PYTHON LAB – 24

MATPLOTLIB MORE PLOTS AND FORMATTING

NAME: Keerthana K R

ID: AF0363623

QUESTIONS

1. Analyze the relationship between the size of houses (measured in square footage) and their selling prices in a particular neighborhood. You have collected data on various houses in that neighborhood.Create a scatter plot using the below data and share your conclusion/analysis.
2. Create a pie chart to visualize the distribution of your monthly income by source. You have collected data on the various sources of your income, such as salary, freelance work, investments, and rental income. Share your conclusion/analysis.
3. Create a pie chart to illustrate the distribution of a company's revenue across its various business segments. You have collected data on the revenue generated by each segment, such as Product A, Product B, Services, and Licensing. Share your conclusion/analysis.
4. Suppose you're a sales manager for an e-commerce company, and you want to create a figure with subplots to compare the sales performance of different product categories over time. You have sales data for four product categories: Electronics, Clothing, Home & Garden, and Sports & Outdoors. Share your conclusion/analysis.

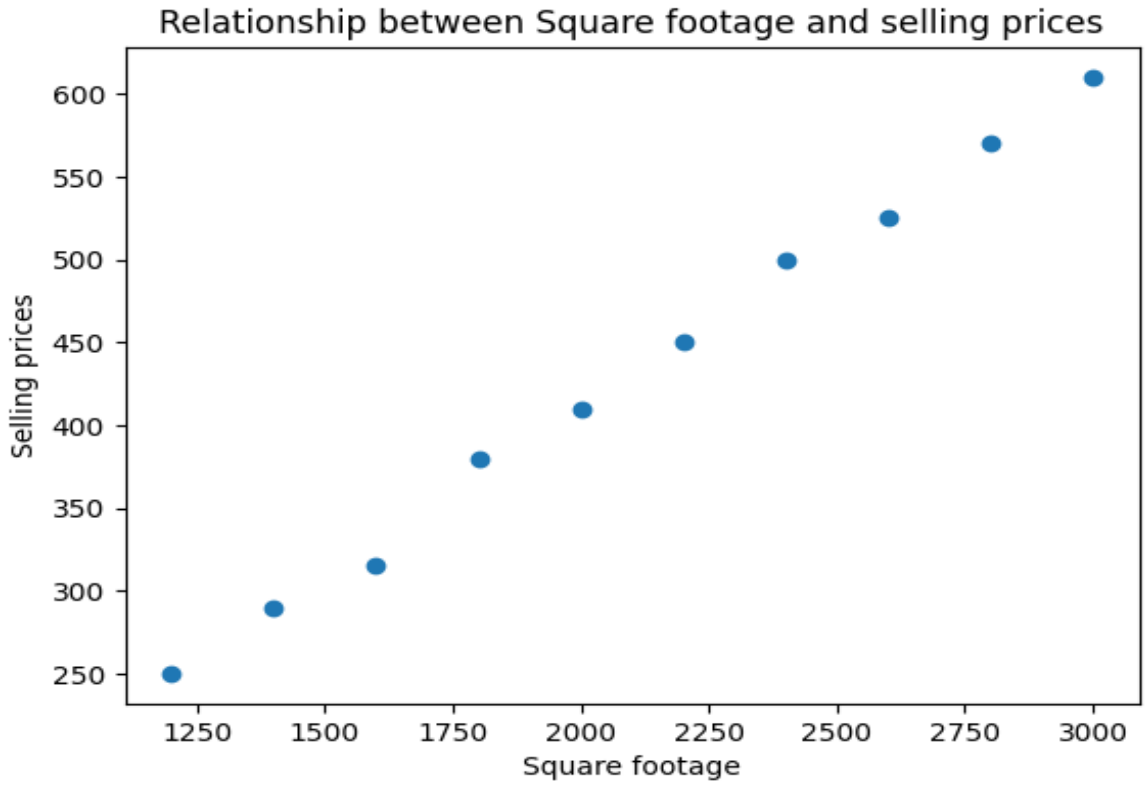
i. Analyze the relationship between the size of houses (measured in square footage) and their selling prices in a particular neighborhood. You have collected data on various houses in that neighborhood.Create a scatter plot using the below data and share your conclusion/analysis.

Input: square_footage = np.array([1200, 1400, 1600, 1800, 2000, 2200, 2400, 2600, 2800, 3000]) selling_prices = np.array([250, 290, 315, 380, 410, 450, 500, 525, 570, 610])

Code :

```
# import necessary packages
import numpy as np
import matplotlib.pyplot as plt

# Input data
square_footage = np.array([1200, 1400, 1600, 1800, 2000,
2200, 2400, 2600, 2800, 3000])
selling_prices = np.array([250, 290, 315, 380, 410, 450,
500, 525, 570, 610])

# Create scatter plot
plt.scatter(square_footage,selling_prices)

# Add title and labels
plt.title('Relationship between Square footage and
selling prices')
plt.xlabel('Square footage')
plt.ylabel('Selling prices')

# Displaying plot
plt.show()
```

Output:

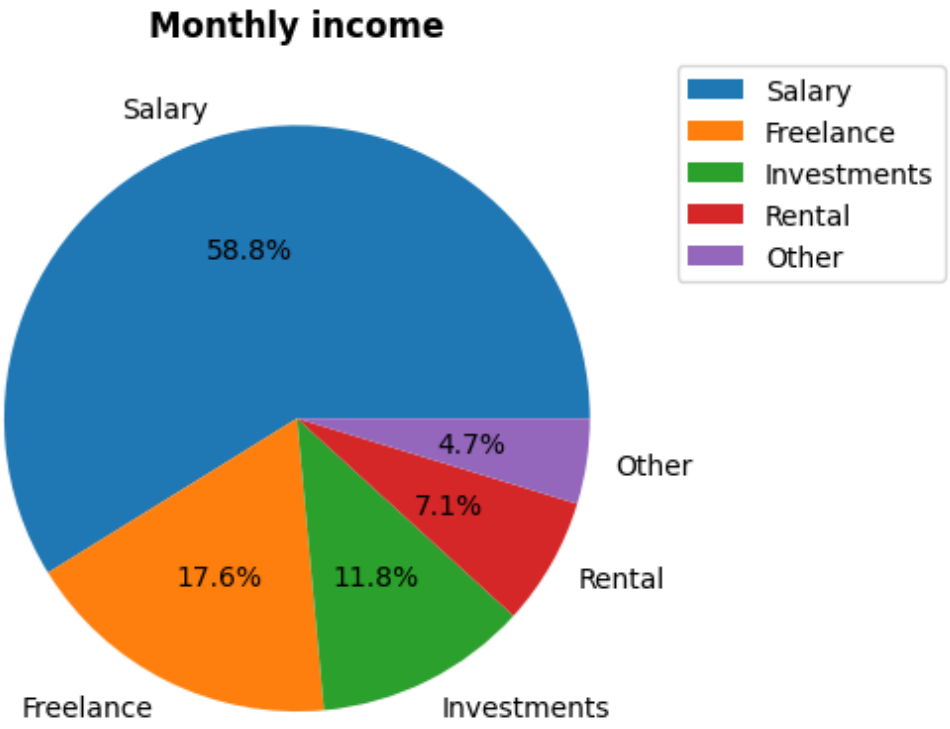

Relationship between Square footage and selling prices
Selling prices
600
550
500
450
400
350
300
250
1250
1500
1750
2000
2250
2500
2750
3000
Square footage

ii. Create a pie chart to visualize the distribution of your monthly income by source. You have collected data on the various sources of your income, such as salary, freelance work, investments, and rental income. Share your conclusion/analysis.
Input: income_sources = ['Salary', 'Freelance', 'Investments', 'Rental', 'Other']
monthly_income = [5000, 1500, 1000, 600, 400]

Code:

```
# import necessary packages
import matplotlib.pyplot as plt

# Input data
income_sources = ['Salary', 'Freelance', 'Investments',
'Rental', 'Other']
monthly_income = [5000, 1500, 1000, 600, 400]

# Create pie chart
plt.pie(monthly_income,labels=income_sources,autopct='%1.
1f%%')

# Add title and legend
plt.title('Monthly income',weight='bold')
plt.legend(loc='best',bbox_to_anchor=(1,1))

# Display the plot
plt.show()
```

Output:

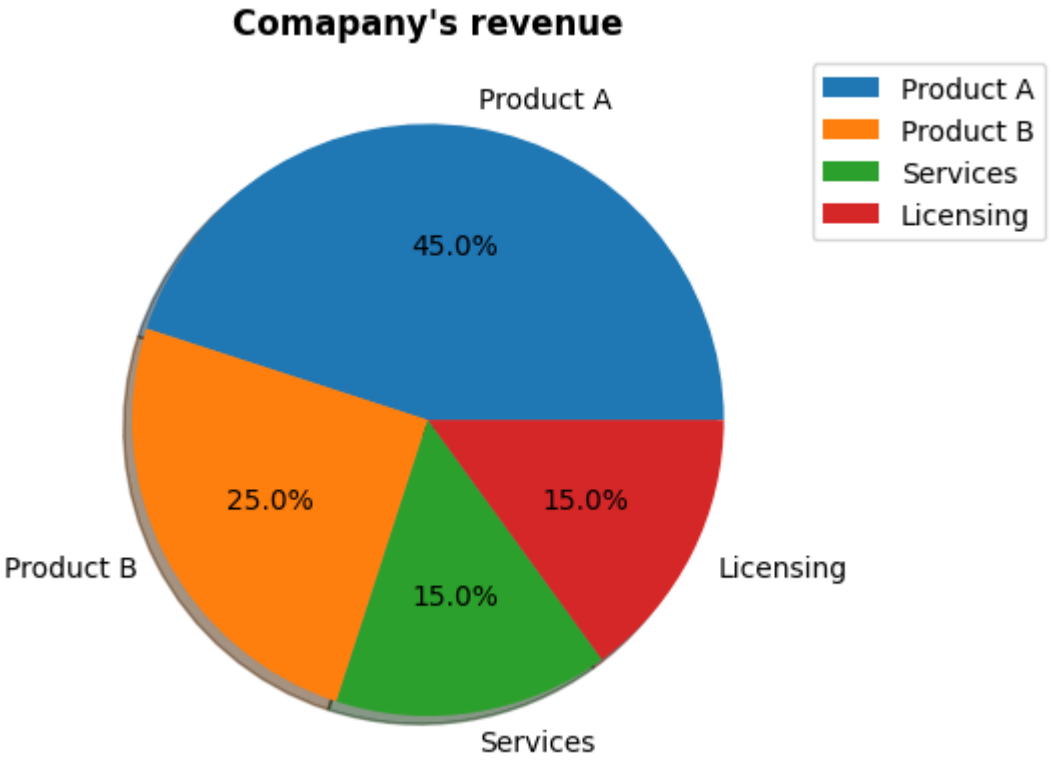

Monthly income
Salary
Freelance
Investments
Rental
Other
58.8%
17.6%
11.8%
7.1%
4.7%
Salary
Freelance
Investments
Rental
Other

3. Create a pie chart to illustrate the distribution of a company's revenue across its various business segments. You have collected data on the revenue generated by each segment, such as Product A, Product B, Services, and Licensing. Share your conclusion/analysis.
   Input: segments = ['Product A', 'Product B', 'Services', 'Licensing'] revenue_percentages = [45, 25, 15, 15]

Code:

```
# import necessary packages
import matplotlib.pyplot as plt

# Input data
segments = ['Product A', 'Product B', 'Services',
'Licensing']
revenue_percentages = [45, 25, 15, 15]

# Create pie chart
plt.pie(revenue_percentages,labels=segments,autopct='%1.1
f%%',shadow=True)

# Add title and legend
plt.title("Comapany's revenue",weight='bold')
plt.legend(loc='best',bbox_to_anchor=(1,1))

# Display plot
plt.show()
```

Output:


Comapany's revenue
Product A
Product B
Services
Licensing
45.0%
25.0%
15.0%
15.0%
Product A
Product B
Licensing
Services

4. Suppose you're a sales manager for an e-commerce company, and you want to create a figure with subplots to compare the sales performance of different product categories over time. You have sales data for four product categories: Electronics, Clothing, Home & Garden, and Sports & Outdoors. Share your conclusion/analysis.
   Input: months = np.arange(1, 13) electronics_sales = np.array([25000, 28000, 31000, 27000, 30000, 32000, 35000, 36000, 38000, 39000, 41000, 42000]) clothing_sales = np.array([15000, 16000, 17000, 18000, 19000, 20000, 21000, 22000, 23000, 24000, 25000, 26000]) home_garden_sales = np.array([18000, 19000, 20000, 21000, 22000, 23000, 24000, 25000, 26000, 27000, 28000, 29000]) sports_outdoors_sales = np.array([12000, 13000, 14000, 15000, 16000, 17000, 18000, 19000, 20000, 21000, 22000, 23000])

Code:

```
# Importing the necessary packages
import numpy as np
import matplotlib.pyplot as plt

# Input data using np.random
months = np.arange(1, 13)
electronics_sales = np.random.randint(25000, 42000,
size=12)
clothing_sales = np.random.randint(15000, 26000, size=12)
home_garden_sales = np.random.randint(18000, 29000,
size=12)
sports_outdoors_sales = np.random.randint(12000, 23000,
size=12)

# Create subplots
fig, axs = plt.subplots(2, 2)

# Plotting sales data for each category
axs[0, 0].plot(months, electronics_sales,
label='Electronics',color='red')
axs[0, 0].set_title('Electronics Sales')
axs[0, 0].set_xlabel('Month')
axs[0, 0].set_ylabel('Sales')

axs[0, 1].plot(months, clothing_sales,
label='Clothing',color='green')
axs[0, 1].set_title('Clothing Sales')
axs[0, 1].set_xlabel('Month')
axs[0, 1].set_ylabel('Sales')

axs[1, 0].plot(months, home_garden_sales, label='Home &
Garden',color='orange')
```

```
axs[1, 0].set_title('Home & Garden Sales')
axs[1, 0].set_xlabel('Month')
axs[1, 0].set_ylabel('Sales')

axs[1, 1].plot(months, sports_outdoors_sales,
label='Sports & Outdoors',color='blue')
axs[1, 1].set_title('Sports & Outdoors Sales')
axs[1, 1].set_xlabel('Month')
axs[1, 1].set_ylabel('Sales')

# Adjust layout
plt.tight_layout()

# Displaying plot
plt.show()
```

Output: