Introduction to plotly

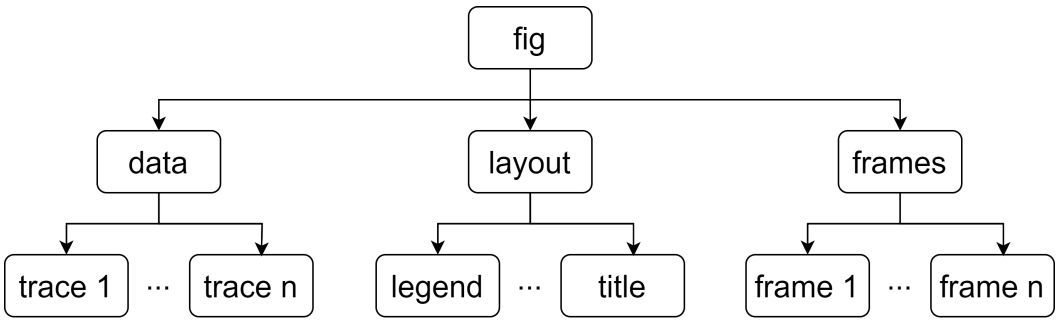
A simple example

In [ ]:
```
import plotly.express as px

# Create a line plot
fig = px.line(x=[1, 2, 3], y=[1, 3, 2])

# Display figure
fig
```

Components of a plotly figure

A plotly figure is a trees of attributes. The root node has three attributes:

- data: list of dictionaries referred to as traces. Each trace has one of more than 40 types (e.g. scatter, bar, pie)
- layout: dictionary that controls positioning and configuration of non-data related parts of the figure (e.g. legend, title)
- frames: list of dictionaries that define sequential frames in an animated plot



In [ ]:
```
print(fig)
```

1) How to create a figure

In [ ]:
```
# Load data
df = px.data.gapminder().query("continent=='Oceania'")

# Create a line plot
px.line(df, x='year', y='lifeExp', color='country')
```

In [ ]:
```
# Load data
df = px.data.gapminder().query("country == 'Canada'")

# Create a bar plot
px.bar(df, x='year', y='pop')
```

In [ ]:
```
# Load data
df = px.data.gapminder().query("year==2007")

# Create a scatter plot
px.scatter(df, x='gdpPercap', y='lifeExp')
```

In [ ]:
```
# Load data
df = px.data.stocks()

# Create area plot
px.area(df, x='date', y=['GOOG', 'AAPL', 'MSFT'])
```

In [ ]:
```
# Create a choropleth map
fig = px.choropleth(locations=['CA', 'AZ', 'TX', 'NY'],
                    scope="usa", locationmode="USA-states",
                    color_continuous_scale='Blues', color=[1, 2, 3, 4])
```

```
# Reduce margins around figure to zero
fig.update_layout(margin={"r":0,"t":0,"l":0,"b":0})
```

2) How to customize a figure

In [ ]:
```
# Load data
df = px.data.gapminder().query("year==2007")

# Size the markers according to feature population
fig = px.scatter(df, x='gdpPercap', y='lifeExp', color='continent', size='pop', size_max=60, hover_data=['country
# Show fig
fig
```

In [ ]:
```
# Update axis labels
fig.update_layout(xaxis = {'title': 'GDP per capita'})
fig.update_layout(yaxis = {'title': 'Life expectancy'})

# Show fig
fig
```

In [ ]:
```
# Add a tile
fig.update_layout(title = 'Comparison of countries')

# Show fig
fig
```

In [ ]:
```
# Customize title of legend
fig.update_layout(legend = {'title': 'Continent'})

# Show fig
fig
```