

CED19I028

ASBD PS - 3

```
In [1]: import random
import plotly.express as px
import plotly.graph_objects as go
import seaborn as sns

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

Question - 1

On New Year's Eve, Tina walked into a random shop and surprised to see a huge crowd there. She is interested to find what kind of products they sell the most, for which she needs the age distribution of customers. Help her to find out the same using histogram. The age details of the customers are given below

7, 9, 27, 28, 55, 45, 34, 65, 54, 67, 34, 23, 24, 66, 53, 45, 44, 88, 22, 33, 55, 35, 33, 37, 47, 41, 31, 30, 29, 12.

```
In [2]: ages = (7,9,27,28,55,45,34,65,54,67,34,23,24,66,53,45,44,88,22,33,55,35,33,37,47,41,31,30,29,12)

plt.figure(figsize=(14,6))

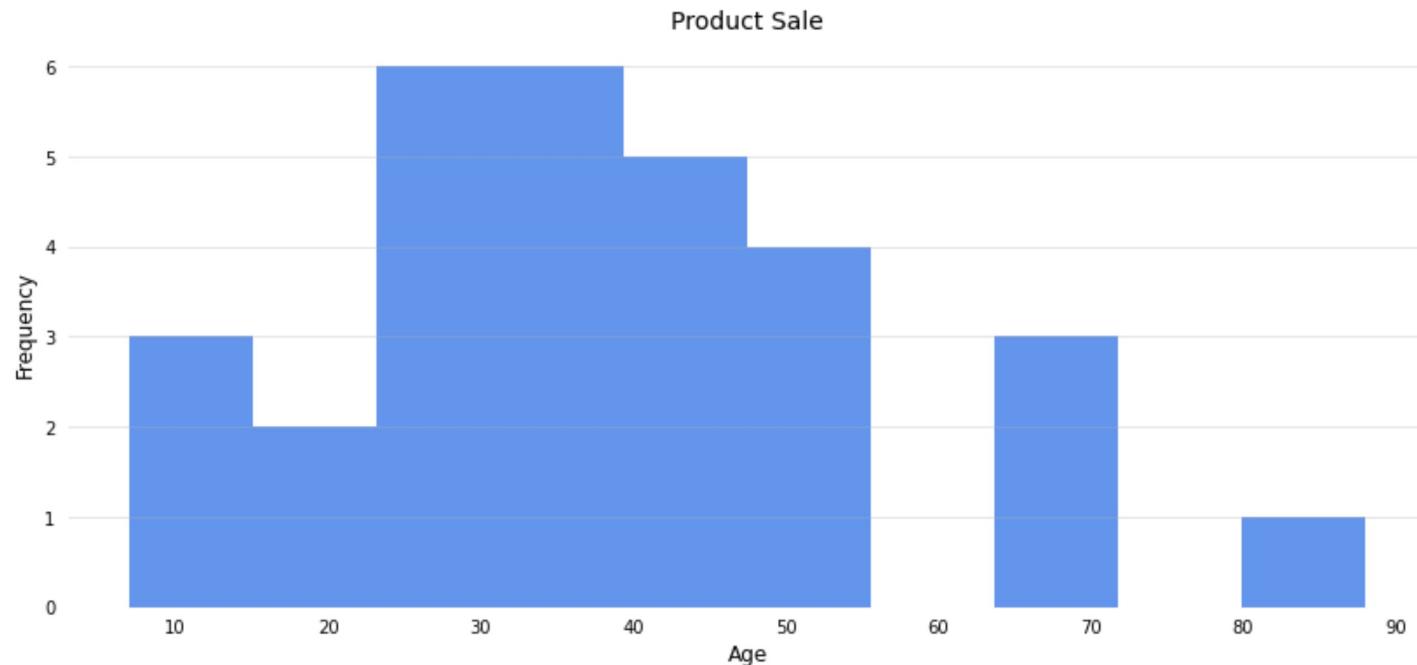
plt.hist(ages, color='cornflowerblue')
plt.xlabel("Age", size=12)
plt.ylabel("Frequency", size=12)

plt.grid(axis = 'y',alpha=0.4)
plt.tick_params(left = False, bottom=False)

for spine in plt.gca().spines.values():
    spine.set_visible(False)

plt.title("Product Sale", size=14)

plt.show()
```



In []:

Question - 2

A Coach tracked the number of points that each of his 30 players on the team had in one game. The points scored by each player is given below. Visualize the data using ordered stem-leaf plot and also detect the outliers and shape of the distribution.

22, 21, 24, 19, 27, 28, 24, 25, 29, 28, 26, 31, 28, 27, 22, 39, 20, 10, 26, 24, 27, 28, 26, 28, 18, 32, 29, 25, 31, 27

```
In [3]: data = (22, 21, 24, 19, 27, 28, 24, 25, 29, 28, 26, 31, 28, 27, 22, 39, 20, 10, 26, 24,
              27, 28, 26, 28, 18, 32, 29, 25, 31, 27)
stems = []
leaves = []

for each in data:
    stems.append(each // 10)
    leaves.append(each % 10)

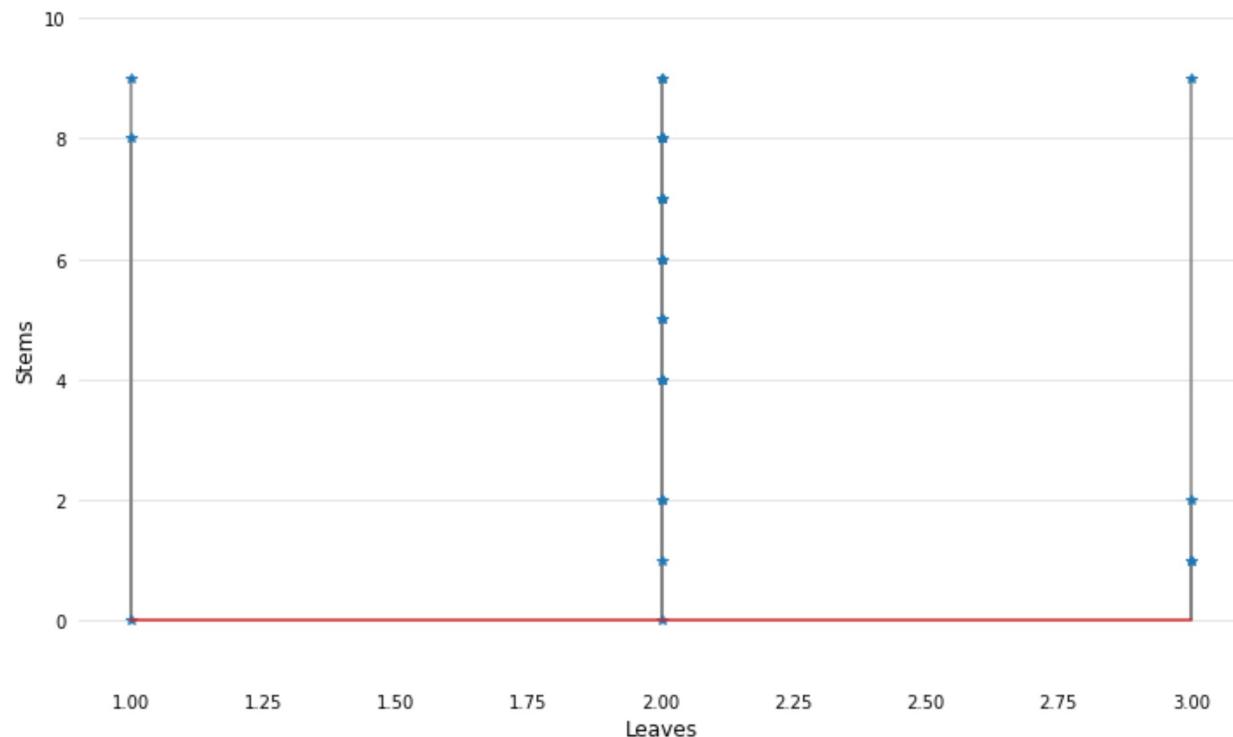
plt.figure(figsize=(12,7))

plt.ylim([-1, 10])
plt.xlabel("Leaves", size=12)
plt.ylabel("Stems", size=12)

plt.grid(axis ='y',alpha=0.4)
plt.tick_params(left = False, bottom=False)

for spine in plt.gca().spines.values():
    spine.set_visible(False)

plt.stem(stems, leaves, linefmt='grey', markerfmt='*')
plt.show()
```



In []:

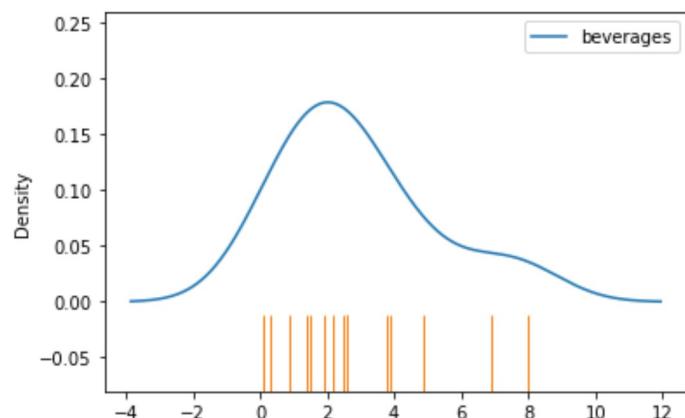
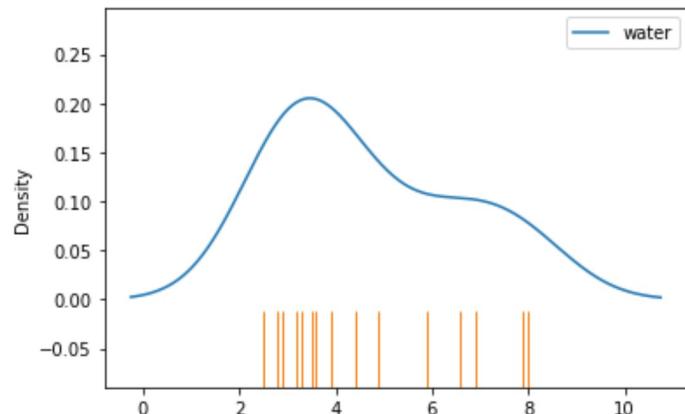
Question - 3

For a sample space of 15 people, a statistician wanted to know the consumption of water and other beverages. He collected their average consumption of water and beverages for 30 days (in litres). Help him to visualize the data using density plot, rug plot and identify the mean, median, mode and skewness of the data from the plot.

```
In [4]: water = [3.2, 3.5, 3.6, 2.5, 2.8, 5.9, 2.9, 3.9, 4.9, 6.9, 7.9, 8.0, 3.3, 6.6, 4.4]
beverages = [2.2, 2.5, 2.6, 1.5, 3.8, 1.9, 0.9, 3.9, 4.9, 6.9, 0.1, 8.0, 0.3, 2.6, 1.4]

df_water = pd.DataFrame(water,columns=["water"])
df_water.plot.density()
sns.rugplot(df_water["water"].tolist(), height=0.2)
plt.show()

df_beverages = pd.DataFrame(beverages, columns=["beverages"])
df_beverages.plot.density()
sns.rugplot(df_beverages["beverages"].tolist(),height=0.2)
plt.show()
```



```
In [ ]:
```

Question - 4

A car company wants to predict how much fuel different cars will use based on their masses. They took a sample of cars, drove each car 100km, and measured how much fuel was used in each case (in litres). Visualize the data using scatterplot and also find co-relation between the 2 variables (eg. Positive//Negative, Linear/ Non-linear co-relation) The data is summarized in the table below.

```
In [5]: fuel = (3.6, 6.7, 9.8, 11.2, 14.7)
mass = (0.45, 0.91, 1.36, 1.81, 2.27)

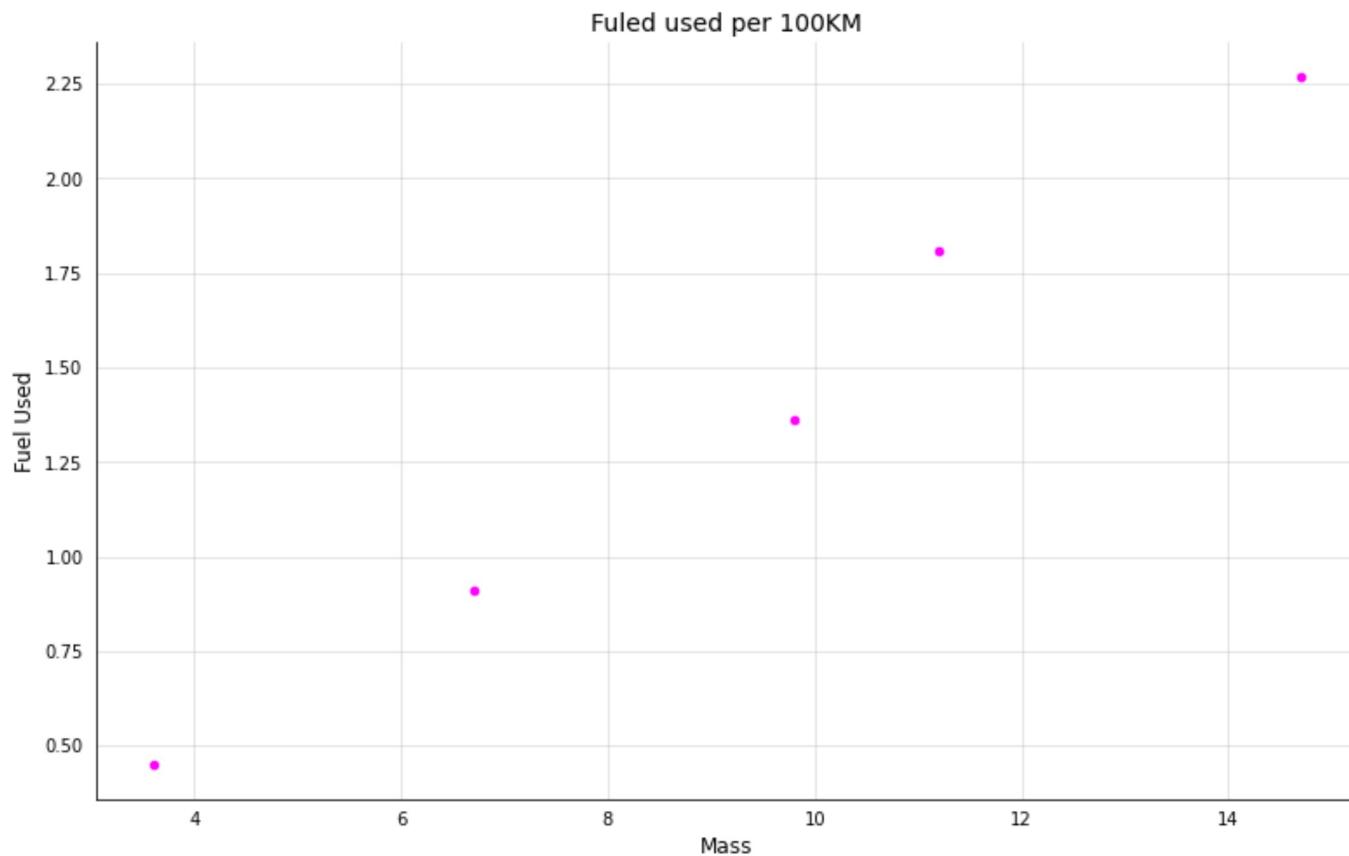
plt.figure(figsize=(13,8))
plt.scatter(fuel, mass, color='magenta', s=20)

plt.ylabel("Fuel Used", size=12)
plt.xlabel("Mass", size=12)

plt.grid(alpha=0.4)
plt.tick_params(left = False, bottom=False)

count = 1;
for spine in plt.gca().spines.values():
    if(count&1):
        pass;
    else:
        spine.set_visible(False)
    count += 1

plt.title("Fuled used per 100KM", size=14)
plt.show()
```



In []:

Question - 5

The data below represents the number of chairs in each class of a government high school. Create a box plot and swarm plot (add jitter) and find the number of data points that are outliers.

35, 54, 60, 65, 66, 67, 69, 70, 72, 73, 75, 76, 54, 25, 15, 60, 65, 66, 67, 69, 70, 72, 130, 73, 75, 76

```
In [6]: chairs_per_class = (35, 54, 60, 65, 66, 67, 69, 70, 72, 73, 75, 76, 54, 25, 15, 60, 65, 66, 67, 69, 70, 72, 130, 73, 75, 76)

plt.boxplot(chairs_per_class)
plt.ylabel("Chairs per class")

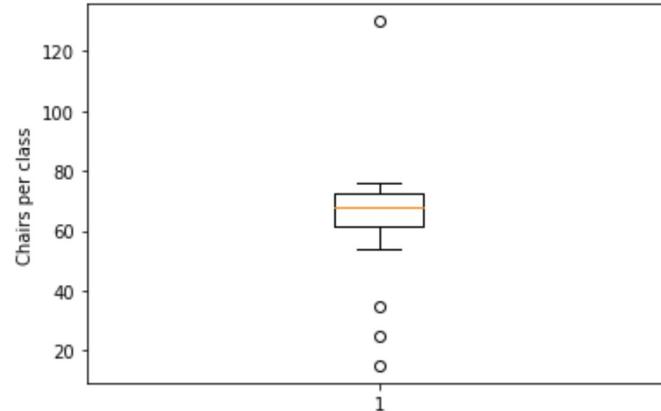
plt.show()

sns.swarmplot(y=chairs_per_class)
plt.ylabel("Chairs per class")

plt.grid(alpha=0.4)
plt.tick_params(left = False, bottom=False)

for spine in plt.gca().spines.values():
    spine.set_visible(False)

plt.show()
```



In []:

Question - 6

Generate random numbers from the following distribution and visualize the data using violin plot.

(i) Standard-Normal distribution.

(ii) Log-Normal distribution.

```
In [7]: a=list(np.random.normal(loc=0,scale=1,size=5))
b=list(np.random.lognormal(mean=0,sigma=1,size=5))

print(a)
print(b)

[-1.4822670753638156, 1.3305540827738502, 0.04857663870875624, -0.5362739715701812, -0.5888838778277624]
[2.3917802307026883, 0.20553046641056658, 0.4295911583772761, 19.647397968121286, 0.22079820227440514]
```

```
In [8]: def plotGraph(data, name):
    sns.violinplot(x = data, scale="count", gridsize=100)

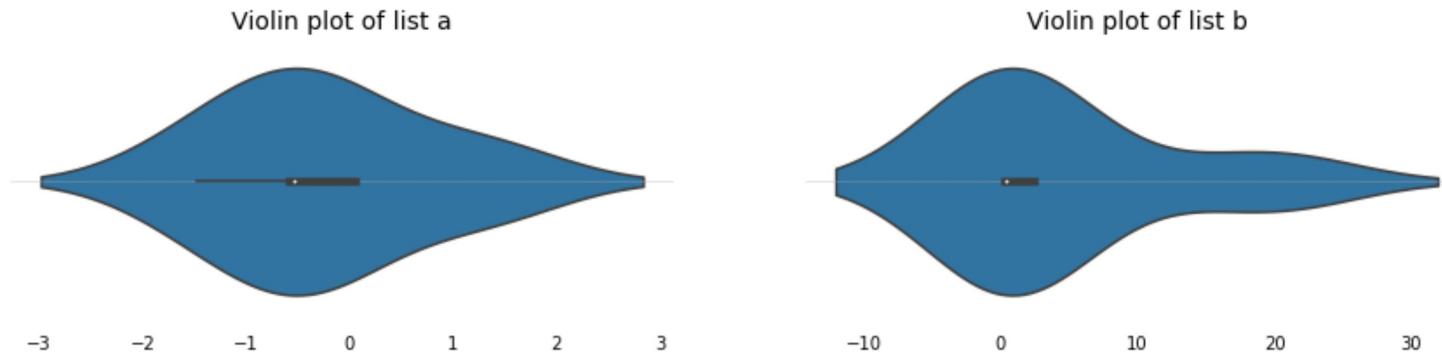
    plt.grid(axis ='y',alpha=0.4)
    plt.tick_params(left = False, bottom=False)

    for spine in plt.gca().spines.values():
        spine.set_visible(False)

    plt.title(f"Violin plot of list {name}", size=14)

plt.figure(figsize=(15,3))
plt.subplot(1, 2, 1)
plotGraph(a, "a");

plt.subplot(1, 2, 2)
plotGraph(b, "b");
```



In []:

Q-7

An Advertisement agency develops new ads for various clients (like Jewellery shops, Textile shops). The Agency wants to assess their performance, for which they want to know the number of ads they developed in each quarter for different shop category. Help them to visualize data using radar/spider charts.

```
In [9]: caterogies = ('Q_1', 'Q_2', 'Q_3', 'Q_4')
textile = (10,6,8,13)
jewellery = (5,5,2,14)
cleaning_essentials = (15,20,16,15)
cosmetics = (14,10,21,11)

fig = go.Figure()

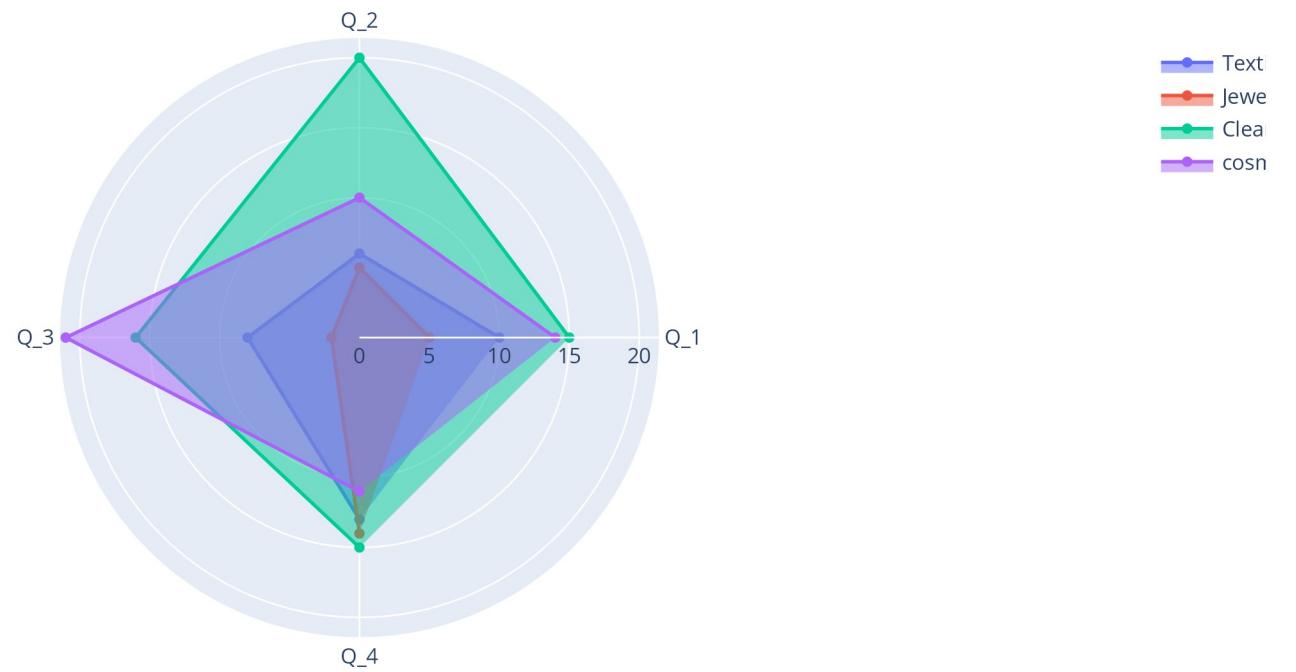
fig.add_trace(go.Scatterpolar(
    r = textile,
    theta = caterogies,
    name='Textile'
))

fig.add_trace(go.Scatterpolar(
    r = jewellery,
    theta = caterogies,
    name='Jewellery'
))

fig.add_trace(go.Scatterpolar(
    r = cleaning_essentials,
    theta = caterogies,
    name='Cleaning Essentials'
))

fig.add_trace(go.Scatterpolar(
    r = cosmetics,
    theta = caterogies,
    name='cosmetics'
))

fig.update_traces(fill='toself')
fig.show()
```



In []:

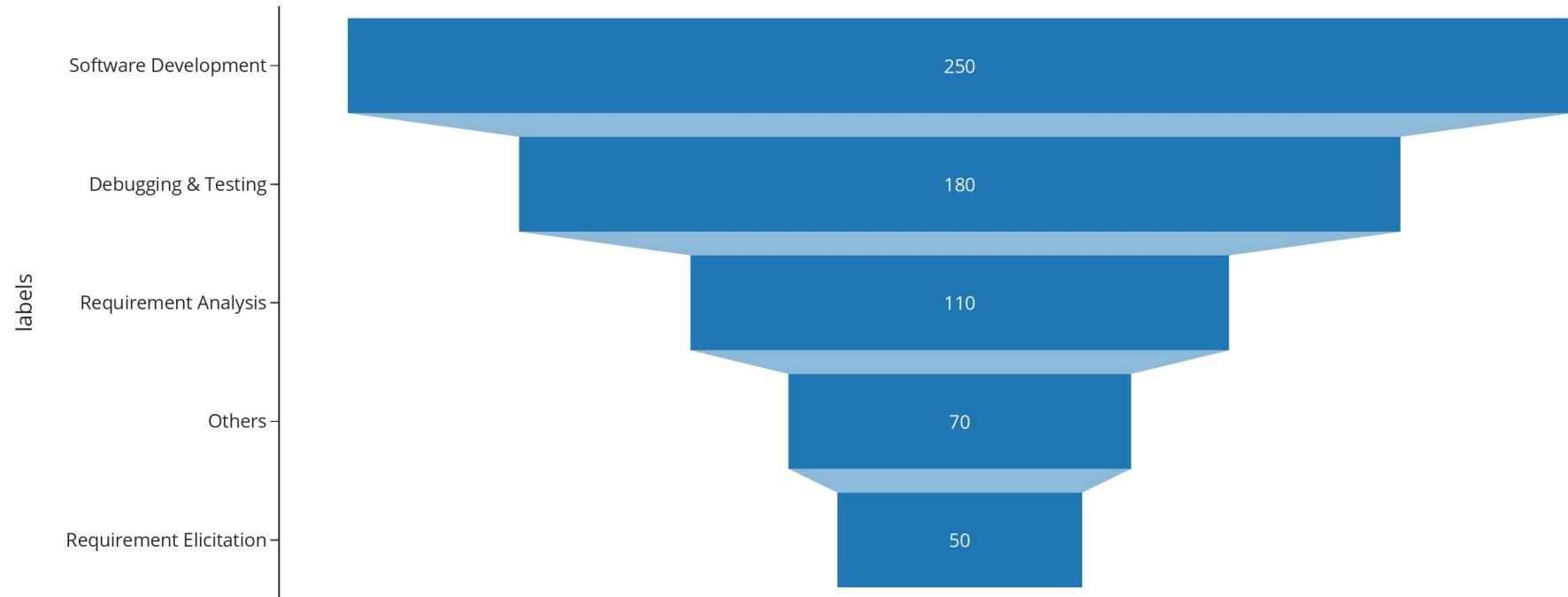
Question 8

An organization wants to calculate the % of time they spent on each process for their product development. Visualize the data using funnel chart with the data given below.

```
In [10]: import plotly.express as px

data = dict(values=[250,180,110,70,50],
            labels = ['Software Development', 'Debugging & Testing', 'Requirement Analysis',
                      'Others', 'Requirement Elicitation'])

fig = px.funnel(data, y='labels', x='values', template="simple_white")
fig.show()
```



```
In [ ]:
```

Question 9

Let's say you are the new owner of a small ice-cream shop in a little village near the beach. You noticed that there was more business in the warmer months than the cooler months. Before you alter your purchasing pattern to match this trend, you want to be sure that the relationship is real. Help him to find the correlation between the data given.

```
In [11]: from scipy.interpolate import make_interp_spline

temp = [98,87,90,85,95,75]
num_of_customers = [15,12,10,10,16,7]

plt.figure(figsize=(13,8))
plt.scatter(temp, num_of_customers, color='blue', s=20)

temp = [75,85,87,90,95,98]
num_of_customers = [7,10,12,10,16,15]

x = np.array(temp)
y = np.array(num_of_customers)

X_Y_Spline = make_interp_spline(x, y)
# Returns evenly spaced numbers
# over a specified interval.
X_ = np.linspace(x.min(), x.max(), 500)
Y_ = X_Y_Spline(X_)

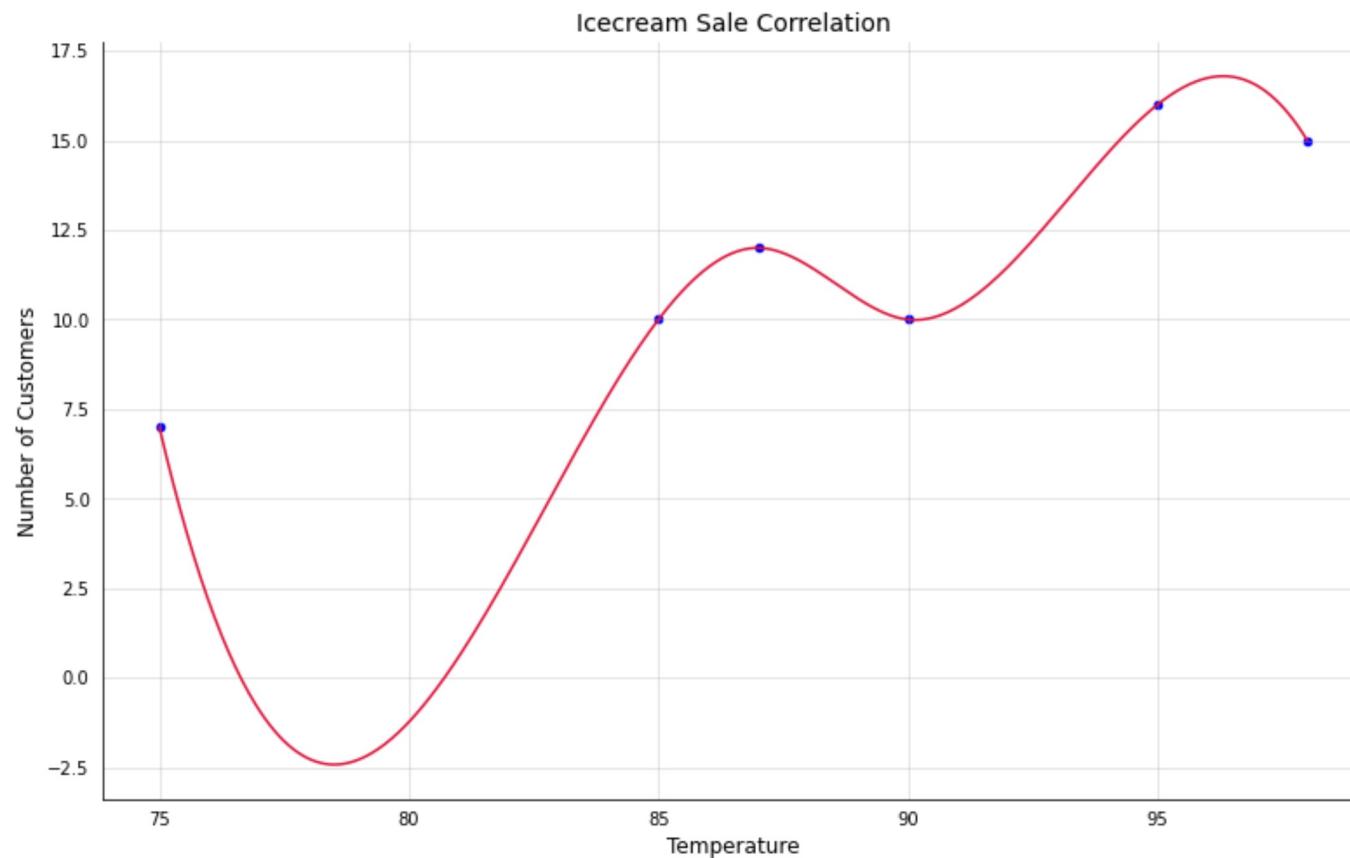
# Plotting the Graph
plt.plot(X_, Y_, color='crimson')

plt.ylabel("Number of Customers", size=12)
plt.xlabel("Temperature", size=12)

plt.grid(alpha=0.4)
plt.tick_params(left = False, bottom=False)

count = 1;
for spine in plt.gca().spines.values():
    if(count&1):
        pass;
    else:
        spine.set_visible(False)
    count += 1

plt.title("Icecream Sale Correlation", size=14)
plt.show()
```



In []:

In []: