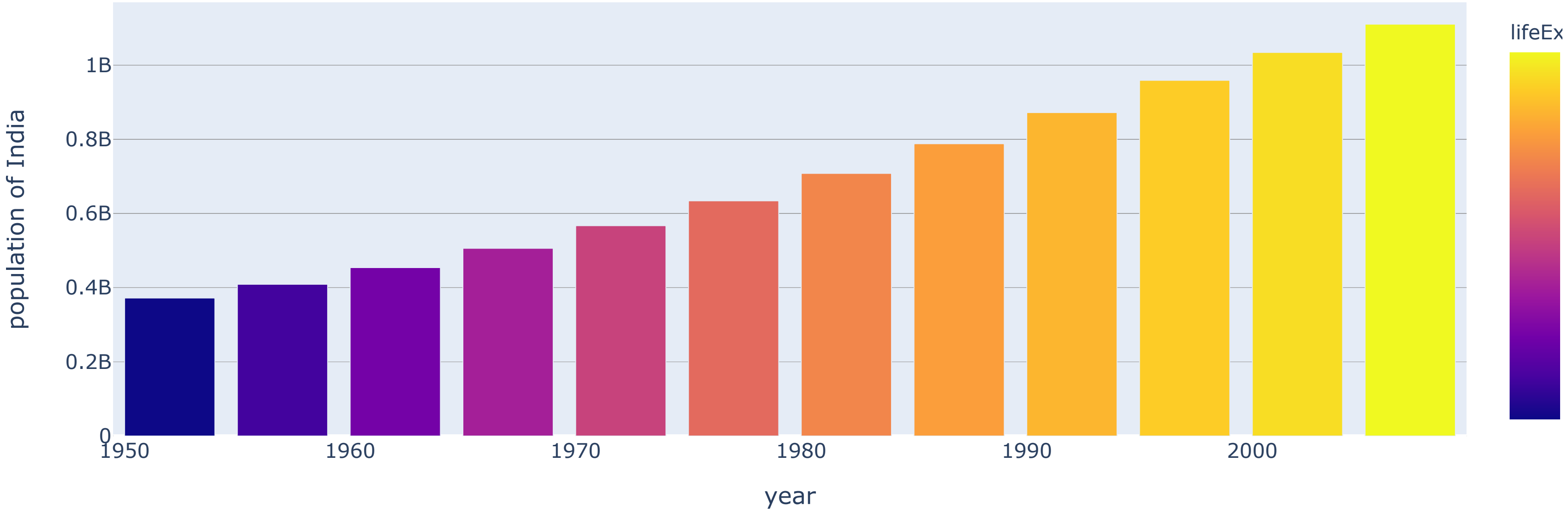
Out[3]:
```
pandas.core.frame.DataFrame
```

Quick Visualizations with Custom Bar Charts

```
In [4]: data_india = px.data.gapminder().query(" country == 'India' ")
        fig = px.bar(data_india, x = "year", y = "pop", height = 400)
        fig.show()
```

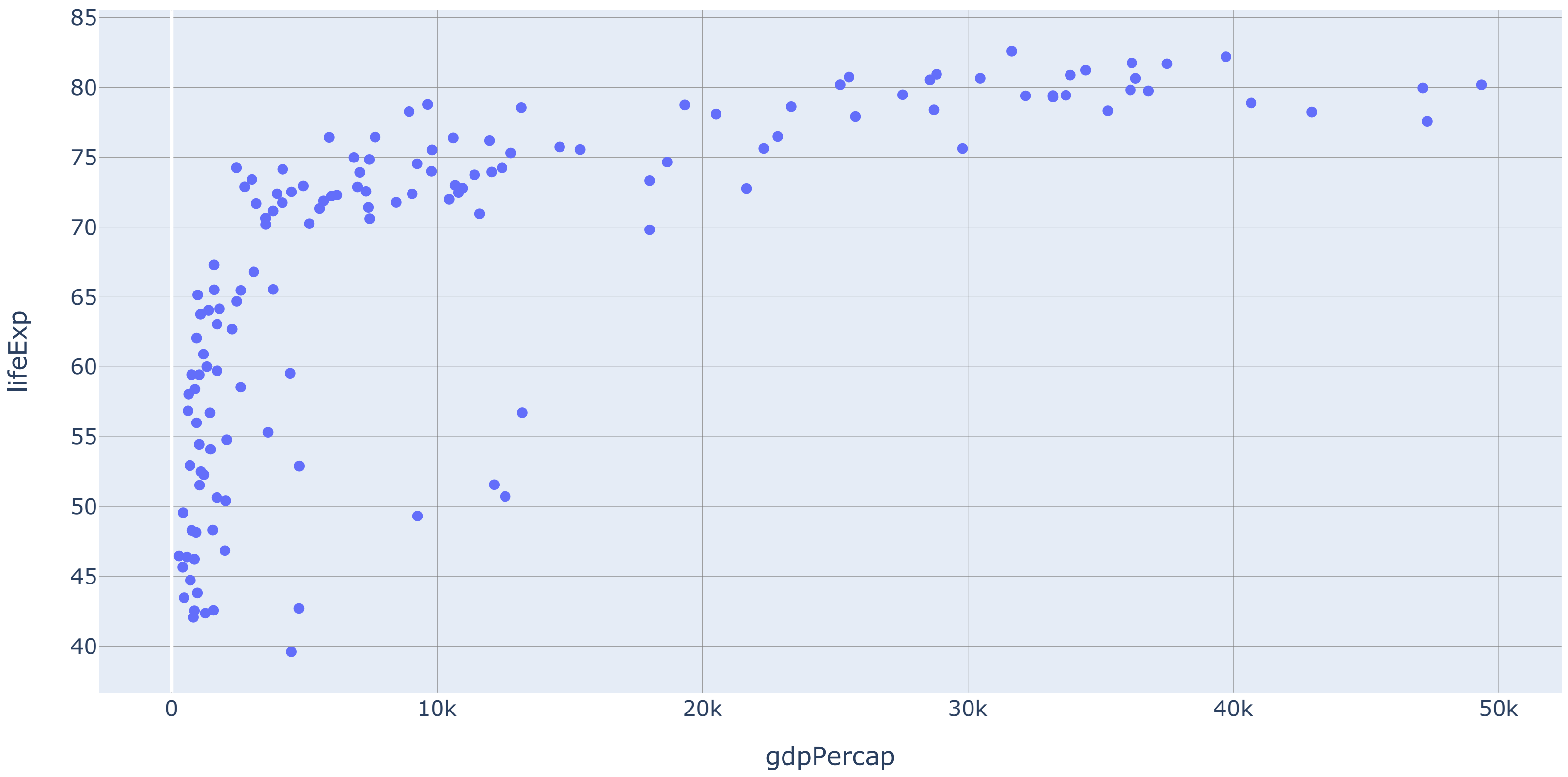


```
In [5]: fig = px.bar(data_india, x = "year", y = "pop", hover_data = ["lifeExp", "gdpPercap"],
                     color = "lifeExp", labels = {"pop":"population of India"}, height = 400)
        fig.show()
```

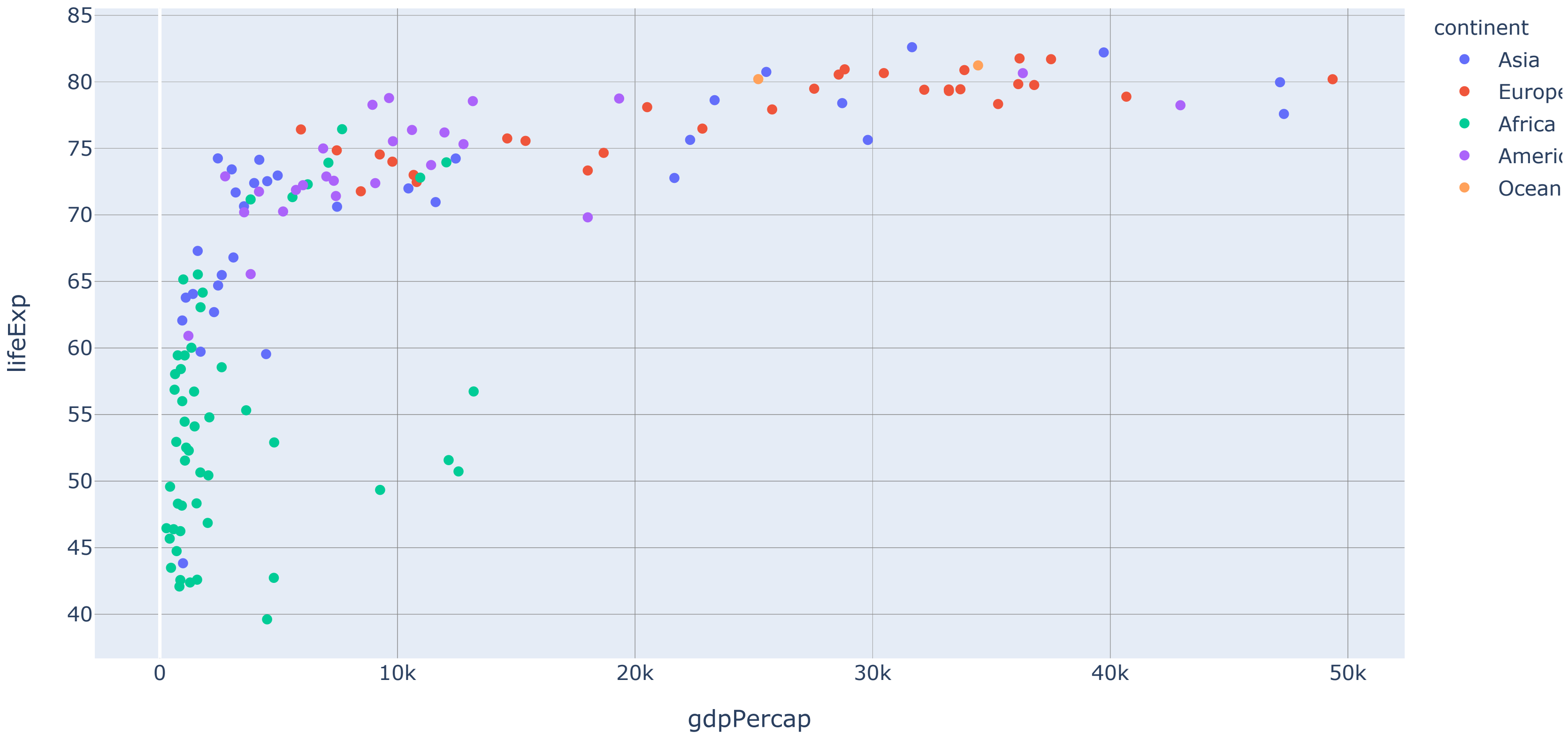


Life Expectancy vs GDP per Capita

```
In [6]: gapminder2007 = gapminder.query("year == 2007")
        px.scatter(gapminder2007, x = "gdpPercap", y = "lifeExp")
```

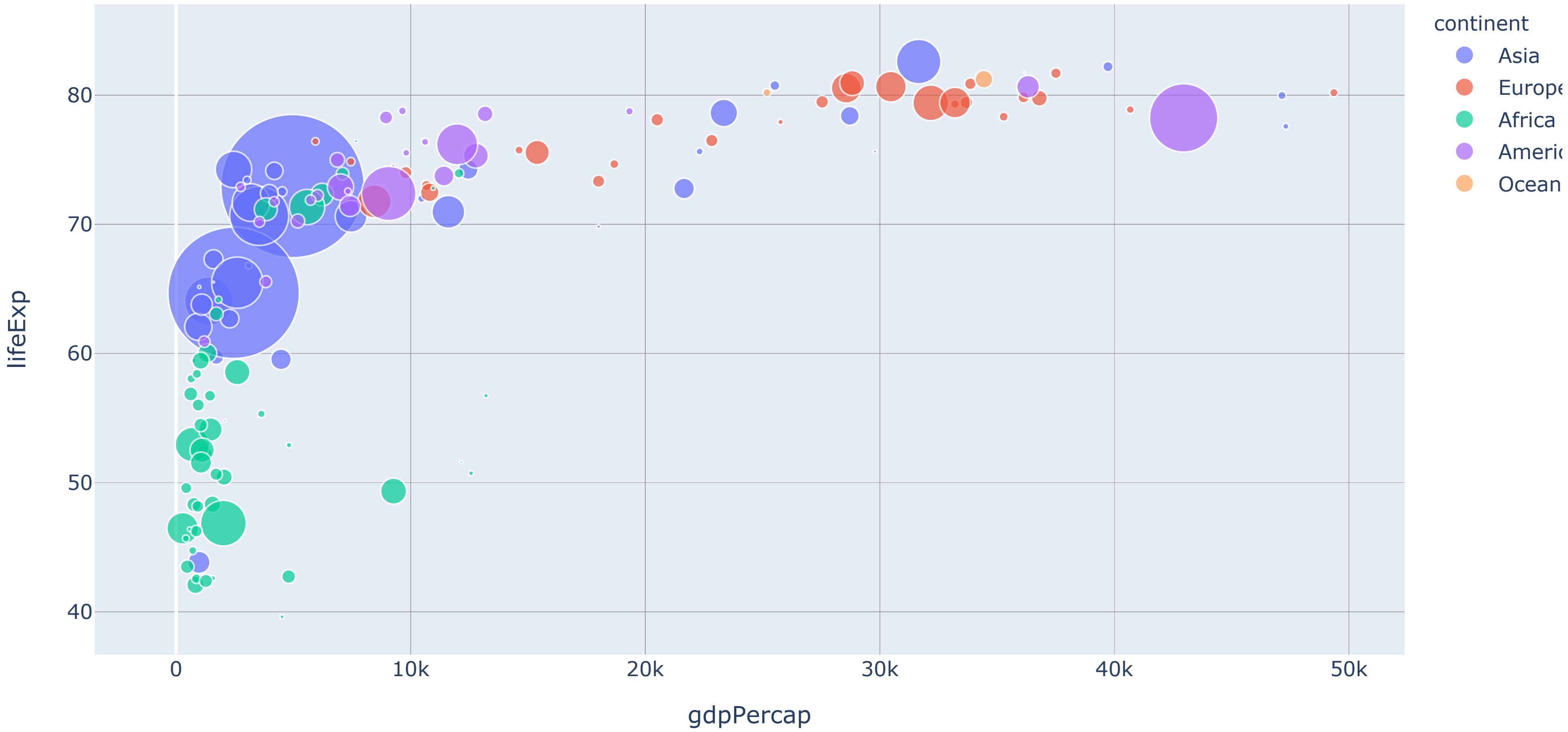


```
In [7]: px.scatter(gapminder2007, x = "gdpPercap", y = "lifeExp", color = "continent")
```

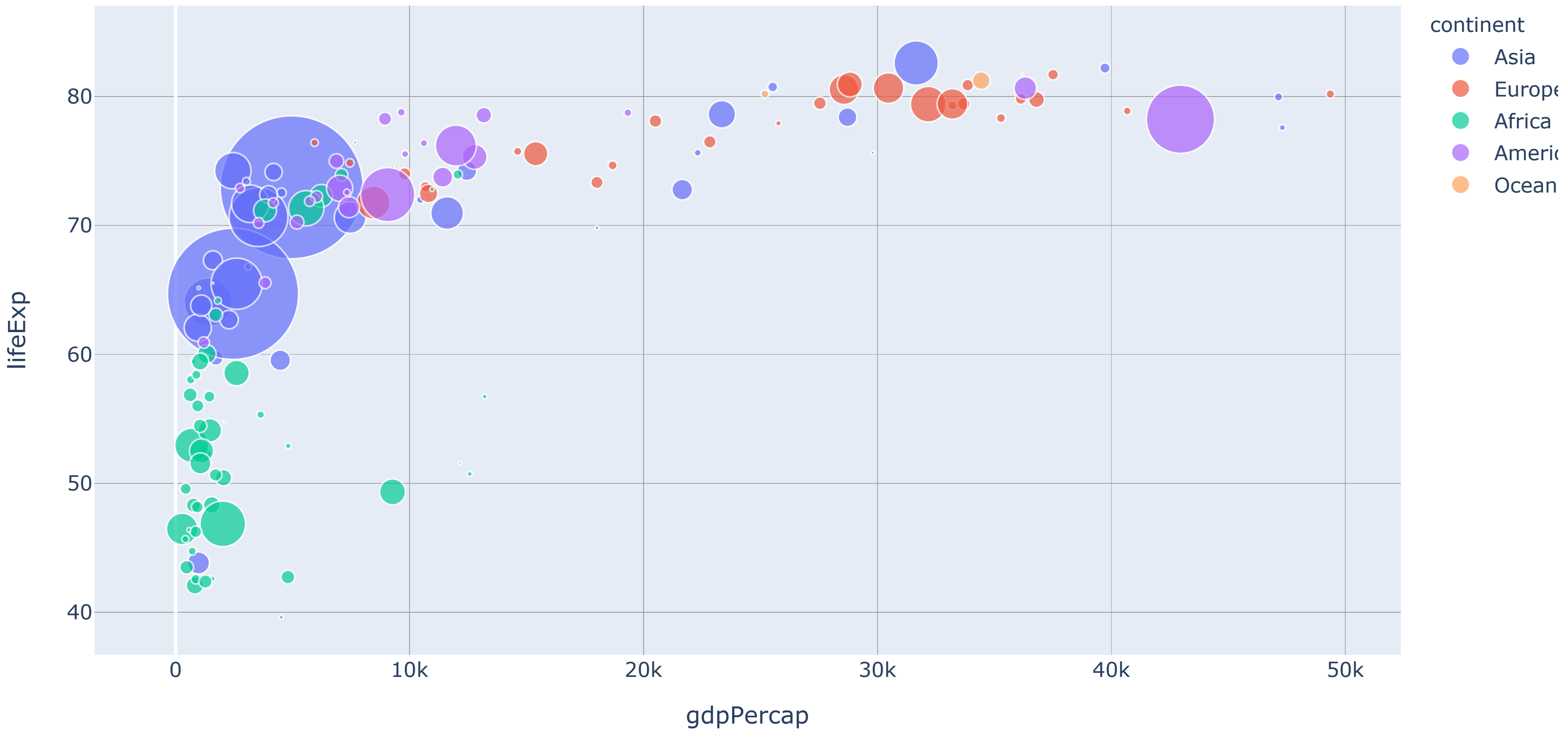


Customize Interactive Bubble Charts

```
In [8]: px.scatter(gapminder2007, x = "gdpPercap", y = "lifeExp", color = "continent", size = "pop", size_max = 60)
```

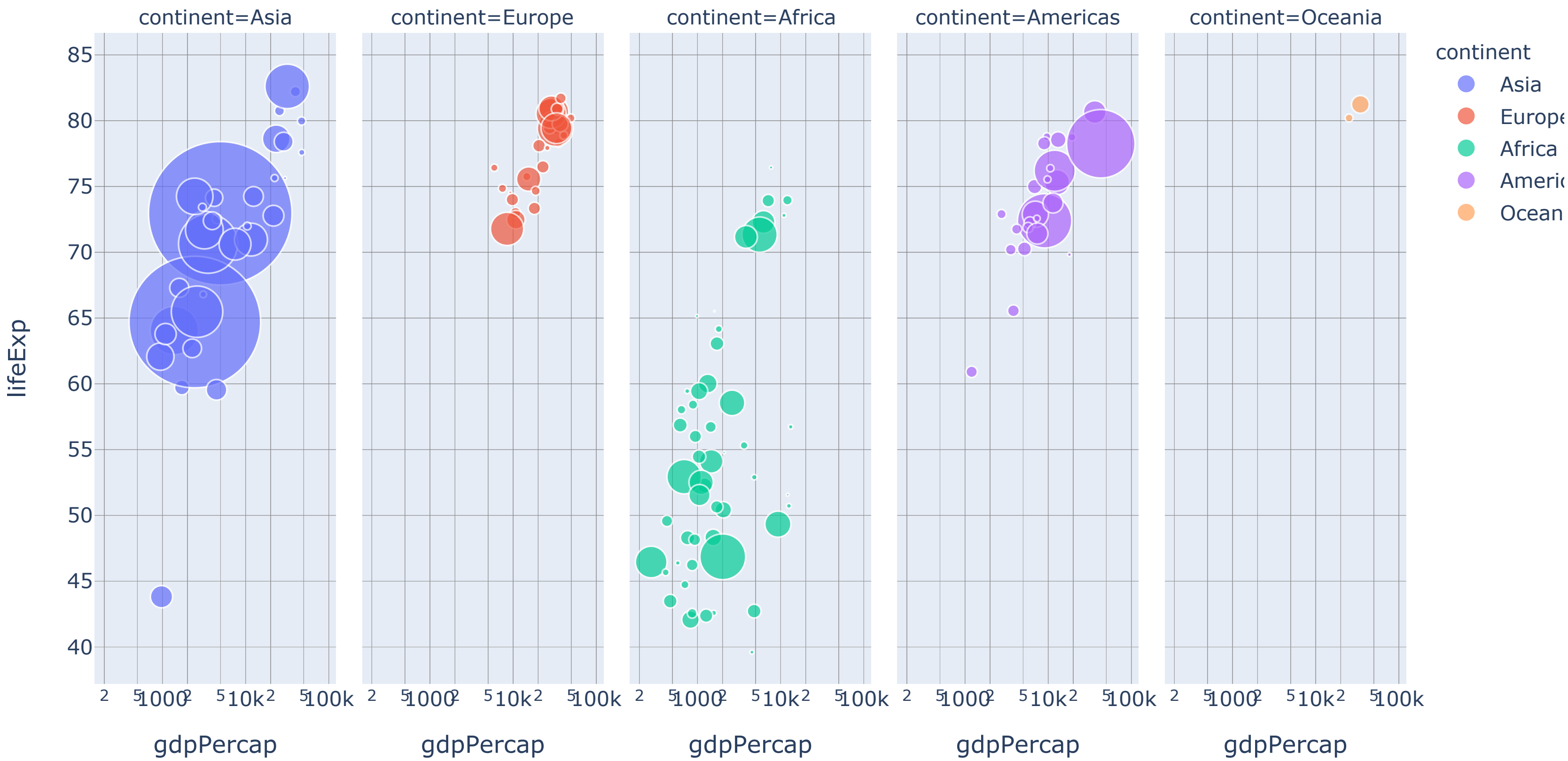


In [9]:
```
px.scatter(gapminder2007, x = "gdpPercap", y = "lifeExp", color = "continent",
           size = "pop", size_max = 60, hover_name = "country")
```

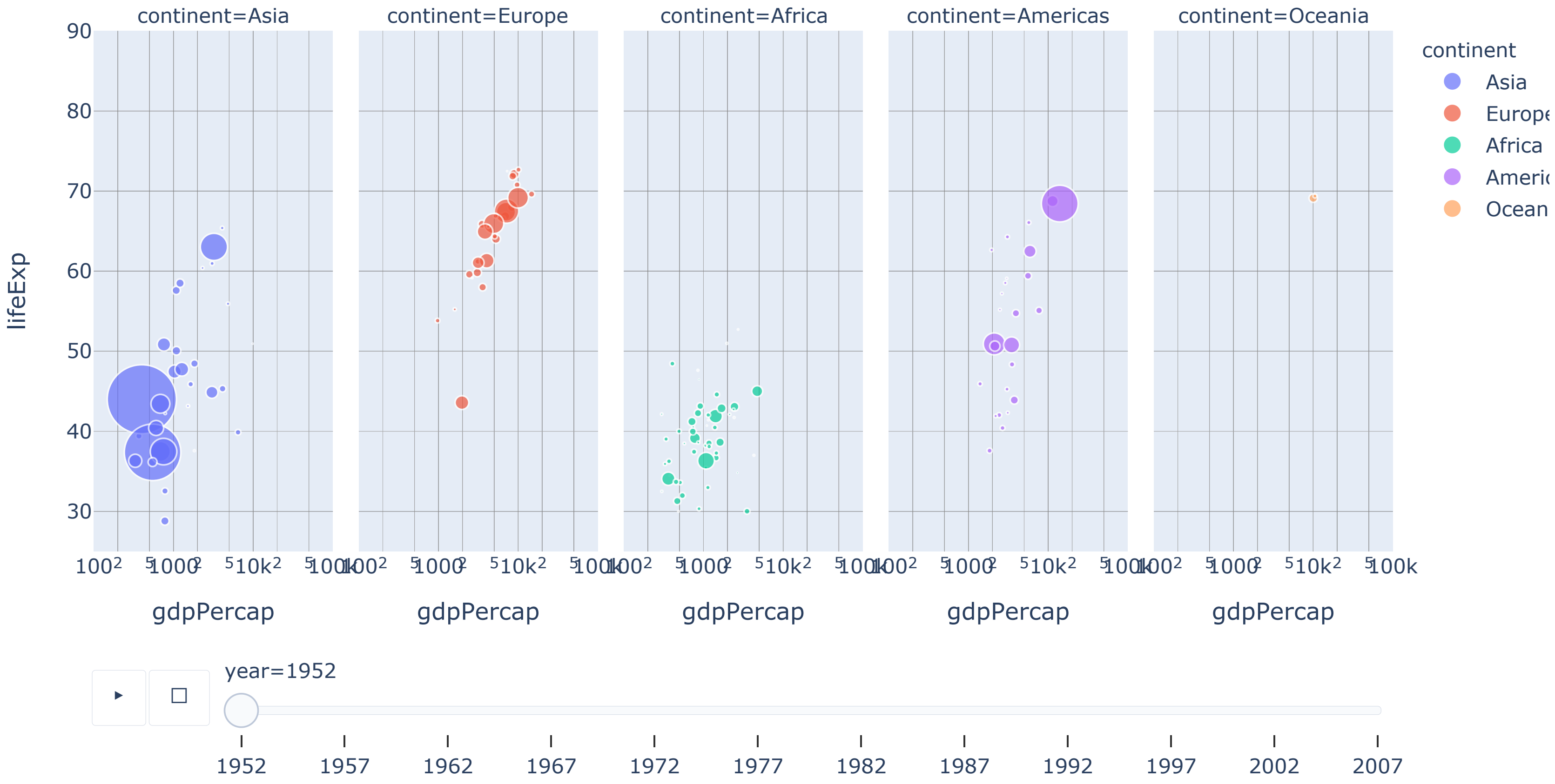


Interactive Animations and Facet Plots

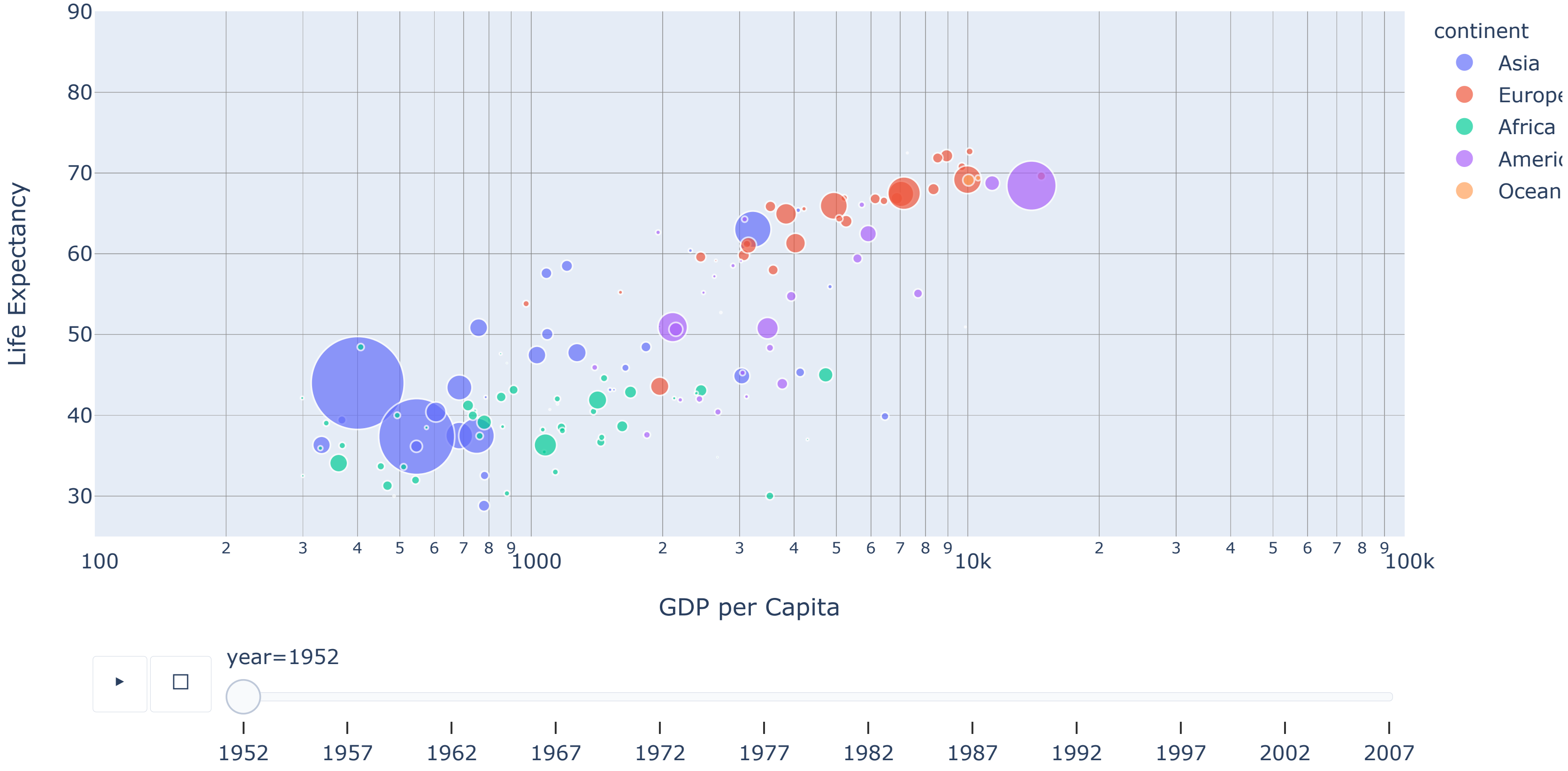
```
In [10]: px.scatter(gapminder2007, x = "gdpPercap", y = "lifeExp", color = "continent", size = "pop", size_max = 60,
                    hover_name = "country", facet_col = "continent", log_x = True)
```



```
In [11]: fig = px.scatter(gapminder, x = "gdpPercap", y = "lifeExp", animation_frame = "year", animation_group = "country",
                    size = "pop", color = "continent", hover_name = "country", facet_col = "continent",
                    log_x = True, size_max = 45, range_x = [100,100000], range_y = [25,90])
         fig.show()
```

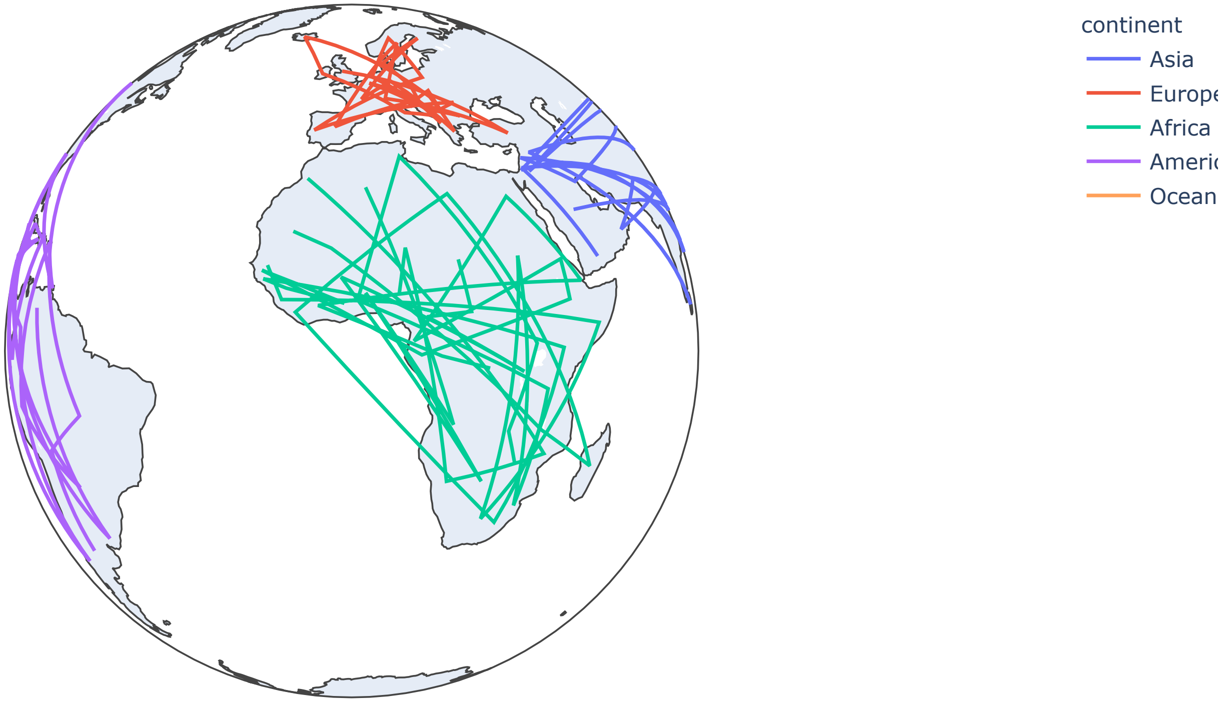


```
In [12]: px.scatter(gapminder, x = "gdpPercap", y = "lifeExp", size = "pop", size_max = 60, color = "continent", hover_name = "country",
                    animation_frame = "year", animation_group = "country", log_x = True, range_x = [100,100000], range_y = [25,90],
                    labels = dict(pop = "Population", gdpPercap = "GDP per Capita", lifeExp = "Life Expectancy"))
```

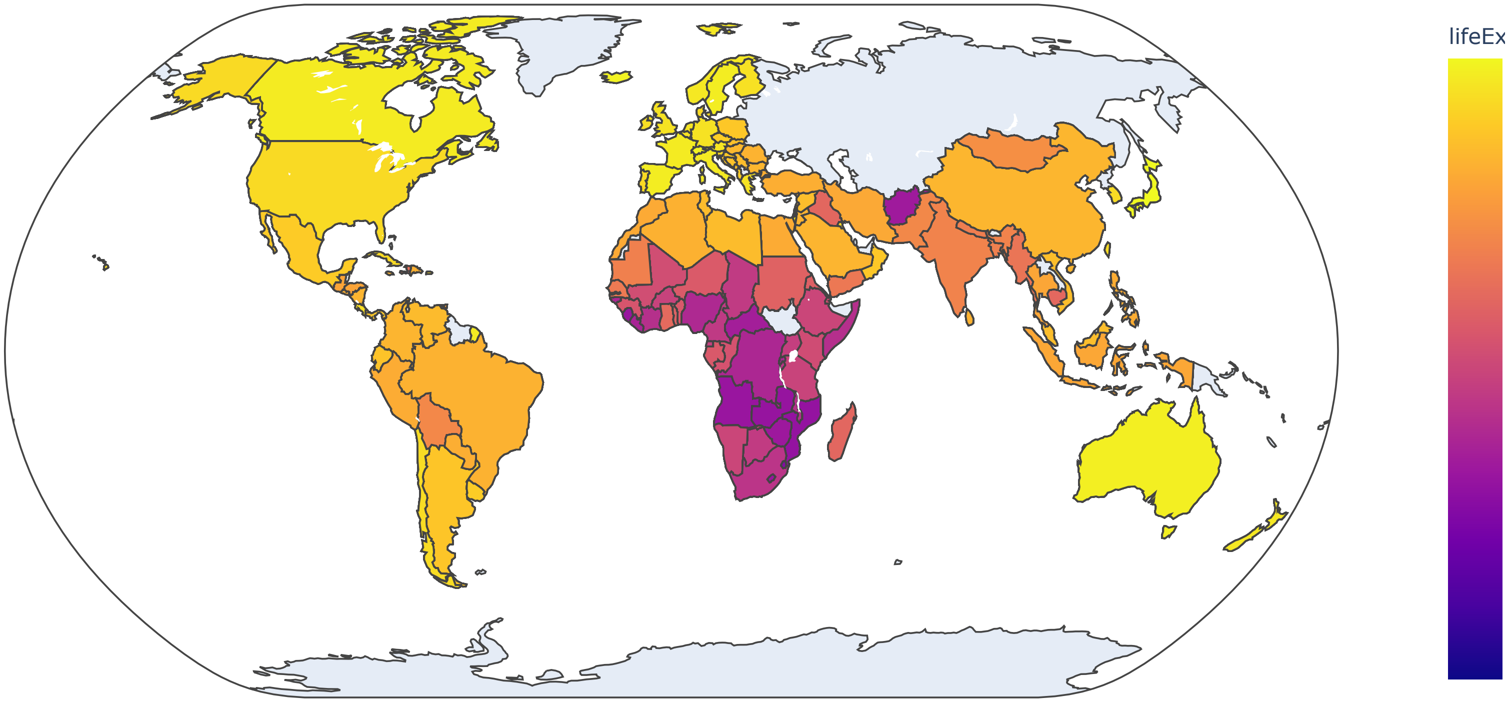


Representing Geographic Data as Animated Maps

```
In [13]: fig = px.line_geo(gapminder.query("year==2007"), locations = "iso_alpha", color = "continent", projection = "orthographic")
         fig.show()
```

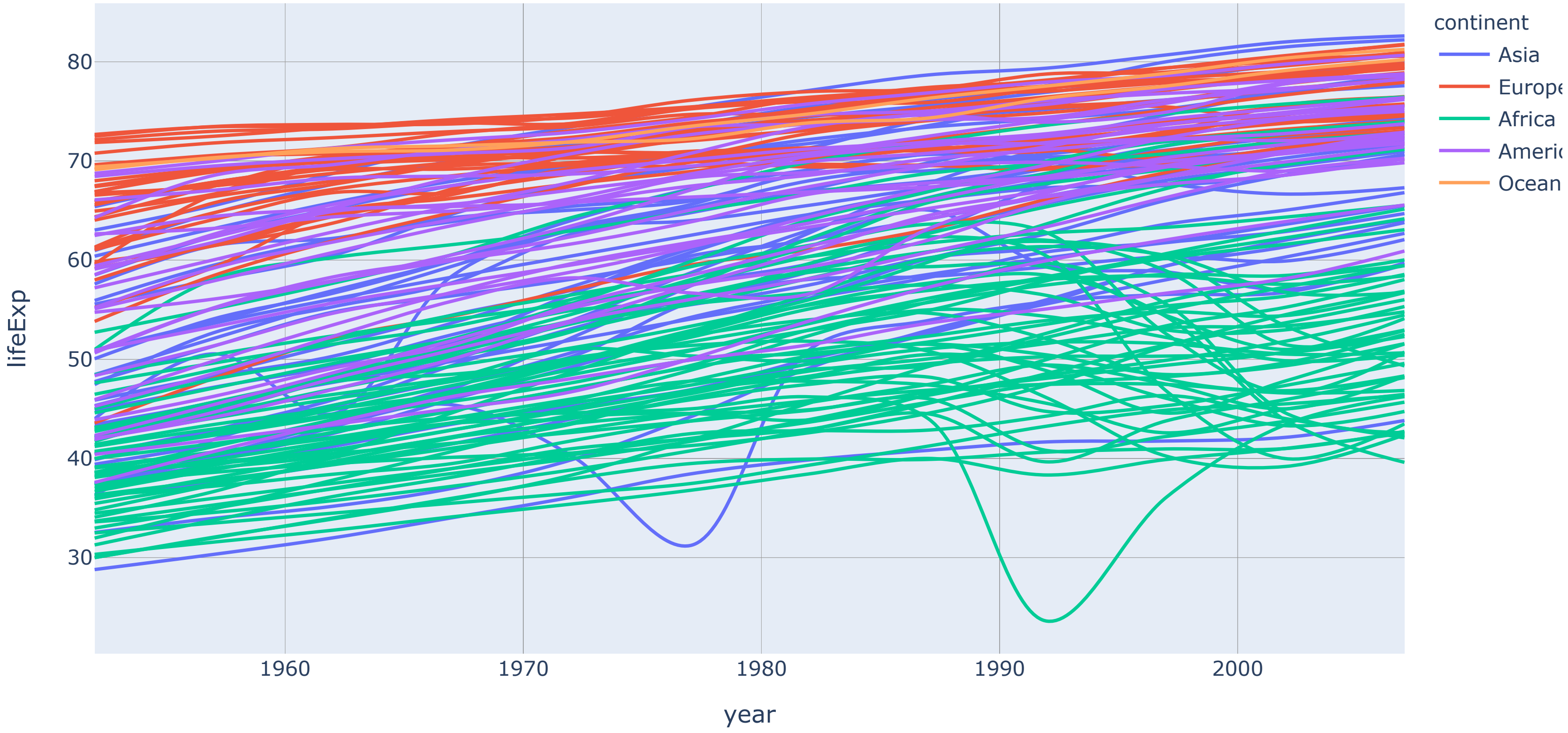


In [14]:
```
fig = px.choropleth(gapminder, locations = "iso_alpha", color = "lifeExp", hover_name = "country",
                    color_continuous_scale = px.colors.sequential.Plasma, projection = "natural earth")
fig.show()
```

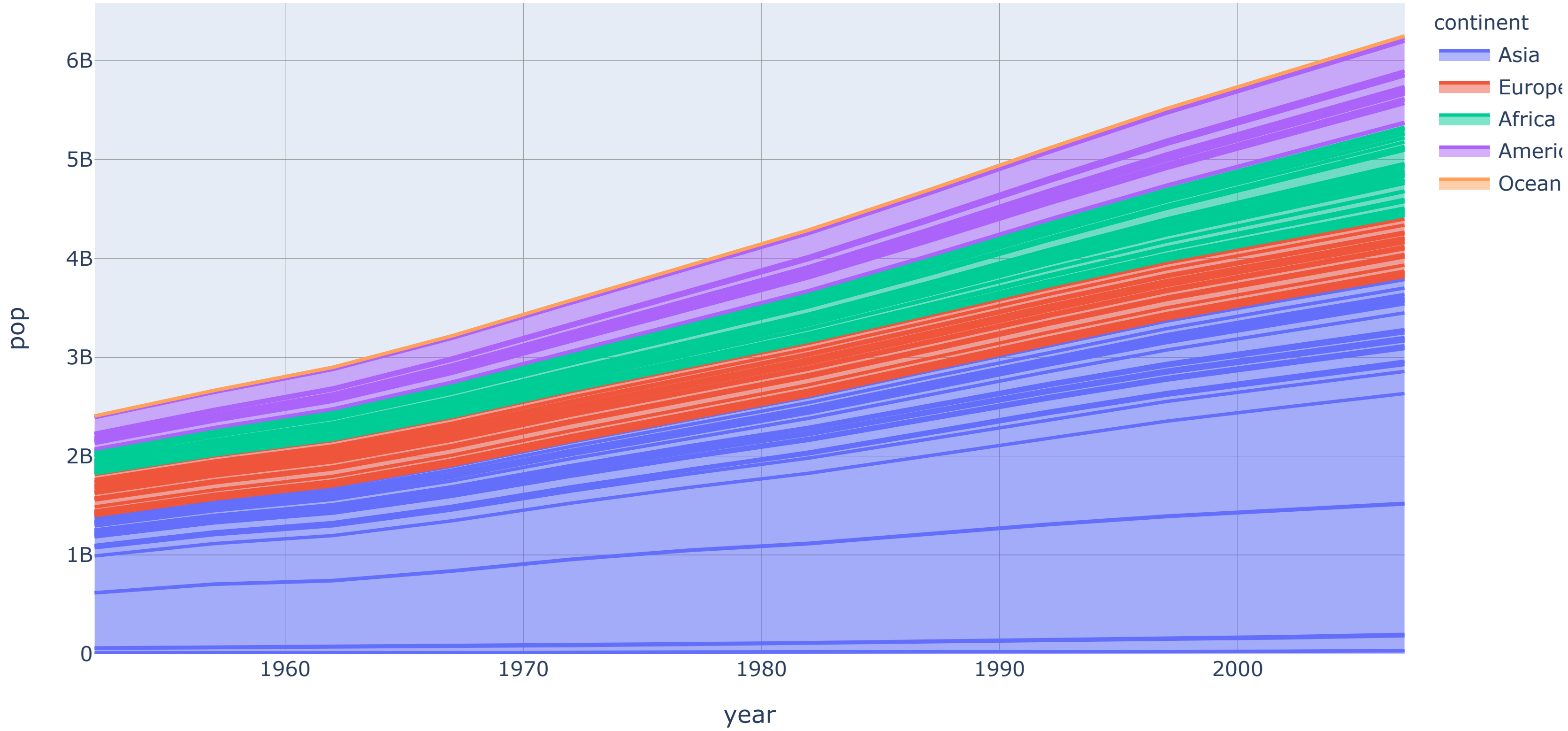


Interactive Line Plots and Area Plots

```
In [15]: fig = px.line(gapminder, x = "year", y = "lifeExp", color = "continent", line_group = "country", hover_name = "country",
                 line_shape = "spline", render_mode = "svg")
         fig.show()
```



```
In [16]: fig = px.area(gapminder, x = "year", y = "pop", color = "continent", line_group = "country")
         fig.show()
```


continent
Asia
Europe
Africa
Americ
Ocean
pop
1B
2B
3B
4B
5B
6B
year