1)To draw line charts of employees where hire date between JAN- 2013 to DEC-2020

Ans:

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



data = {
    'ID': [1, 2, 3, 4, 5],
    'Hire Date': ['2013-05-15', '2014-06-20', '2015-07-30', '2016-08-10', '2020-12-25']
}
df = pd.DataFrame(data)
df['Hire Date'] = pd.to_datetime(df['Hire Date'])
filtered_df = df[(df['Hire Date'] >= '2013-01-01') & (df['Hire Date'] <= '2020-12-31')]


filtered_df['Year'] = filtered_df['Hire Date'].dt.year
```

```
hire_counts =
filtered_df['Year'].value_counts().sort_index()


plt.plot(hire_counts.index, hire_counts.values,
marker='o')
plt.title('Employee Hires from 2013 to 2020')
plt.xlabel('Year')
plt.ylabel('Number of Hires')
plt.xticks(hire_counts.index)
plt.grid()
plt.show()
```

output:-

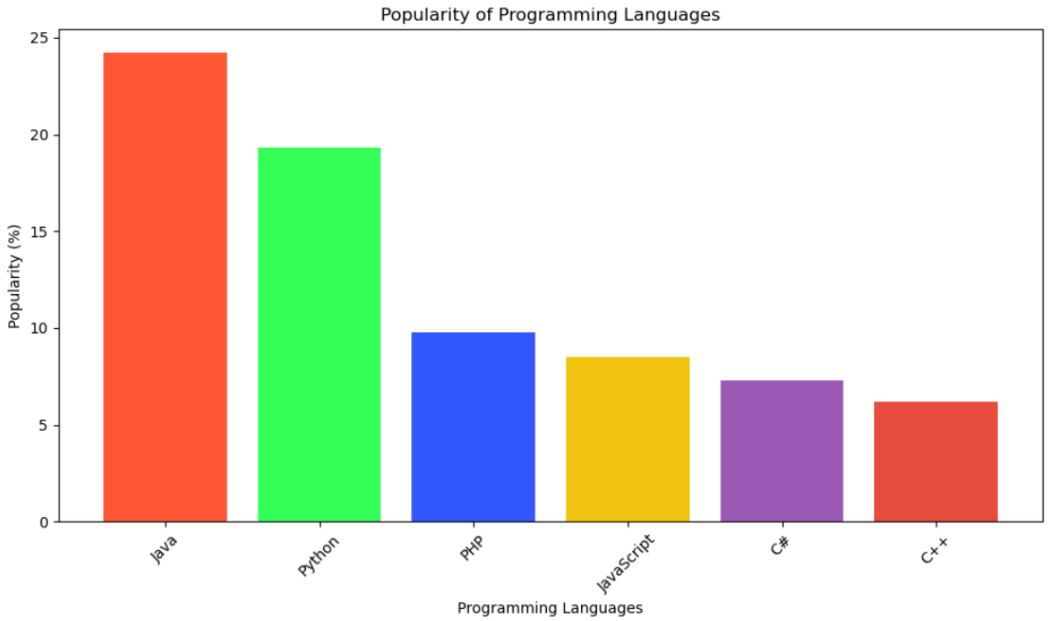
2)To display a bar chart of the popularity of programming Languages. Use different colour for each bar. Programming languages: Java, Python, PHP, JavaScript, C#, C++ Popularity: 24.2, 19.3, 9.8, 8.5, 7.3, 6.2

Ans:

```
import matplotlib.pyplot as plt
```

```
languages = ['Java', 'Python', 'PHP', 'JavaScript', 'C#', 'C++']

popularity = [24.2, 19.3, 9.8, 8.5, 7.3, 6.2]

colors = ['#FF5733', '#33FF57', '#3357FF', '#F1C40F', '#9B59B6', '#E74C3C']


plt.figure(figsize=(10, 6))

plt.bar(languages, popularity, color=colors)

plt.title('Popularity of Programming Languages')

plt.xlabel('Programming Languages')

plt.ylabel('Popularity (%)')

plt.xticks(rotation=45)

plt.tight_layout()

plt.show()
```

OP:



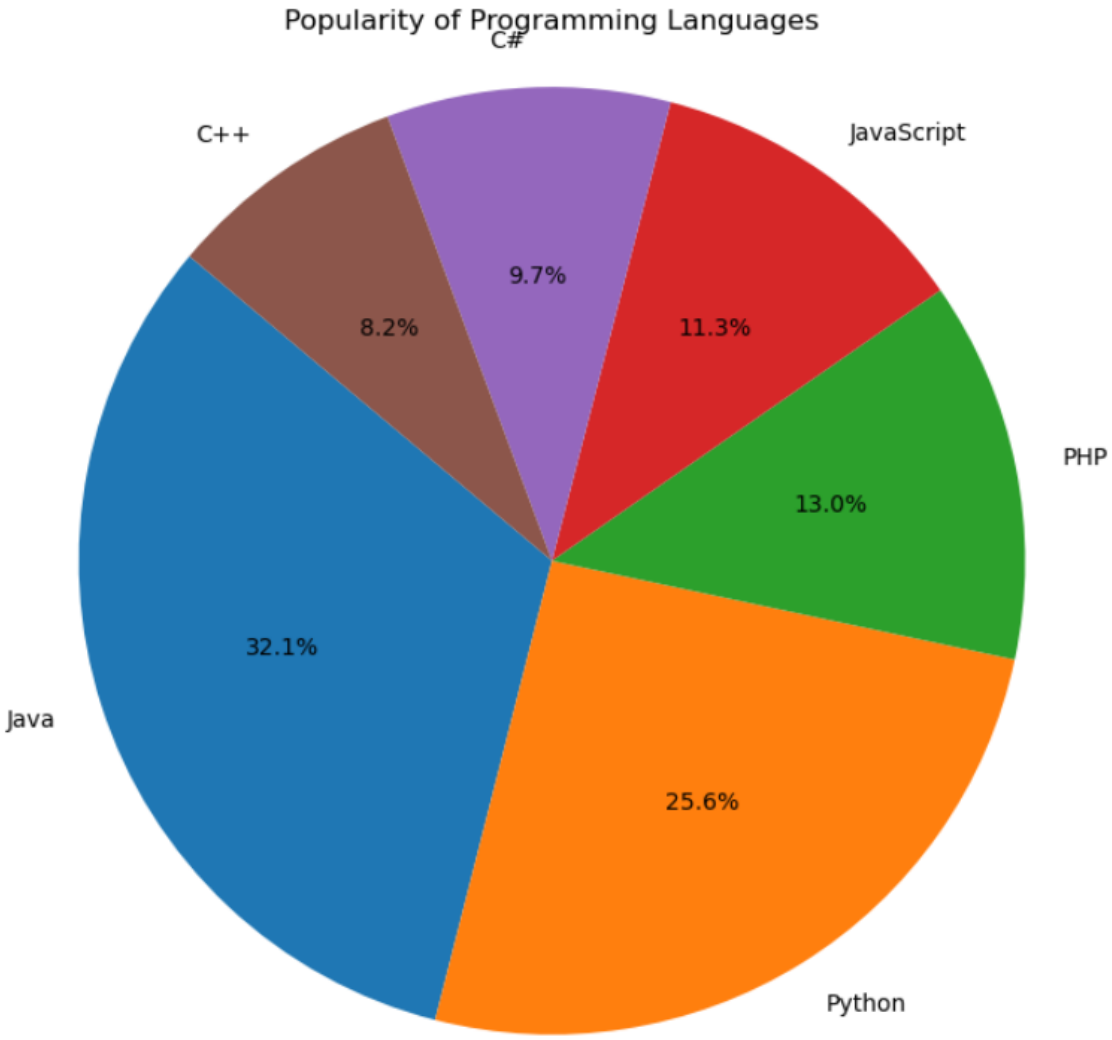
3)To create a pie chart of the popularity of programming Languages.

Ans:

```
import matplotlib.pyplot as plt

languages = ['Java', 'Python', 'PHP', 'JavaScript', 'C#', 'C++']
popularity = [24.2, 19.3, 9.8, 8.5, 7.3, 6.2]

plt.figure(figsize=(8, 8))
```

```
plt.pie(popularity, labels=languages,
autopct='%1.1f%%', startangle=140)
plt.title('Popularity of Programming Languages')
plt.axis('equal')
plt.show()
```

OP:



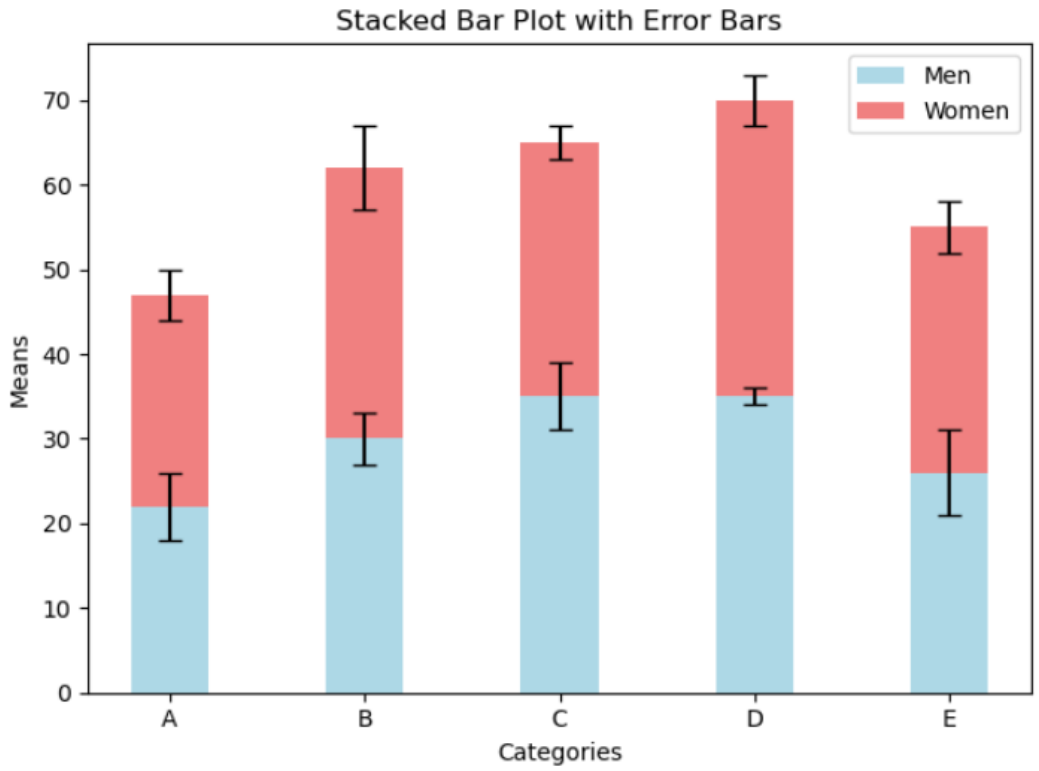
4)To create a stacked bar plot with error bars. Use bottom to stack the womens bars on top of the bar men's. Means (men) = (22, 30, 35, 35, 26) Means (women) = (25, 32, 30, 35, 29) Men Standard deviation = (4, 3, 4, 1, 5) Women Standard deviation = (3, 5, 2, 3, 3)

Ans:

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

categories = ['A', 'B', 'C', 'D', 'E']
men_means = [22, 30, 35, 35, 26]
women_means = [25, 32, 30, 35, 29]
men_std = [4, 3, 4, 1, 5]
women_std = [3, 5, 2, 3, 3]

x = np.arange(len(categories))
bar_width = 0.4
```

```
plt.bar(x, men_means, yerr=men_std,
width=bar_width, label='Men', color='lightblue',
capsize=5)

plt.bar(x, women_means, yerr=women_std,
bottom=men_means, width=bar_width,
label='Women', color='lightcoral', capsize=5)


plt.xlabel('Categories')

plt.ylabel('Means')

plt.title('Stacked Bar Plot with Error Bars')

plt.xticks(x, categories)

plt.legend()

plt.tight_layout()

plt.show()
```

OP:



5)To draw a scatter plot comparing two subject marks of Mathematics and Science. Use marks of 10 students. Data: math_marks = [88, 92, 80, 89, 100, 80, 60, 100, 80, 34] science_marks = [35, 79, 79, 48, 100, 88, 32, 45, 20, 30] marks_range = [10, 20, 30, 40, 50, 60, 70, 80, 90, 100]

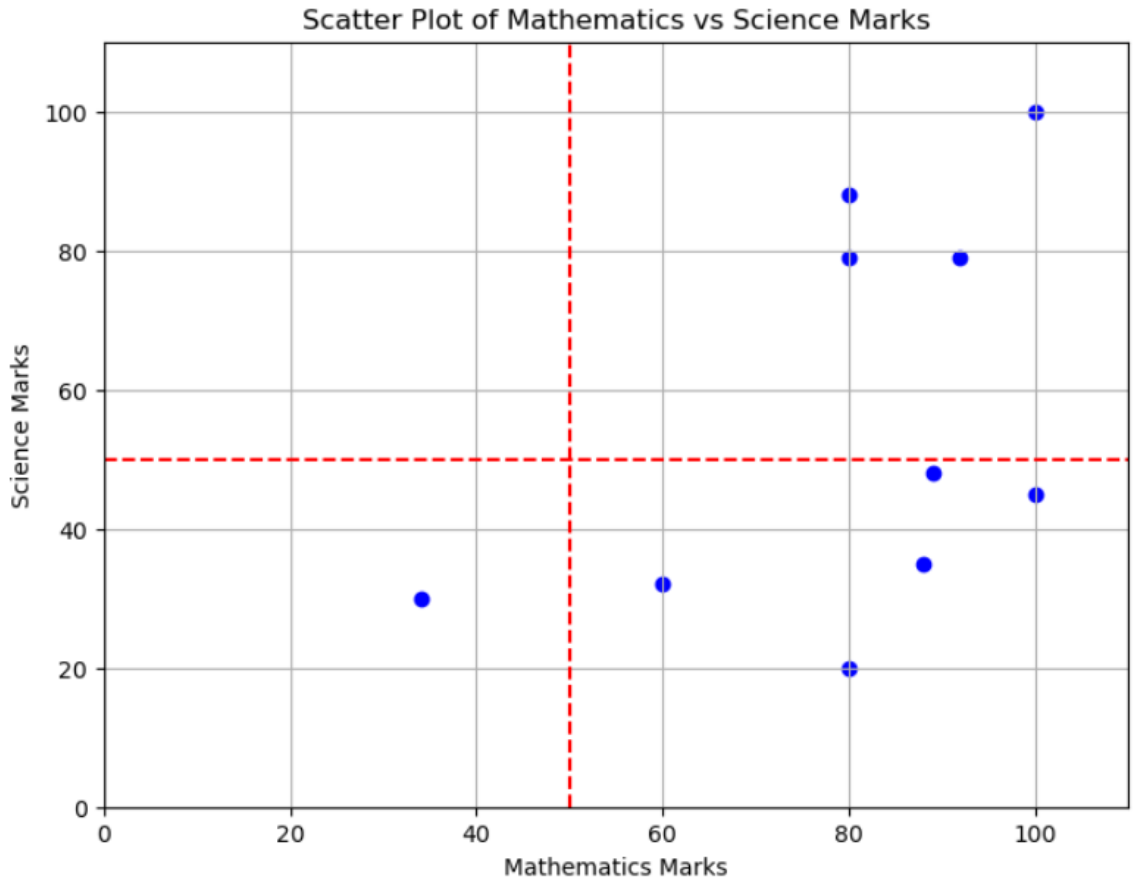
Ans:

```
import matplotlib.pyplot as plt
```

```
math_marks = [88, 92, 80, 89, 100, 80, 60, 100, 80, 34]
science_marks = [35, 79, 79, 48, 100, 88, 32, 45, 20, 30]
marks_range = [10, 20, 30, 40, 50, 60, 70, 80, 90, 100]

plt.figure(figsize=(8, 6))
plt.scatter(math_marks, science_marks, color='blue')
plt.xlabel('Mathematics Marks')
plt.ylabel('Science Marks')
plt.title('Scatter Plot of Mathematics vs Science Marks')
plt.xlim(0, 110)
plt.ylim(0, 110)
plt.grid(True)
plt.axhline(50, color='red', linestyle='--')
plt.axvline(50, color='red', linestyle='--')
```

plt.show()

OP:



6)Solve some following questions based on below table. Get total profit of all months and show line plot with the following Style properties • Line Style dotted and Line-color should be red • Show legend at the lower right location. • X label name = Month Number • Y label name = Sold

units number • Add a circle marker. • Line marker color as read

• Line width should be 3

• Line width should be 3

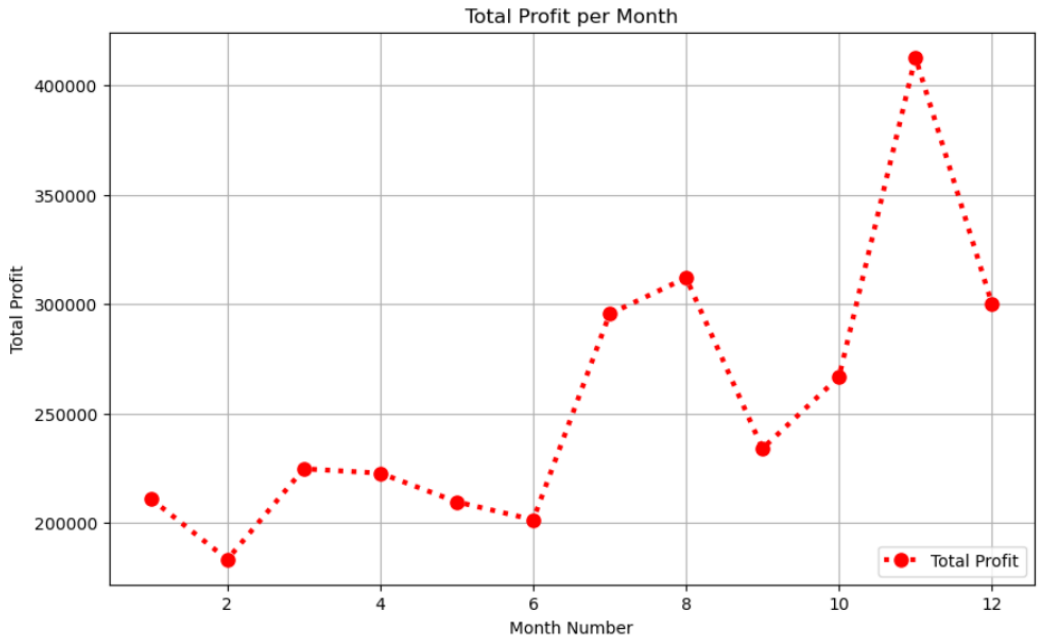
month_number	facecream	facewash	toothpaste	bathingsoap	shampoo	moisturizer	total_units	total_profit
1	2500	1500	5200	9200	1200	1500	21100	211000
2	2630	1200	5100	6100	2100	1200	18330	183300
3	2140	1340	4550	9550	3550	1340	22470	224700
4	3400	1130	5870	8870	1870	1130	22270	222700
5	3600	1740	4560	7760	1560	1740	20960	209600
6	2760	1555	4890	7490	1890	1555	20140	201400
7	2980	1120	4780	8980	1780	1120	29550	295500
8	3700	1400	5860	9960	2860	1400	36140	361400
9	3540	1780	6100	8100	2100	1780	23400	234000
10	1990	1890	8300	10300	2300	1890	26670	266700
11	2340	2100	7300	13300	2400	2100	41280	412800
12	2900	1760	7400	14400	1800	1760	30020	300200

Ans:

```
import matplotlib.pyplot as plt


month_number = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12]

total_profit = [211000, 183300, 224700, 222700, 209600, 201400, 295500, 312000, 234000, 266700, 412800, 300200]


plt.figure(figsize=(10, 6))

plt.plot(month_number, total_profit, linestyle=':', color='red', marker='o', markersize=8,
```

```
        markerfacecolor='red', linewidth=3,
label='Total Profit')


plt.xlabel('Month Number')
plt.ylabel('Total Profit')
plt.title('Total Profit per Month')
plt.legend(loc='lower right')

plt.grid(True)
plt.show()
```

OP:



7)Draw scatter plot of toothpaste sales data of each month

Ans:

toothpaste_sales = [5200, 5100, 4550, 5870, 4560, 7490, 4780, 5860, 6110, 10300, 8300, 7400]

plt.figure(figsize=(10, 6))

plt.scatter(month_number, toothpaste_sales, color='blue', s=100, label='Toothpaste Sales')

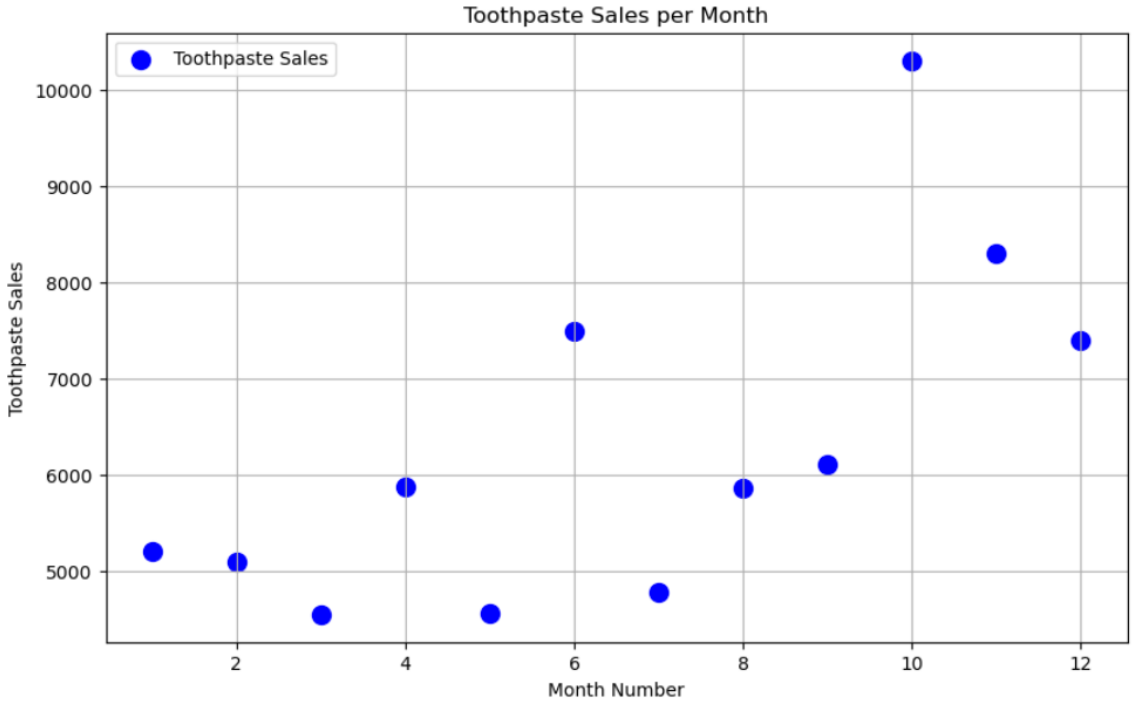
plt.xlabel('Month Number')

plt.ylabel('Toothpaste Sales')

plt.title('Toothpaste Sales per Month')

```
plt.legend()
plt.grid(True)
plt.show()
```

OP:



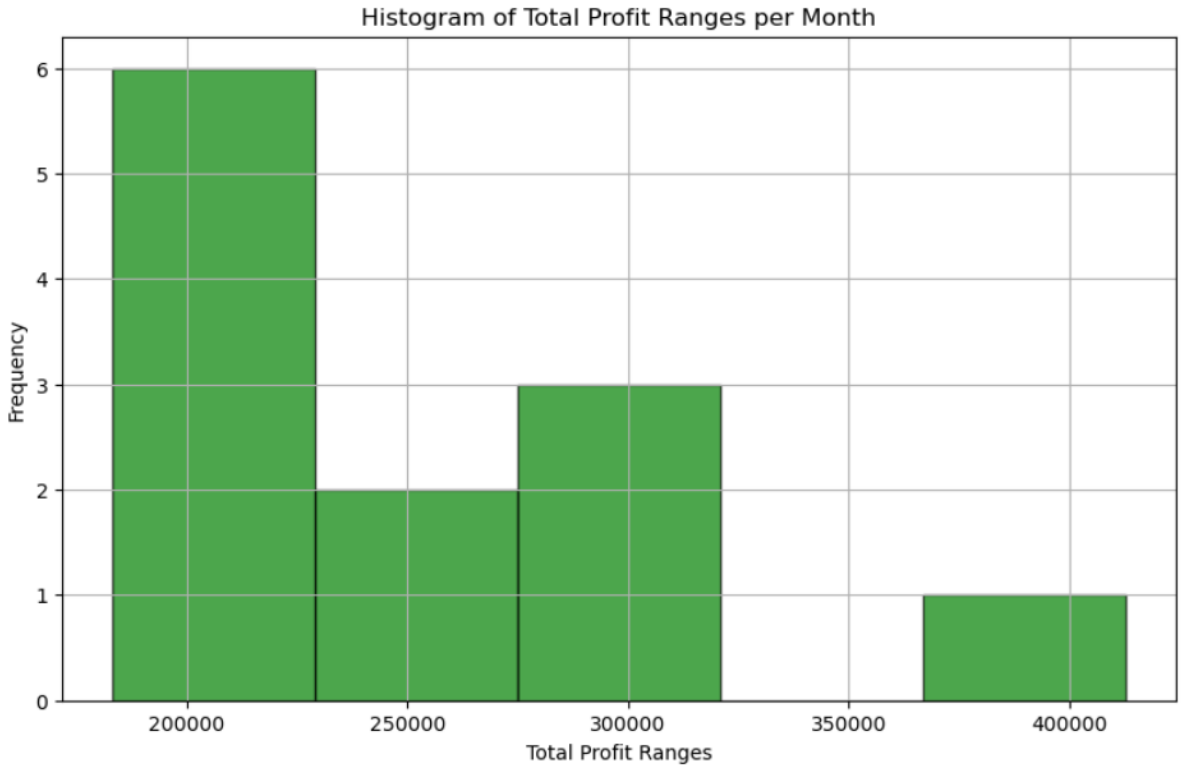
8)Construct histogram of most common profit ranges of the total profit of each month.

Ans:

```
# Create a histogram of total profits
plt.figure(figsize=(10, 6))
plt.hist(total_profit, bins=5, color='green', edgecolor='black', alpha=0.7)
```

```
# Add labels and title
plt.xlabel('Total Profit Ranges')
plt.ylabel('Frequency')
plt.title('Histogram of Total Profit Ranges per Month')

# Display the histogram
plt.grid(True)
plt.show()
```

OP:



9)Draw Pie chart calculate total sale data for last year for each product.

Ans:

facecream_sales = [2500, 2630, 2140, 3400, 3150, 2760, 2980, 3700, 3540, 1990, 2340, 2900]

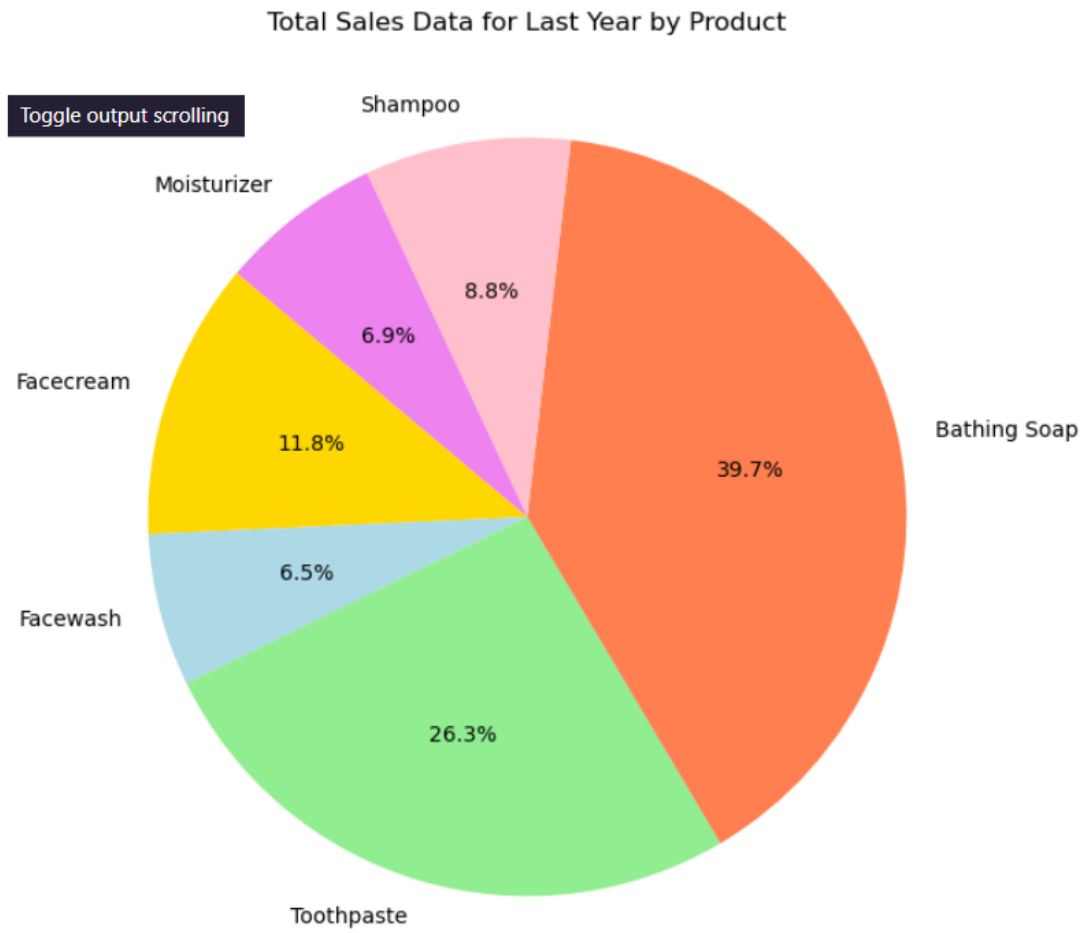
facewash_sales = [1500, 1200, 1340, 1130, 1555, 1555, 1120, 2140, 1780, 1890, 1760, 1760]

toothpaste_sales = [5200, 5100, 4550, 5870, 4560, 7490, 4780, 5860, 6110, 10300, 8300, 7400]

```
bathingsoap_sales = [9200, 6100, 9550, 8870, 7760, 7490, 8980, 9960, 8110, 10300, 13300, 14400]

shampoo_sales = [1200, 2100, 3550, 1870, 1560, 1740, 1780, 2860, 2110, 2300, 2400, 1800]

moisturizer_sales = [1500, 1200, 1340, 1870, 1560, 1740, 1120, 2090, 1780, 1890, 2100, 1760]


total_sales = [
    sum(facecream_sales),
    sum(facewash_sales),
    sum(toothpaste_sales),
    sum(bathingsoap_sales),
    sum(shampoo_sales),
    sum(moisturizer_sales)
]
```

```
products = ['Facecream', 'Facewash',
'Toothpaste', 'Bathing Soap', 'Shampoo',
'Moisturizer']


plt.figure(figsize=(8, 8))
plt.pie(total_sales, labels=products,
autopct='%1.1f%%', startangle=140,
colors=['gold', 'lightblue', 'lightgreen', 'coral',
'pink', 'violet'])


plt.title('Total Sales Data for Last Year by
Product')


plt.show()
```

OP:



10) Draw stack plot of all product sales data.

Ans:

```
import matplotlib.pyplot as plt

month_number = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12]

facecream_sales = [2500, 2630, 2140, 3400, 3150, 2760, 2980, 3700, 3540, 1990, 2340, 2900]
```

facewash_sales = [1500, 1200, 1340, 1130, 1555, 1555, 1120, 2140, 1780, 1890, 1760, 1760]

toothpaste_sales = [5200, 5100, 4550, 5870, 4560, 7490, 4780, 5860, 6110, 10300, 8300, 7400]

bathingsoap_sales = [9200, 6100, 9550, 8870, 7760, 7490, 8980, 9960, 8110, 10300, 13300, 14400]

shampoo_sales = [1200, 2100, 3550, 1870, 1560, 1740, 1780, 2860, 2110, 2300, 2400, 1800]

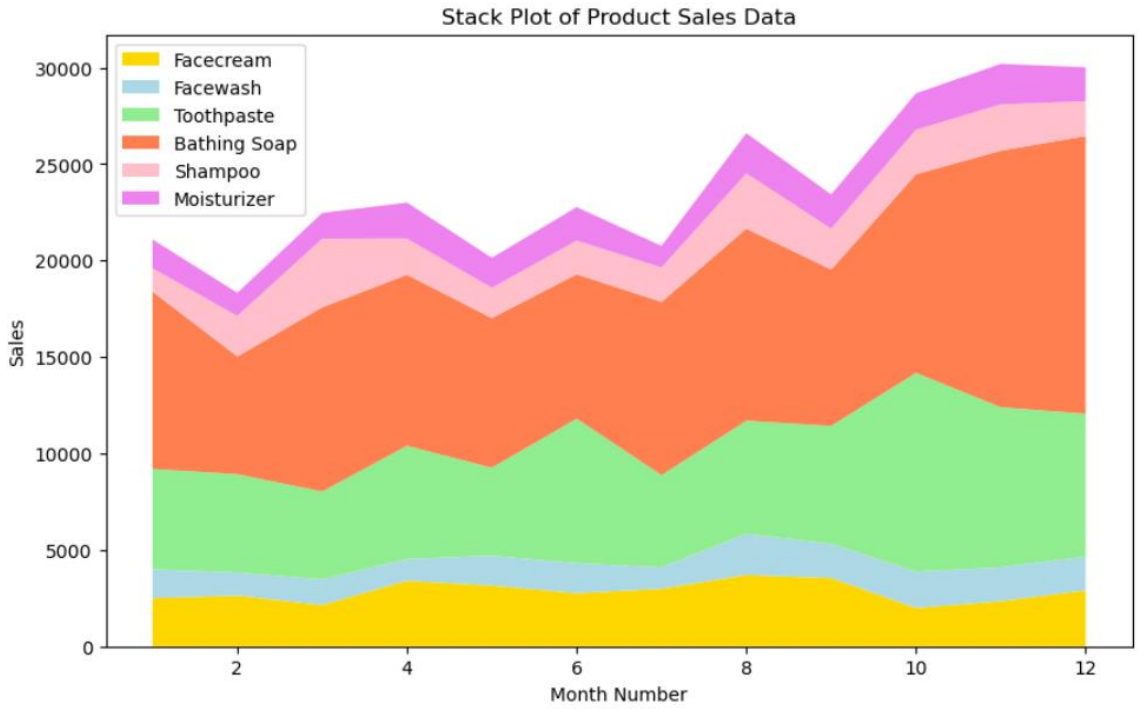
moisturizer_sales = [1500, 1200, 1340, 1870, 1560, 1740, 1120, 2090, 1780, 1890, 2100, 1760]

plt.figure(figsize=(10, 6))

plt.stackplot(month_number, facecream_sales, facewash_sales, toothpaste_sales, bathingsoap_sales,

```
        shampoo_sales, moisturizer_sales,
labels=['Facecream', 'Facewash', 'Toothpaste',
'Bathing Soap', 'Shampoo', 'Moisturizer'],
colors=['gold', 'lightblue', 'lightgreen', 'coral',
'pink', 'violet'])

plt.xlabel('Month Number')
plt.ylabel('Sales')
plt.title('Stack Plot of Product Sales Data')
plt.legend(loc='upper left')

plt.show()
```

OP:



11) Construct Subplot of Bathing soap face wash of all months

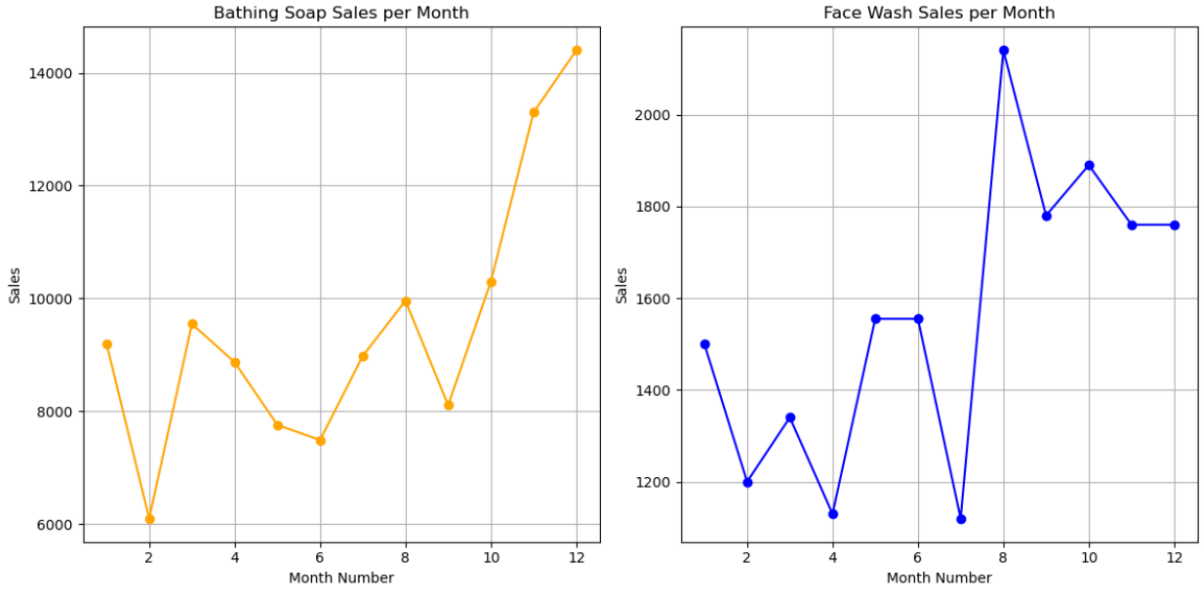
Ans:

```
import matplotlib.pyplot as plt

month_number = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12]

bathingsoap_sales = [9200, 6100, 9550, 8870, 7760, 7490, 8980, 9960, 8110, 10300, 13300, 14400]
```

```
facewash_sales = [1500, 1200, 1340, 1130,
1555, 1555, 1120, 2140, 1780, 1890, 1760,
1760]

fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(12, 6))

ax1.plot(month_number, bathingsoap_sales,
color='orange', marker='o')
ax1.set_title('Bathing Soap Sales per Month')
ax1.set_xlabel('Month Number')
ax1.set_ylabel('Sales')
ax1.grid(True)

ax2.plot(month_number, facewash_sales,
color='blue', marker='o')
ax2.set_title('Face Wash Sales per Month')
ax2.set_xlabel('Month Number')
ax2.set_ylabel('Sales')
ax2.grid(True)
```

```
plt.tight_layout()

plt.show()
```

OP:



12) Draw box plot and violin plot.

Ans:

```
import matplotlib.pyplot as plt

products = ['Facecream', 'Facewash',
'Toothpaste', 'Bathing Soap', 'Shampoo',
'Moisturizer']
```

facecream_sales = [2500, 2630, 2140, 3400, 3150, 2760, 2980, 3700, 3540, 1990, 2340, 2900]

facewash_sales = [1500, 1200, 1340, 1130, 1555, 1555, 1120, 2140, 1780, 1890, 1760, 1760]

toothpaste_sales = [5200, 5100, 4550, 5870, 4560, 7490, 4780, 5860, 6110, 10300, 8300, 7400]

bathingsoap_sales = [9200, 6100, 9550, 8870, 7760, 7490, 8980, 9960, 8110, 10300, 13300, 14400]

shampoo_sales = [1200, 2100, 3550, 1870, 1560, 1740, 1780, 2860, 2110, 2300, 2400, 1800]

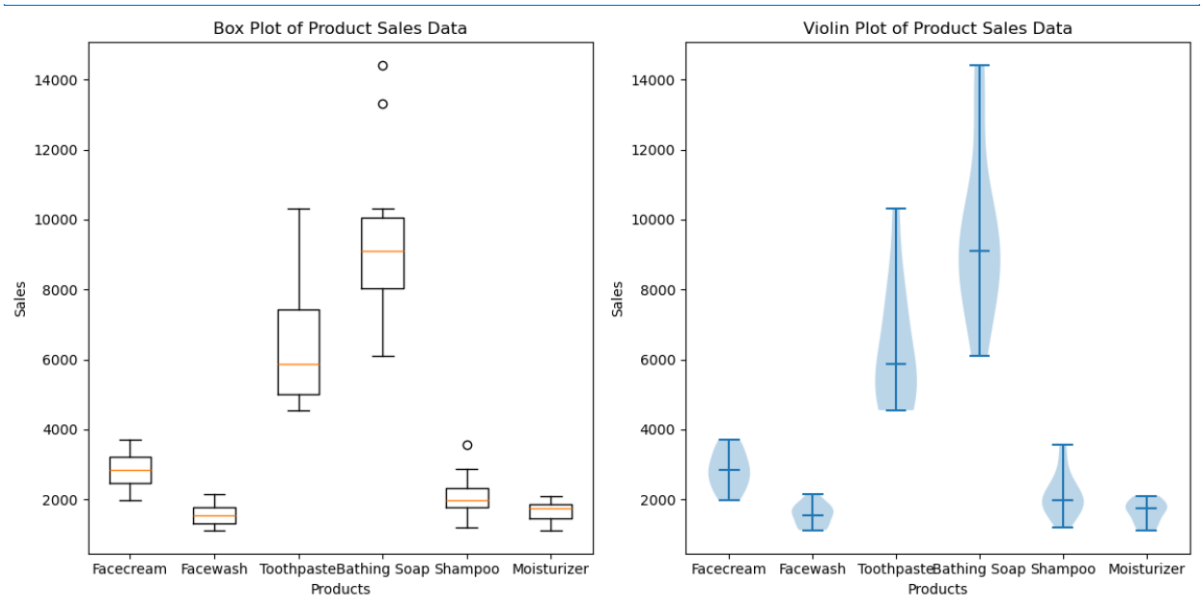
moisturizer_sales = [1500, 1200, 1340, 1870, 1560, 1740, 1120, 2090, 1780, 1890, 2100, 1760]

sales_data = [facecream_sales, facewash_sales, toothpaste_sales, bathingsoap_sales, shampoo_sales, moisturizer_sales]

```
plt.figure(figsize=(12, 6))

plt.subplot(1, 2, 1)
plt.boxplot(sales_data, labels=products)
plt.title('Box Plot of Product Sales Data')
plt.xlabel('Products')
plt.ylabel('Sales')

plt.subplot(1, 2, 2)
plt.violinplot(sales_data, showmeans=False, showmedians=True)
plt.title('Violin Plot of Product Sales Data')
plt.xticks([1, 2, 3, 4, 5, 6], products)  # Setting x-ticks to match the product labels
plt.xlabel('Products')
plt.ylabel('Sales')

plt.tight_layout()
```

plt.show()

OP:


Box Plot of Product Sales Data
Sales
14000
12000
10000
8000
6000
4000
2000
Facecream
Facewash
Toothpaste
Bathing Soap
Shampoo
Moisturizer
Products
Violin Plot of Product Sales Data
Sales
14000
12000
10000
8000
6000
4000
2000
Facecream
Facewash
Toothpaste
Bathing Soap
Shampoo
Moisturizer
Products