

Submission

Put the ipynb file and html file in the github branch you created in the last assignment and submit the link to the commit in brightspace

```
In [1]: from plotly.offline import init_notebook_mode
import plotly.io as pio
import plotly.express as px

init_notebook_mode.connected=True
pio.renderers.default = "plotly_mimetype+notebook"
```

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

```
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

Question 1:

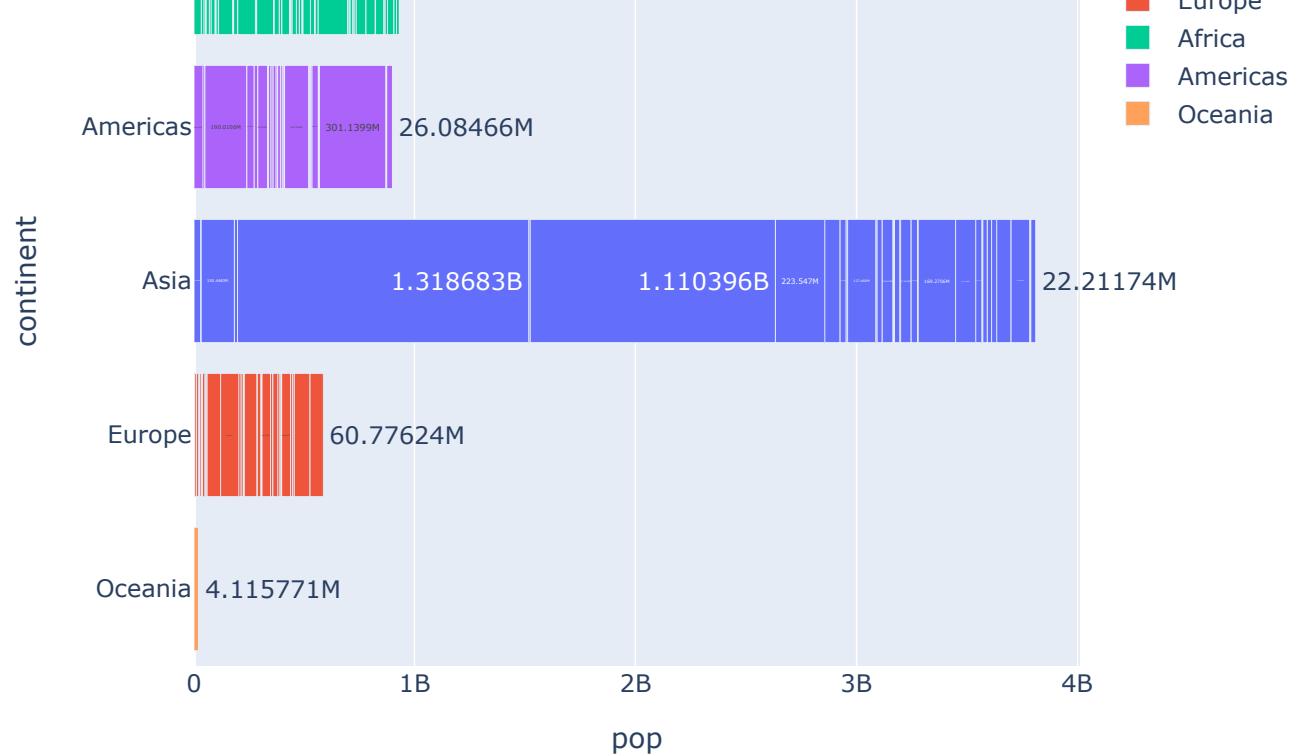
Recreate the barplot below that shows the population of different continents for the year 2007.

Hints:

- Extract the 2007 year data from the dataframe. You have to process the data accordingly
- use [plotly bar](#)
- Add different colors for different continents
- Sort the order of the continent for the visualisation. Use [axis layout setting](#)
- Add text to each bar that represents the population

```
In [3]: df_2007 = df.query('year==2007')
fig = px.bar(df_2007, x='pop', y='continent', color='continent',
             orientation='h', text_auto=True)
fig.update_yaxes(categoryorder='category descending')
fig.update_traces(textfont_size=12, textangle=0, textposition='outside',
                  cliponaxis=False)
fig.show()
```



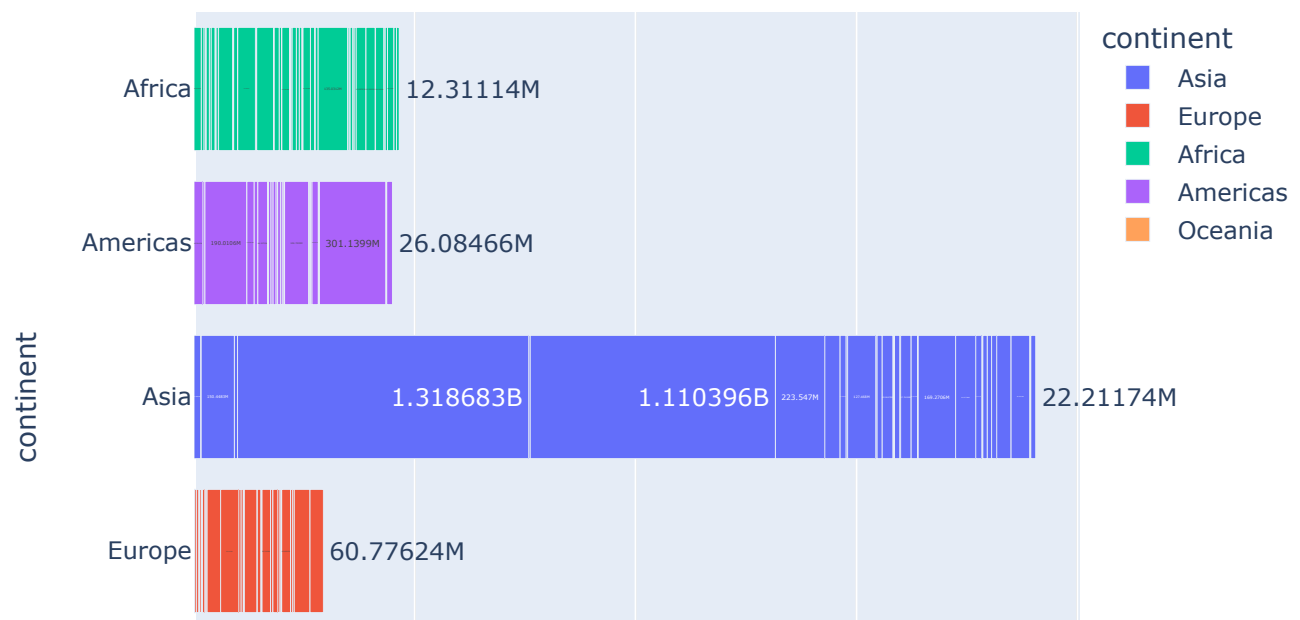


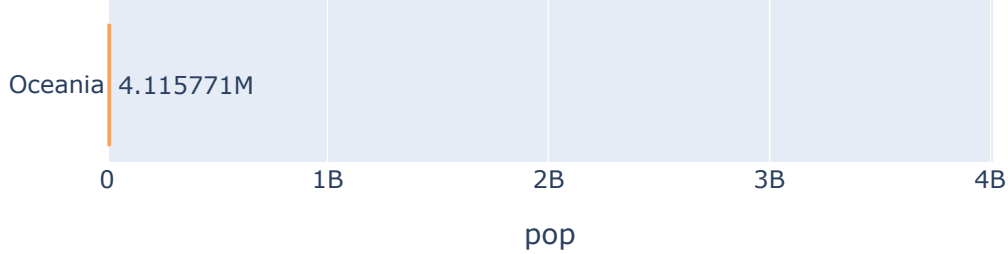
Question 2:

Sort the order of the continent for the visualisation

Hint: Use [axis layout setting](#)

```
In [4]: df_2007 = df.query('year==2007')
fig = px.bar(df_2007, x='pop', y='continent', color='continent',
             orientation='h', text_auto=True)
fig.update_yaxes(categoryorder='category descending')
fig.update_traces(textfont_size=12, textangle=0, textposition='outside',
                  cliponaxis=False)
fig.show()
```

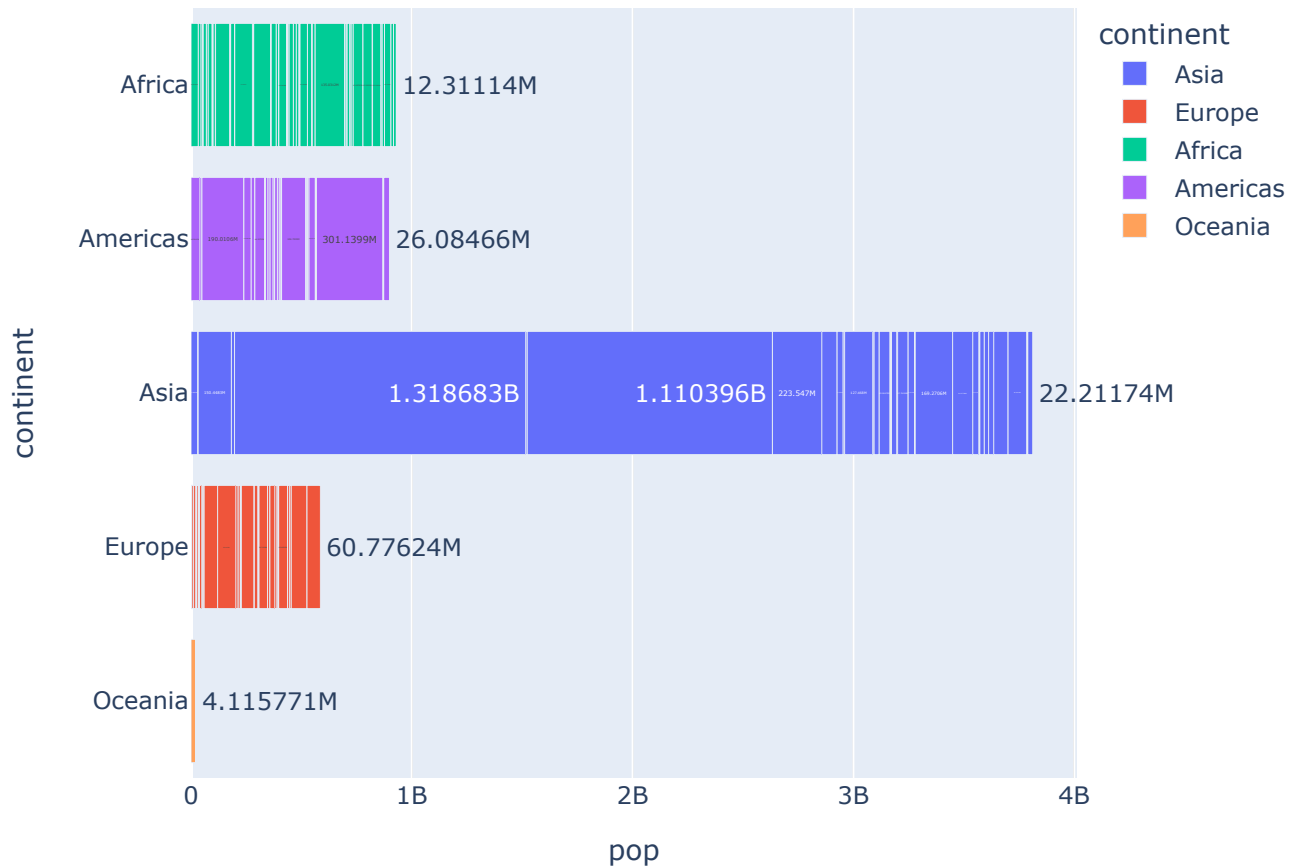




Question 3:

Add text to each bar that represents the population

```
In [5]: df_2007 = df.query('year==2007')
fig = px.bar(df_2007, x='pop', y='continent', color='continent',
             orientation='h', text_auto=True)
fig.update_yaxes(categoryorder='category descending')
fig.update_traces(textfont_size=12, textangle=0, textposition='outside',
                  cliponaxis=False)
fig.show()
```



Question 4:

Thus far we looked at data from one year (2007). Lets create an animation to see the population growth of the continents through the years

```
In [6]: df
```

Out[6]:

	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
...
1699	Zimbabwe	Africa	1987	62.351	9216418	706.157306	ZWE	716
1700	Zimbabwe	Africa	1992	60.377	10704340	693.420786	ZWE	716
1701	Zimbabwe	Africa	1997	46.809	11404948	792.449960	ZWE	716
1702	Zimbabwe	Africa	2002	39.989	11926563	672.038623	ZWE	716
1703	Zimbabwe	Africa	2007	43.487	12311143	469.709298	ZWE	716

1704 rows × 8 columns

```
In [7]: df_new = df.groupby(['continent', 'year']).sum().reset_index()  
df_new
```

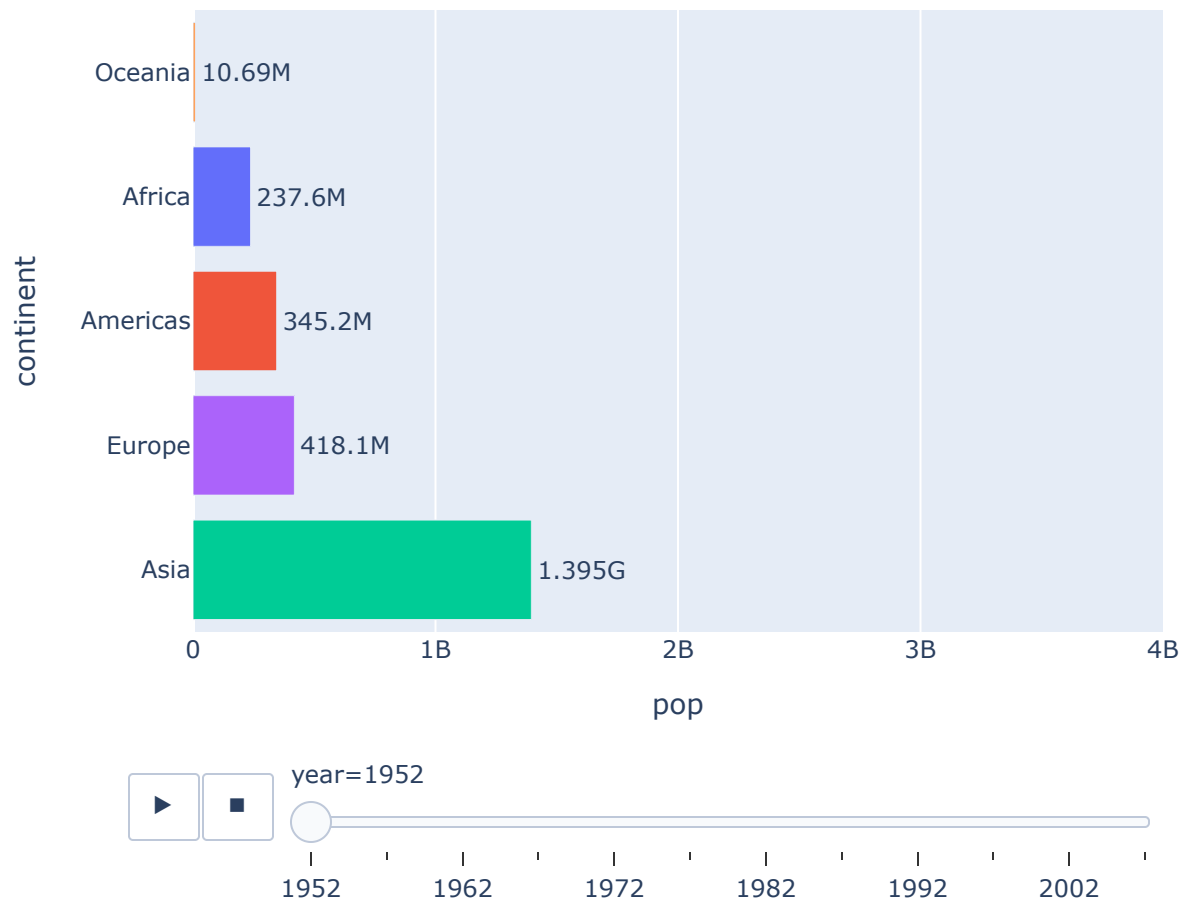
Out[7]:

	continent	year	lifeExp	pop	gdpPercap	iso_num
0	Africa	1952	2035.04600	237640501	65133.768223	23859
1	Africa	1957	2145.85000	264837738	72032.275237	23859
2	Africa	1962	2252.61100	296516865	83100.098892	23859
3	Africa	1967	2357.39600	335289489	106618.917645	23859
4	Africa	1972	2467.44900	379879541	121660.015058	23859
5	Africa	1977	2578.18200	433061021	134468.802440	23859
6	Africa	1982	2682.82900	499348587	129042.833907	23859
7	Africa	1987	2773.92900	574834110	118698.787546	23859
8	Africa	1992	2788.73800	659081517	118654.137329	23859
9	Africa	1997	2787.11000	743832984	123695.496865	23859
10	Africa	2002	2772.91200	833723916	135168.028262	23859
11	Africa	2007	2849.91400	929539692	160629.695446	23859
12	Americas	1952	1331.99600	345152446	101976.563805	9843
13	Americas	1957	1399.00700	386953916	115401.093329	9843
14	Americas	1962	1459.96900	433270254	122538.546760	9843
15	Americas	1967	1510.27300	480746623	141706.337401	9843
16	Americas	1972	1559.87300	529384210	162283.353476	9843
17	Americas	1977	1609.78900	578067699	183800.178157	9843
18	Americas	1982	1655.72100	630290920	187668.427202	9843

19	Americas	1987	1702.26800	682753971	194835.006528	9843
20	Americas	1992	1739.20900	739274104	201123.360138	9843
21	Americas	1997	1778.76200	796900410	222232.521564	9843
22	Americas	2002	1810.55100	849772762	232191.927683	9843
23	Americas	2007	1840.20300	898871184	275075.790634	9843
24	Asia	1952	1528.37500	1395357351	171450.972133	13354
25	Asia	1957	1627.51196	1562780599	190995.187018	13354
26	Asia	1962	1701.58636	1696357182	189069.197618	13354
27	Asia	1967	1803.90012	1905662900	197048.721329	13354
28	Asia	1972	1891.53588	2150972248	270186.467082	13354
29	Asia	1977	1967.14836	2384513556	257113.362658	13354
30	Asia	1982	2066.39200	2610135582	245326.460197	13354
31	Asia	1987	2140.08900	2871220762	251071.474755	13354
32	Asia	1992	2195.72800	3133292191	285109.778175	13354
33	Asia	1997	2244.67700	3383285500	324525.078743	13354
34	Asia	2002	2284.71800	3601802203	335744.983087	13354
35	Asia	2007	2334.04000	3811953827	411609.886714	13354
36	Europe	1952	1932.25500	418120846	169831.723043	12829
37	Europe	1957	2001.09200	437890351	208890.384478	12829
38	Europe	1962	2056.17700	460355155	250964.604429	12829
39	Europe	1967	2092.12800	481178958	304314.712696	12829
40	Europe	1972	2123.25100	500635059	374387.257394	12829
41	Europe	1977	2158.13300	517164531	428519.373288	12829
42	Europe	1982	2184.19200	531266901	468536.896537	12829
43	Europe	1987	2209.26500	543094160	516429.321799	12829
44	Europe	1992	2233.20300	558142797	511847.042526	12829
45	Europe	1997	2265.15500	568944148	572303.454048	12829
46	Europe	2002	2301.01800	578223869	651351.972673	12829
47	Europe	2007	2329.45800	586098529	751634.449078	12829
48	Oceania	1952	138.51000	10686006	20596.171300	590
49	Oceania	1957	140.59000	11941976	23197.044910	590
50	Oceania	1962	142.17000	13283518	25392.904860	590
51	Oceania	1967	142.62000	14600414	28990.043580	590
52	Oceania	1972	143.82000	16106100	32834.666760	590
53	Oceania	1977	145.71000	17239000	34567.915210	590
54	Oceania	1982	148.58000	18394850	37109.419680	590
55	Oceania	1987	150.64000	19574415	40896.080320	590
56	Oceania	1992	153.89000	20919651	41788.091770	590

57	Oceania	1997	156.38000	22241430	48048.350340	590
58	Oceania	2002	159.48000	23454829	53877.556080	590
59	Oceania	2007	161.43900	24549947	59620.376550	590

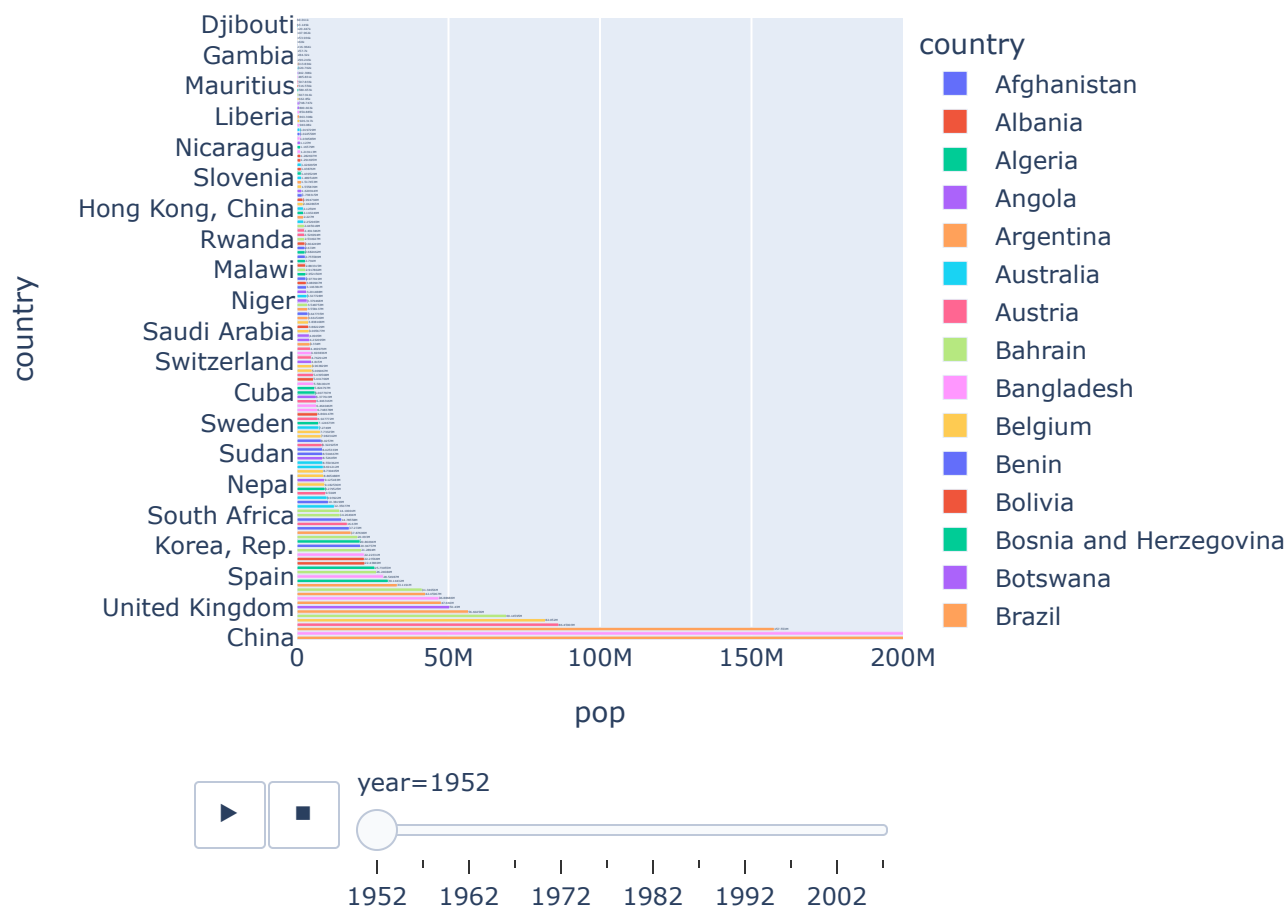
```
In [8]: fig2 = px.bar(df_new, x='pop', y='continent', color='continent', animation_frame="year",
                    animation_group = 'continent', orientation='h', text_auto='.4s')
fig2.update_layout(showlegend=False, xaxis_range=[0,4000000000])
fig2.update_yaxes(categoryorder='total_descending')
fig2.update_traces(textfont_size=12, textangle=0, textposition='outside',
                    cliponaxis=False)
fig2.show()
```



Question 5:

Instead of the continents, let's look at individual countries. Create an animation that shows the population growth of the countries through the years

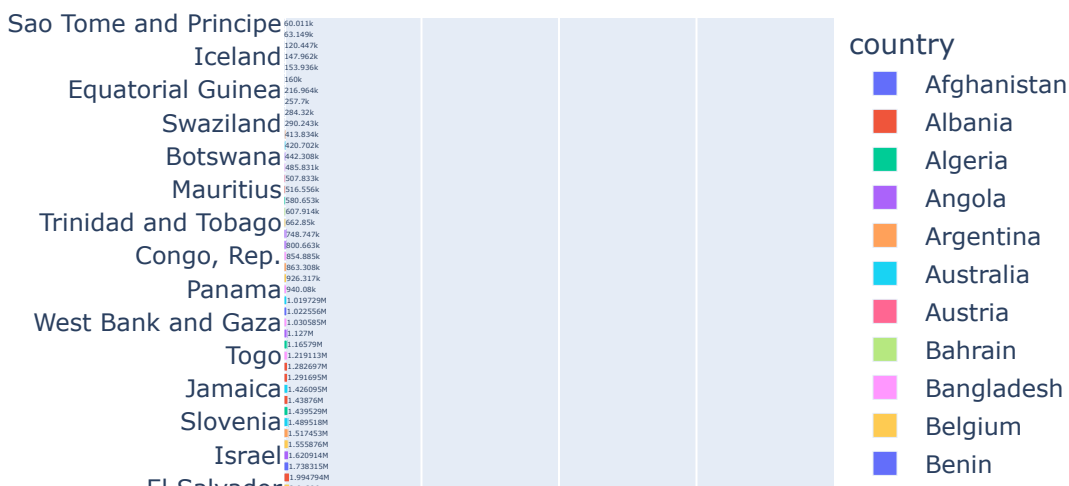
```
In [9]: fig3 = px.bar(df, x='pop', y='country', color='country', animation_frame="year",
                    animation_group = 'country', orientation='h', text_auto=True)
fig3.update_xaxes(range=(0,2000000000))
fig3.update_yaxes(categoryorder='total_descending')
fig3.update_traces(textfont_size=12, textangle=0, textposition='outside',
                    cliponaxis=False)
fig3.show()
```

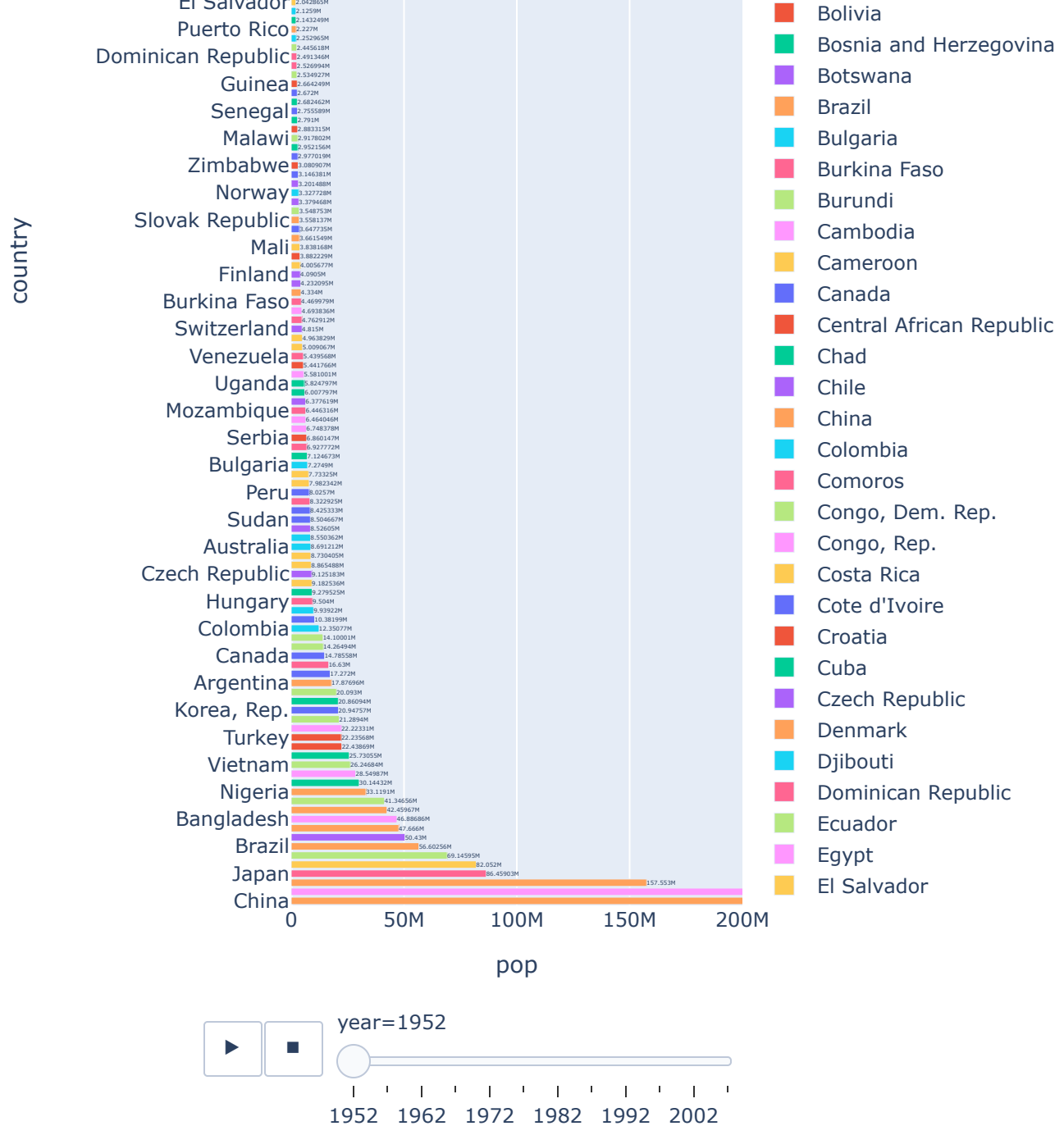


Question 6:

Clean up the country animation. Set the height size of the figure to 1000 to have a better view of the animation

```
In [10]: fig3a = px.bar(df, x='pop', y='country', color='country', animation_frame="year",
                    animation_group = 'country', height = 1000, orientation='h', text_auto=True)
fig3a.update_xaxes(range=(0,200000000))
fig3a.update_yaxes(categoryorder='total descending')
fig3a.update_traces(textfont_size=12, textangle=0, textposition='outside',
                    cliponaxis=False)
fig3a.show()
```





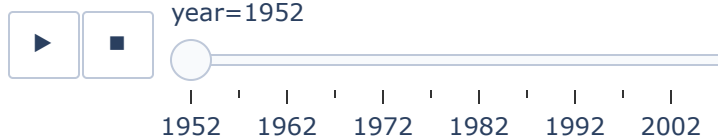
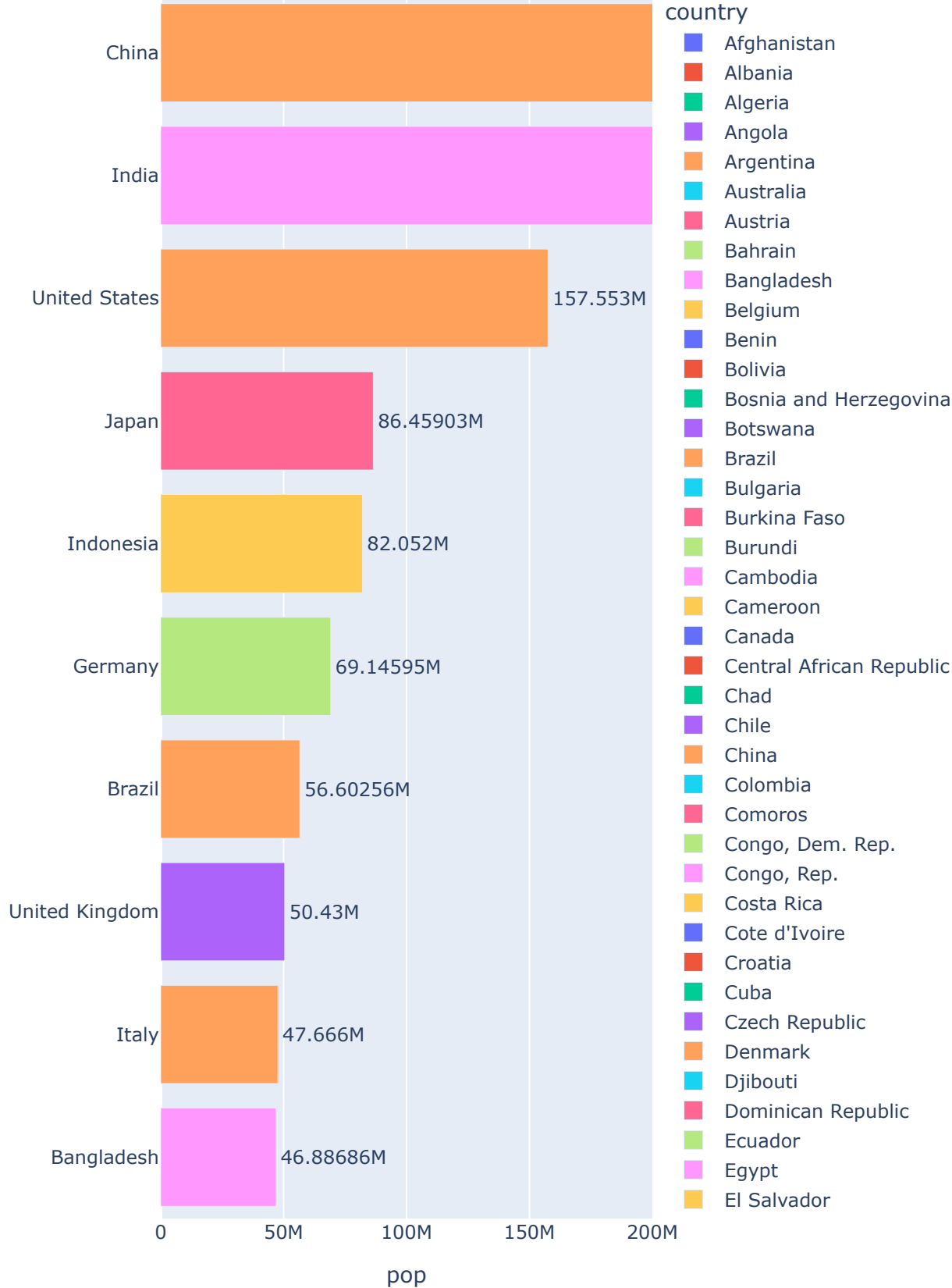
Question 7:

Show only the top 10 countries in the animation

Hint: Use the axis limit to set this.

```
In [11]: fig4 = px.bar(df, x='pop', y='country', color='country', animation_frame="year",
                    animation_group = 'country', height = 1000, orientation='h', text_auto=True)
fig4.update_xaxes(range=(0,200000000))
fig4.update_yaxes(range=(131.5, 141.5))
fig4.update_traces(textfont_size=12, textangle=0, textposition='outside',
                    cliponaxis=False)
fig4.update_layout(yaxis={'categoryorder': 'total ascending'})
fig4.show()
```


country



In []: