

# LabSession\_AdvancedViz

October 3, 2022

## 1 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

```
[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"
```

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

```
[2]:      country continent  year  lifeExp      pop  gdpPercap iso_alpha \
0  Afghanistan      Asia  1952   28.801  8425333  779.445314      AFG
1  Afghanistan      Asia  1957   30.332  9240934  820.853030      AFG
2  Afghanistan      Asia  1962   31.997 10267083  853.100710      AFG
3  Afghanistan      Asia  1967   34.020 11537966  836.197138      AFG
4  Afghanistan      Asia  1972   36.088 13079460  739.981106      AFG

      iso_num
0           4
1           4
2           4
3           4
4           4
```

### 1.1 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

```
[15]: df_2007 = df.query('year==2007').groupby('continent').sum()
fig = px.bar(df_2007, x=df_2007.index, y='pop', color = df_2007.index)
fig.show()
```

## 1.2 Question 2:

Sort the order of the continent for the visualisation

Hint: Use [axis layout setting](#)

```
[16]: fig.update_xaxes(categoryorder = 'total descending')
fig.show()
```

## 1.3 Question 3:

Add text to each bar that represents the population

```
[19]: fig = px.bar(df_2007, x=df_2007.index, y='pop', color = df_2007.index,
    ↪text_auto = '.2s')
fig.update_xaxes(categoryorder='total descending')
fig.update_traces(textposition="outside")
fig.show()
```

## 1.4 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

```
[103]: df_allyear = df.groupby(['continent', 'year']).sum()
df_allyear2 = df_allyear.reset_index(level=[1])
fig =px.bar(df_allyear2, x=df_allyear2.index, y='pop', color=df_allyear2.index,
    ↪text_auto = '.2s', animation_frame='year', range_y=[0,4000000000])
fig.update_xaxes(categoryorder='total descending')
fig.update_traces(textposition="outside")
fig.show()
```

## 1.5 Question 5:

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

```
[41]: fig = px.bar(df, y='country', x='pop', color='country', text_auto = '.2s',
    ↪animation_frame='year', range_x=[0,1500000000])
fig.update_yaxes(categoryorder='total ascending')
fig.update_traces(textposition="outside")
fig.show()
```

## 1.6 Question 6:

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

```
[97]: fig = px.bar(df, y='country', x='pop', color='country', animation_frame='year',  
    ↪range_x=[0,1800000000], height=1000)  
fig.update_yaxes(categoryorder='total ascending')  
fig.show()
```

## 1.7 Question 7:

Show only the top 10 countries in the animation

Hint: Use the axis limit to set this.

```
[98]: fig = px.bar(df, y='country', x='pop', color='country', animation_frame='year',  
    ↪range_x=[0,1500000000], height=1000)  
fig.update_yaxes(categoryorder='total ascending')  
n = len(df.groupby('country').sum())  
fig.update_yaxes(range=[(n-10)-0.5, (n)-0.5])  
fig.update_layout(showlegend=False)  
fig.show()
```