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

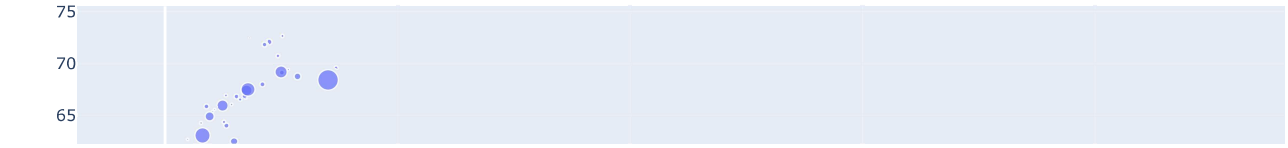
In [2]:
```
df=px.data.gapminder()
df.head(15)
```

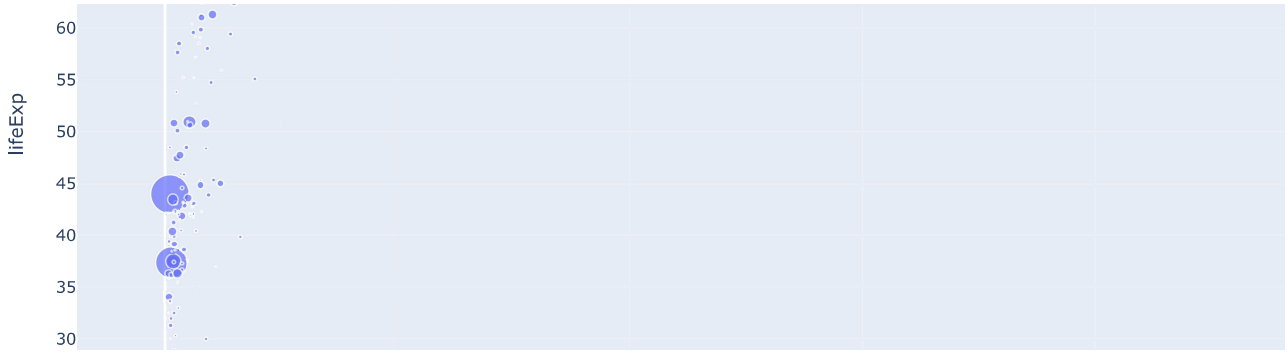
Out[2]:

	country	continent	year	lifeExp	pop	gdpPercap	iso_alpha	iso_num
0	Afghanistan	Asia	1952	28.801	8425333	779.445314	AFG	4
1	Afghanistan	Asia	1957	30.332	9240934	820.853030	AFG	4
2	Afghanistan	Asia	1962	31.997	10267083	853.100710	AFG	4
3	Afghanistan	Asia	1967	34.020	11537966	836.197138	AFG	4
4	Afghanistan	Asia	1972	36.088	13079460	739.981106	AFG	4
5	Afghanistan	Asia	1977	38.438	14880372	786.113360	AFG	4
6	Afghanistan	Asia	1982	39.854	12881816	978.011439	AFG	4
7	Afghanistan	Asia	1987	40.822	13867957	852.395945	AFG	4
8	Afghanistan	Asia	1992	41.674	16317921	649.341395	AFG	4
9	Afghanistan	Asia	1997	41.763	22227415	635.341351	AFG	4
10	Afghanistan	Asia	2002	42.129	25268405	726.734055	AFG	4
11	Afghanistan	Asia	2007	43.828	31889923	974.580338	AFG	4
12	Albania	Europe	1952	55.230	1282697	1601.056136	ALB	8
13	Albania	Europe	1957	59.280	1476505	1942.284244	ALB	8
14	Albania	Europe	1962	64.820	1728137	2312.888958	ALB	8

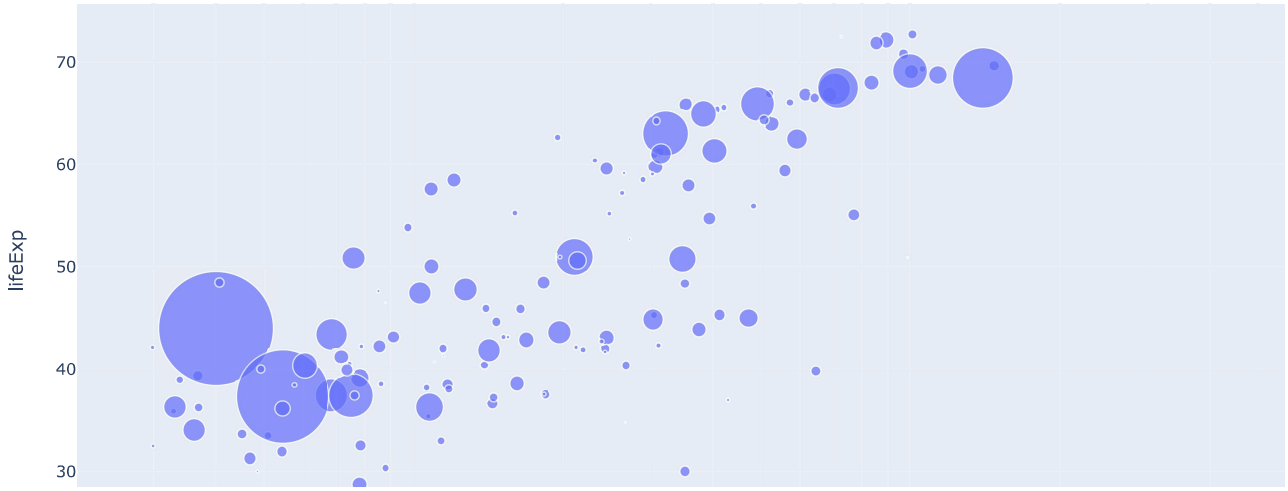
In [3]:
```
fig=px.scatter(df.query("year==1952"),x="gdpPercap", y="lifeExp")
fig.show()
```

In [4]:
```
fig=px.scatter(df.query("year==1952"),x="gdpPercap", y="lifeExp", size="pop")
fig.show()
```

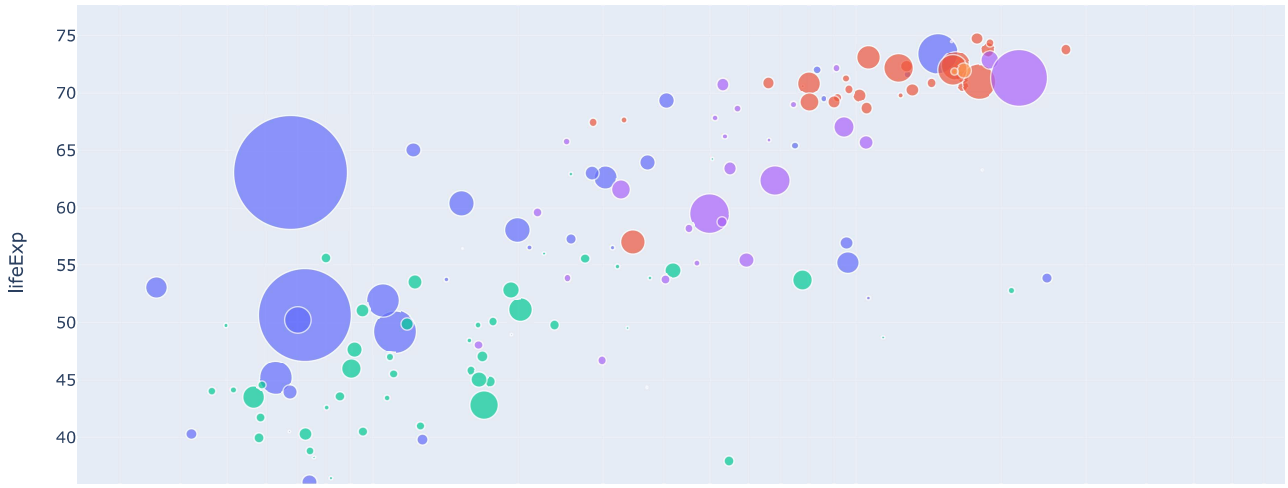




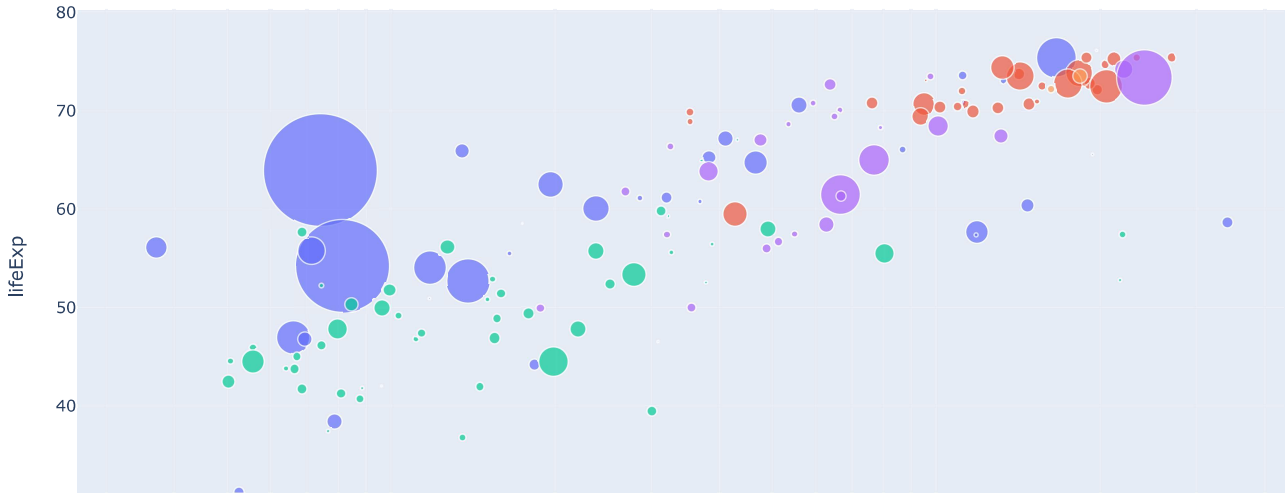
In [5]:
```
fig = px.scatter(df.query("year==1952"), x="gdpPercap", y="lifeExp", size="pop", log_x=True, size_max=60)
fig.show()
```



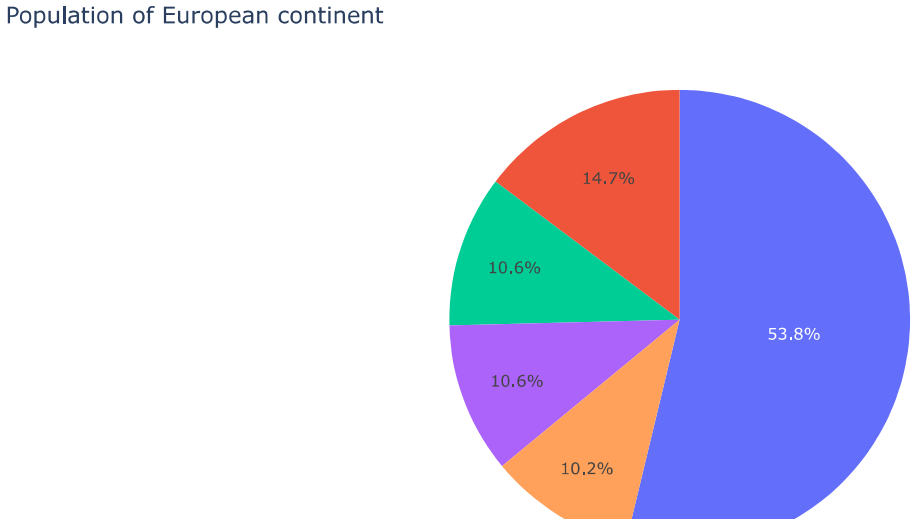
In [6]:
```
fig = px.scatter(df.query("year==1972"), x="gdpPercap", y="lifeExp", size="pop", color="continent", log_x=True, size_max=60)
fig.show()
```



In [7]:
```
fig = px.scatter(df.query("year==1977"), x="gdpPercap", y="lifeExp", size="pop", color="continent",
                 hover_name="country", log_x=True, size_max=60)
fig.show()
```

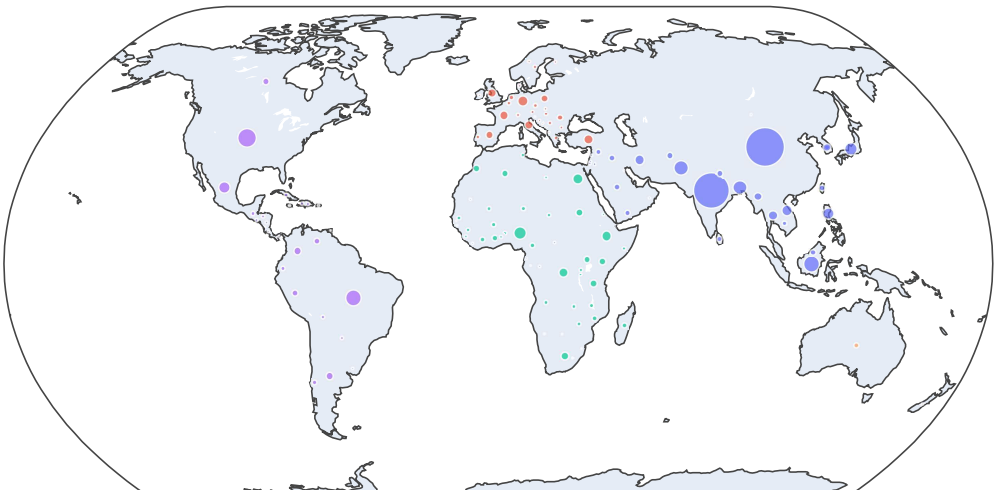


In [8]:
```
df = px.data.gapminder().query("year == 1982").query("continent == 'Europe'")
df.loc[df['pop'] < 5.e7, 'country'] = 'Other countries' # Represent only large countries
fig = px.pie(df, values='pop', names='country', title='Population of European continent')
fig.show()
```



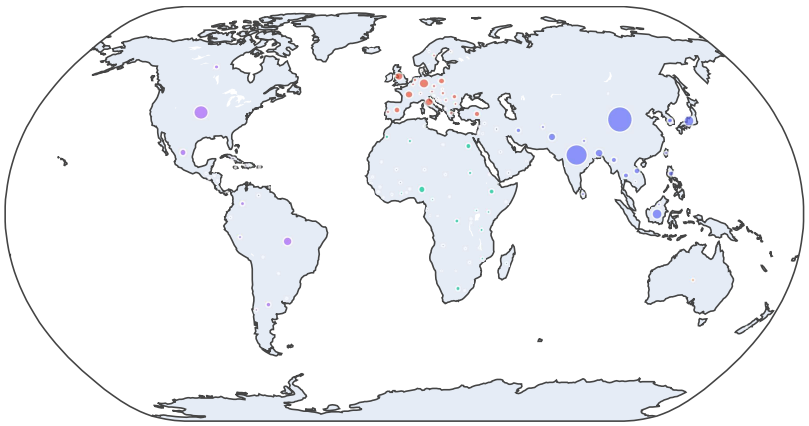
In [9]:
```
df=px.data.gapminder()

fig = px.scatter_geo(df.query("year==2007"), locations="iso_alpha", color="continent",
                     hover_name="country", size="pop",
```

```
                    projection="natural earth")
fig.show()
```



In [10]:

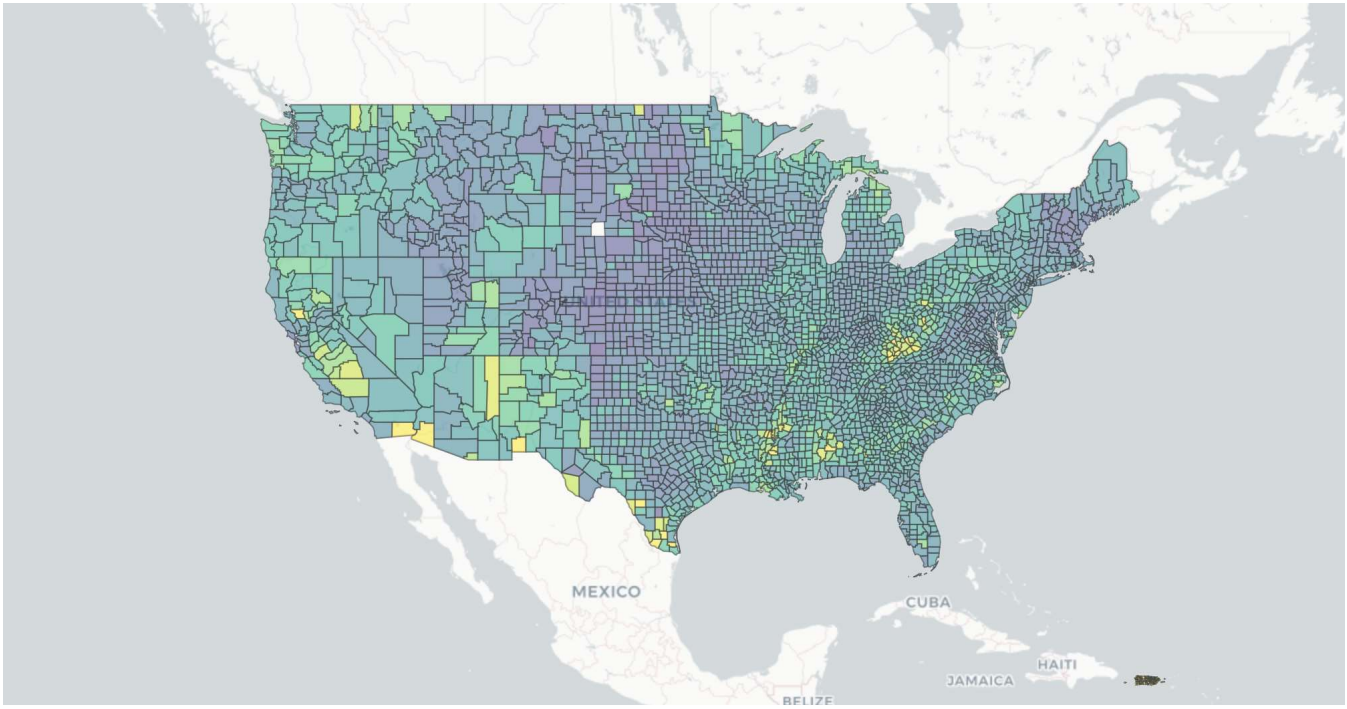
```
df = px.data.gapminder()
fig = px.scatter_geo(df, locations="iso_alpha", color="continent",
                     hover_name="country", size="pop",
                     animation_frame="year",
                     projection="natural earth")
fig.show()
```



In [11]:

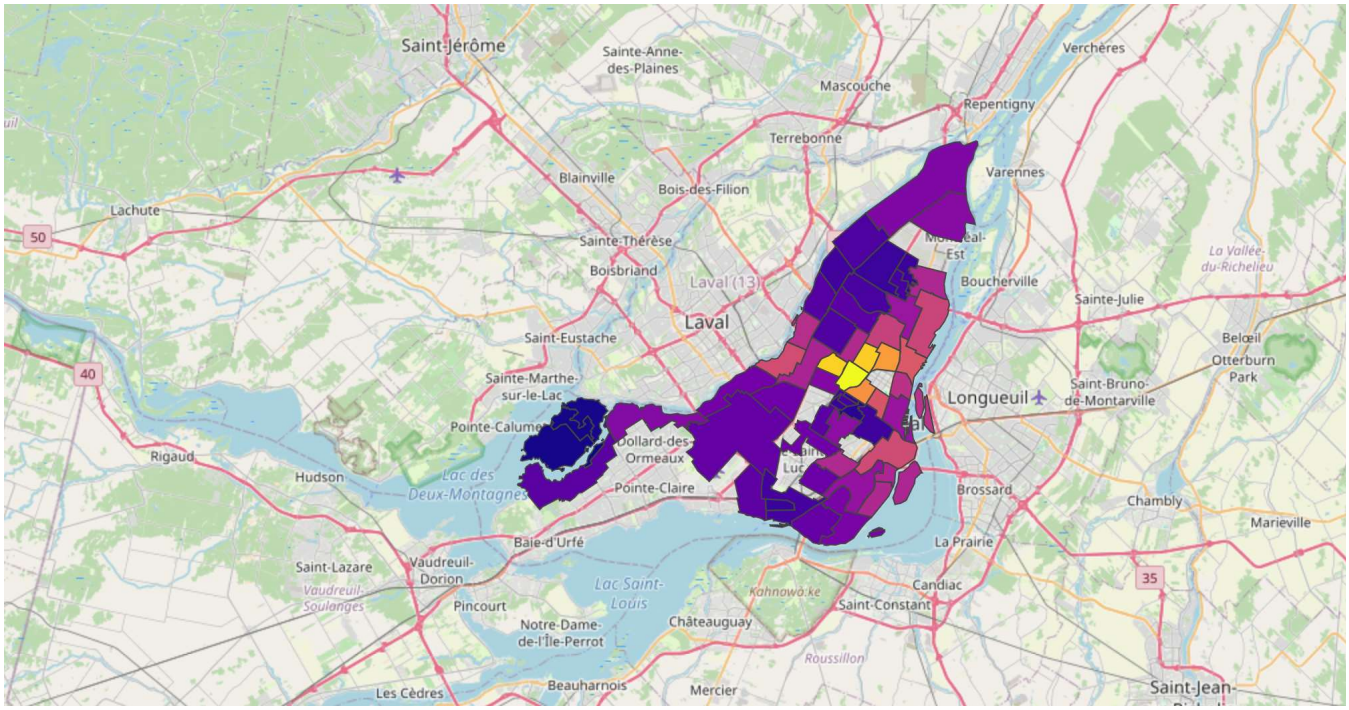
```
from urllib.request import urlopen
import json
with urlopen('https://raw.githubusercontent.com/plotly/datasets/master/geojson-counties-fips.json') as response:
    counties = json.load(response)

import pandas as pd
df = pd.read_csv("https://raw.githubusercontent.com/plotly/datasets/master/fips-unemp-16.csv",
                   dtype={"fips": str})
```

In [12]:
```
fig = px.choropleth_mapbox(df, geojson=counties, locations='fips', color='unemp',
                           color_continuous_scale="Viridis",
                           range_color=(0, 12),
                           mapbox_style="carto-positron",
                           zoom=3, center = {"lat": 37.0902, "lon": -95.7129},
                           opacity=0.5,
                           labels={'unemp':'unemployment rate'}
                          )
fig.update_layout(margin={"r":0,"t":0,"l":0,"b":0})
fig.show()
```

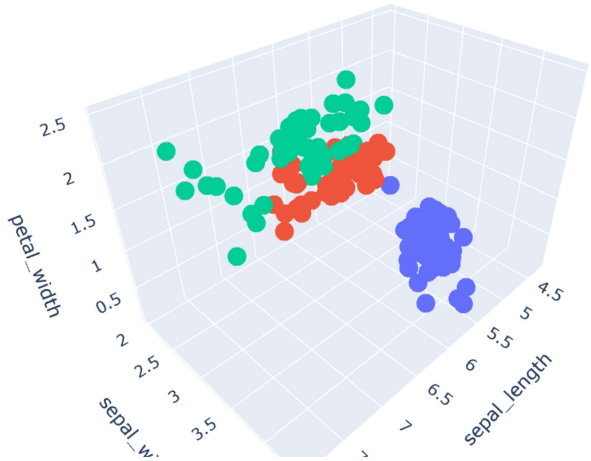


In [13]:
```
df = px.data.election()
geojson = px.data.election_geojson()

fig = px.choropleth_mapbox(df, geojson=geojson, color="Bergeron",
                           locations="district", featureidkey="properties.district",
                           center={"lat": 45.5517, "lon": -73.7073},
                           mapbox_style="carto-positron", zoom=9)
fig.update_layout(margin={"r":0,"t":0,"l":0,"b":0})
fig.show()
```

In [14]:
```
fig = px.choropleth_mapbox(df, geojson=geojson, color="Bergeron",
                           locations="district", featureidkey="properties.district",
                           center={"lat": 45.5517, "lon": -73.7073},
                           mapbox_style="open-street-map", zoom=9)
fig.update_layout(margin={"r":0,"t":0,"l":0,"b":0})
fig.show()
```



In [15]:
```
df = px.data.iris()
fig = px.scatter_3d(df, x='sepal_length', y='sepal_width', z='petal_width',
                    color='species')
fig.show()
```



In [ ]: