```
[5]:  %pip install seaborn
      %pip install folium
```

Importing Required Libraries

We recommend you import all required libraries in one place (here):

```
[6]:  import numpy as np
      import pandas as pd
      %matplotlib inline
      import matplotlib as mpl
      import matplotlib.pyplot as plt
      import seaborn as sns
      import folium
```

For your convenience, we have already written code to import the data below.
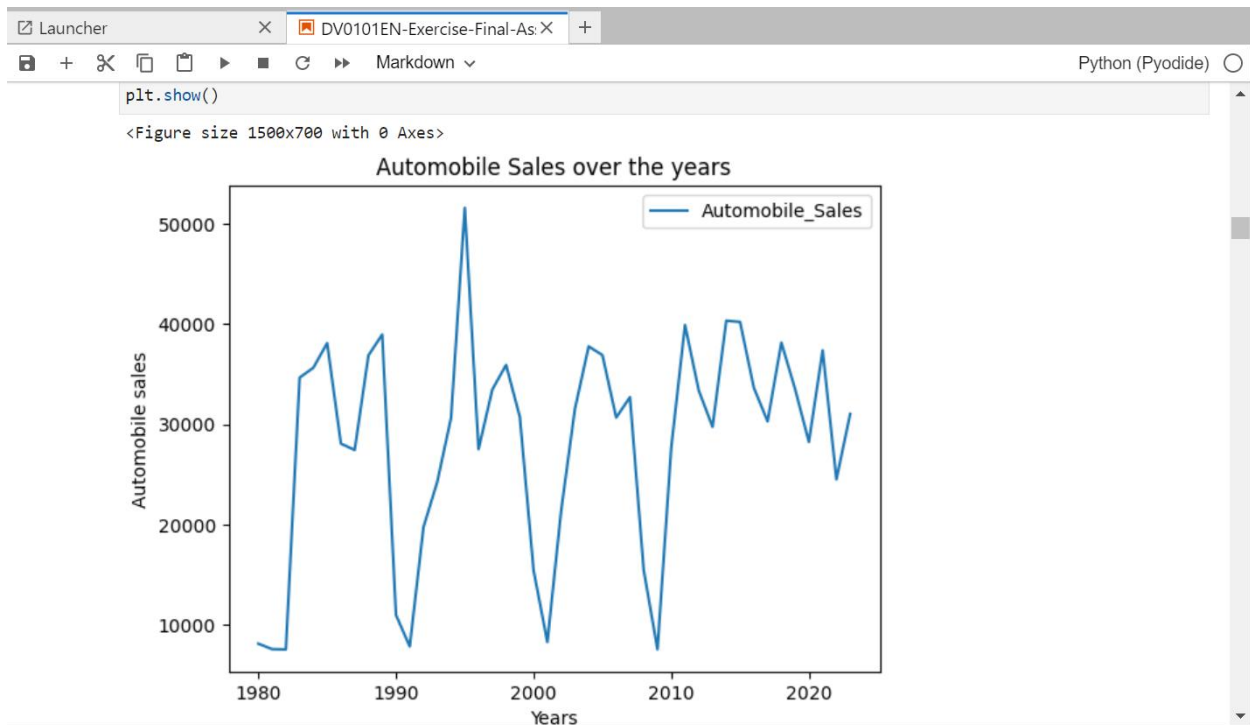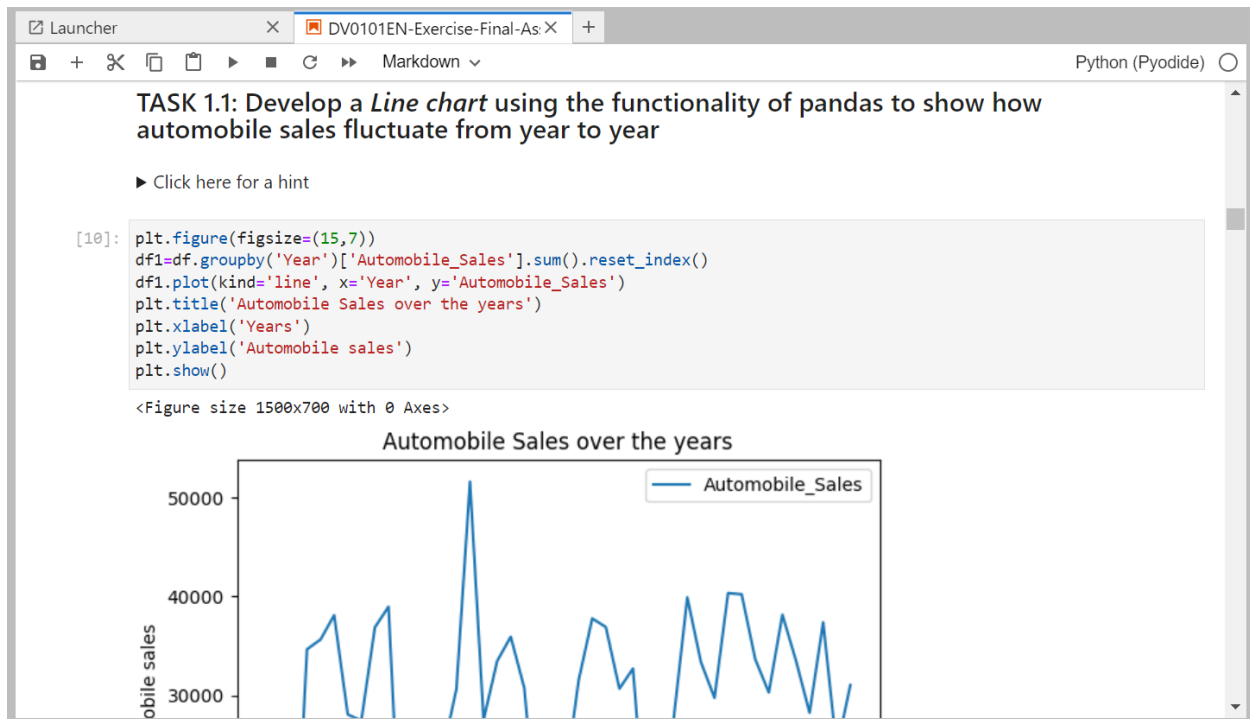
```
[7]:  from js import fetch
      import io

      URL = "https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBMDeveloperSkillsNetwork-DV0101EN-SkillsN
      resp = await fetch(URL)
      text = io.BytesIO((await resp.arrayBuffer()).to_py())
      import pandas as pd
      df = pd.read_csv(text)
      print('Data downloaded and read into a dataframe!')
```
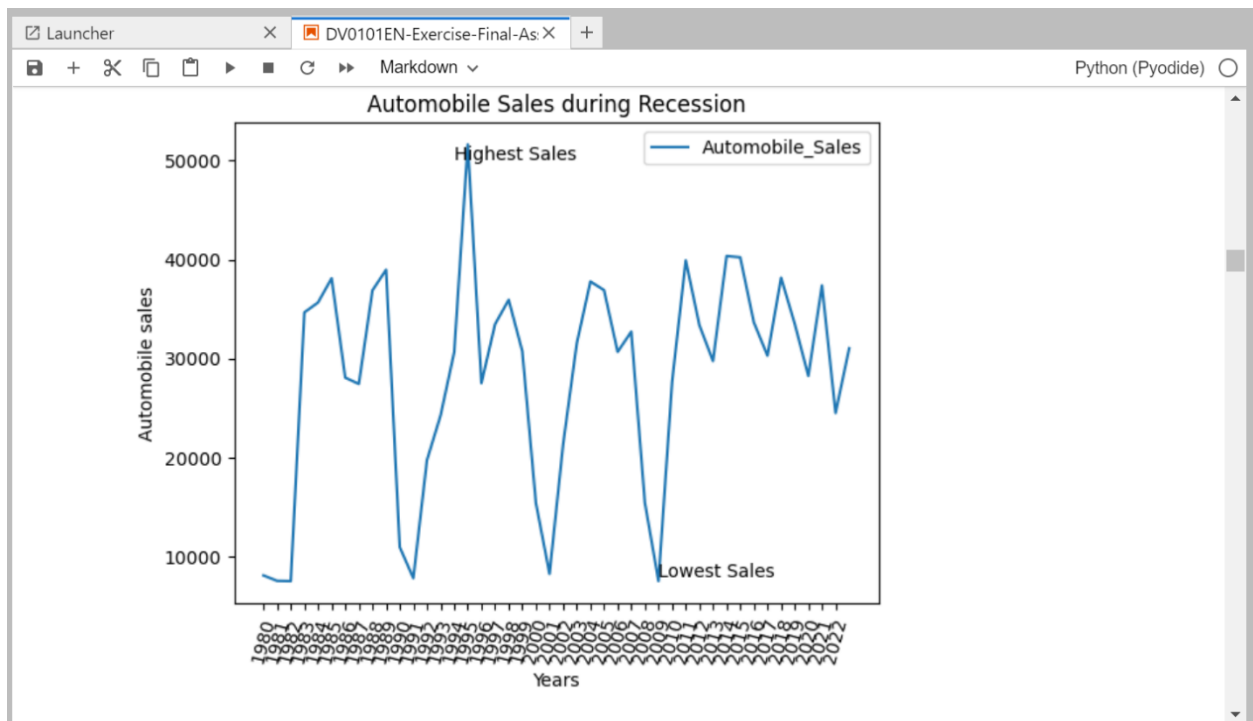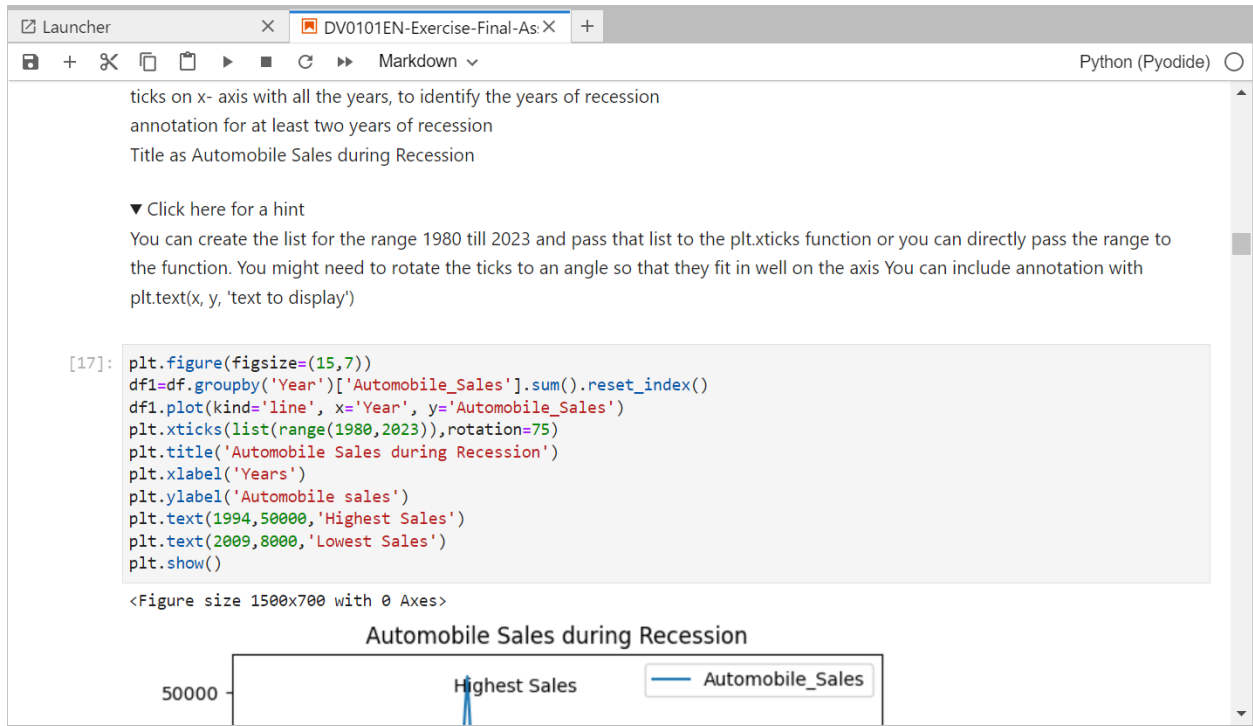
Data downloaded and read into a dataframe!

```
[8]:  df.describe()
```

[8]:	Year	Recession	Consumer_Confidence	Seasonality_Weight	Price	Advertising_Expenditure	Competition	
	528.000000	528.000000	528.000000	528.000000	528.000000	528.000000	528.000000	528.00
	2001.500000	0.214015	101.140170	0.575795	24964.991956	3067.456439	6.064394	40.07
	12.710467	0.410526	10.601154	0.454477	4888.073433	1139.564637	1.968350	16.24
	1980.000000	0.000000	73.900000	0.000000	8793.663000	1009.000000	3.000000	12.50
	1990.750000	0.000000	94.035000	0.250000	21453.300500	2083.500000	4.000000	27.23

TASK 1.1: Develop a *Line chart* using the functionality of pandas to show how automobile sales fluctuate from year to year

▶ Click here for a hint

```
[10]: plt.figure(figsize=(15,7))
      df1=df.groupby('Year')['Automobile_Sales'].sum().reset_index()
      df1.plot(kind='line', x='Year', y='Automobile_Sales')
      plt.title('Automobile Sales over the years')
      plt.xlabel('Years')
      plt.ylabel('Automobile sales')
      plt.show()
```

<Figure size 1500x700 with 0 Axes>

```
      plt.show()
```

<Figure size 1500x700 with 0 Axes>

ticks on x- axis with all the years, to identify the years of recession

annotation for at least two years of recession

Title as Automobile Sales during Recession

▼ Click here for a hint

You can create the list for the range 1980 till 2023 and pass that list to the plt.xticks function or you can directly pass the range to the function. You might need to rotate the ticks to an angle so that they fit in well on the axis You can include annotation with plt.text(x, y, 'text to display')

```python
plt.figure(figsize=(15,7))
df1=df.groupby('Year')['Automobile_Sales'].sum().reset_index()
df1.plot(kind='line', x='Year', y='Automobile_Sales')
plt.xticks(list(range(1980,2023)),rotation=75)
plt.title('Automobile Sales during Recession')
plt.xlabel('Years')
plt.ylabel('Automobile sales')
plt.text(1994,50000,'Highest Sales')
plt.text(2009,8000,'Lowest Sales')
plt.show()
```

```
<Figure size 1500x700 with 0 Axes>
```

Automobile Sales during Recession

50000 —   Highest Sales   ── Automobile_Sales
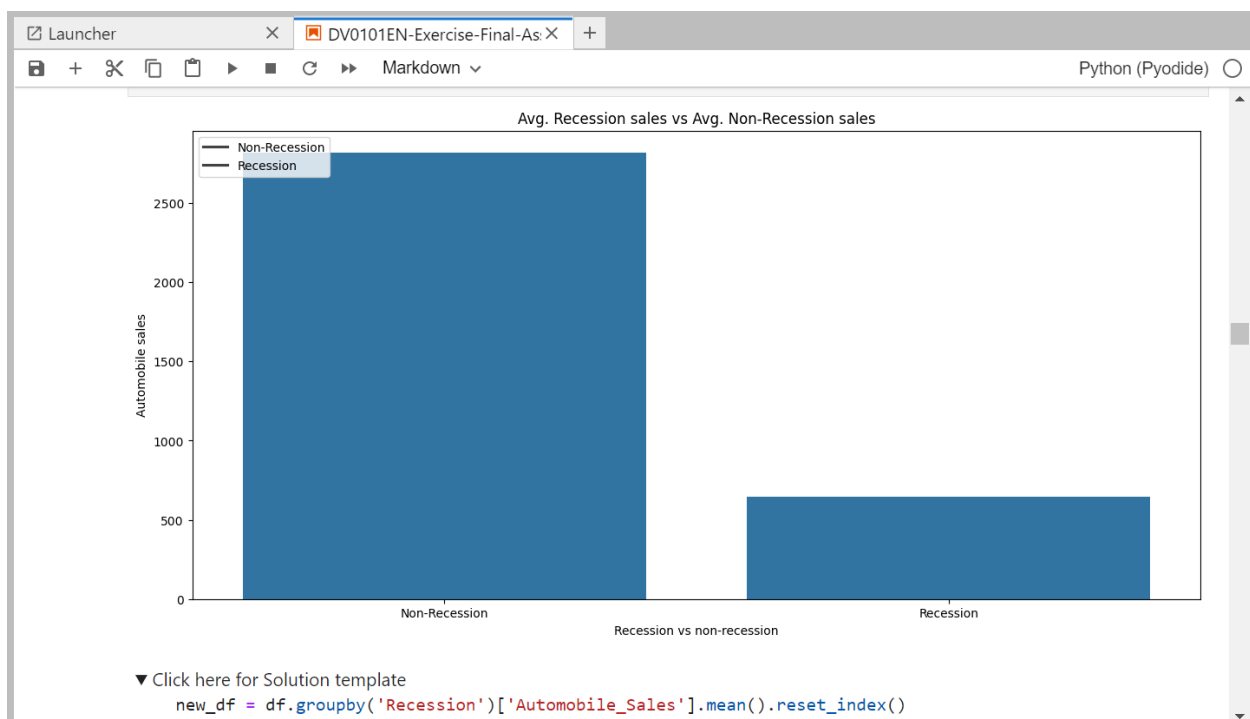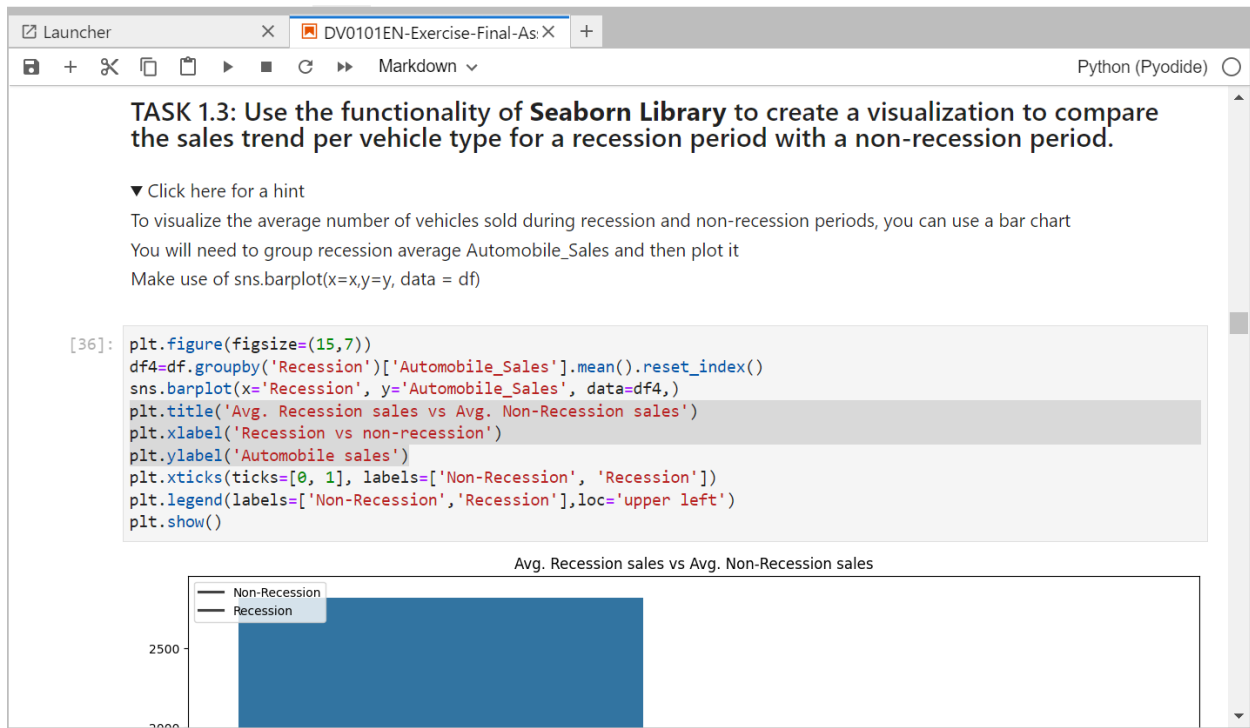
TASK 1.2: Plot different lines for categories of vehicle type and analyse the trend to answer the question Is there a noticeable difference in sales trends between different vehicle types during recession periods?
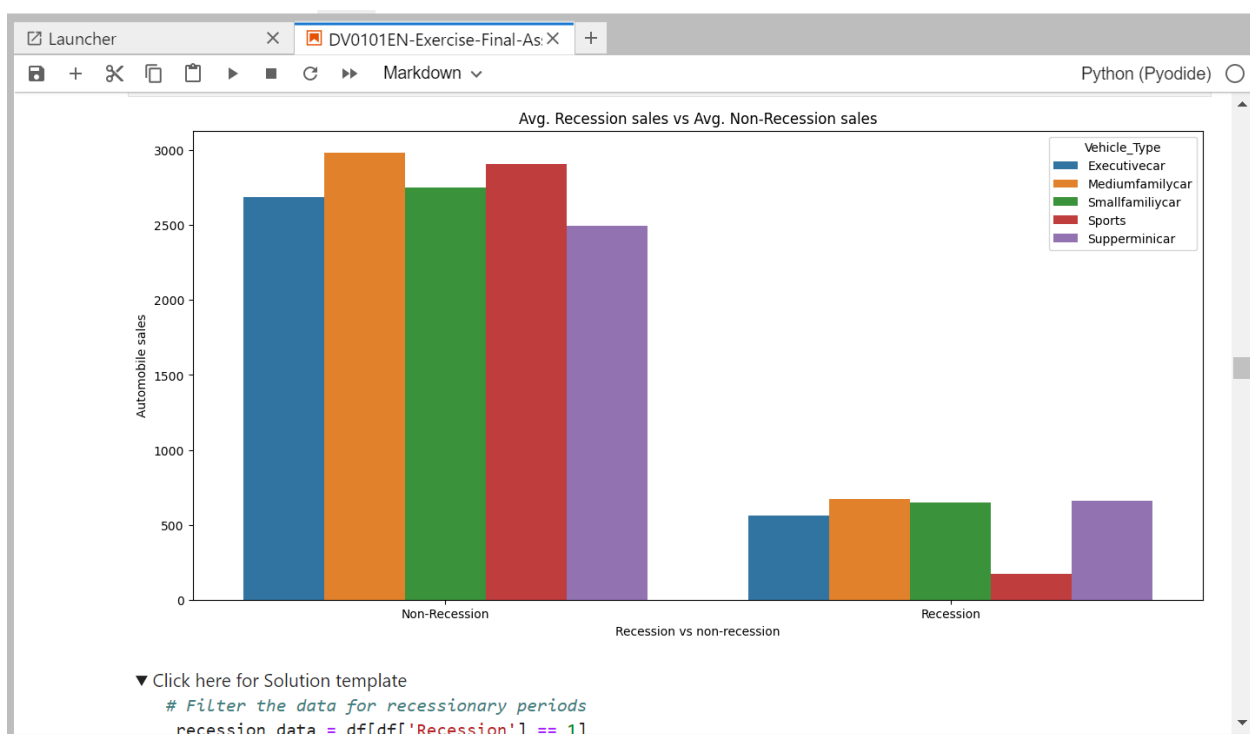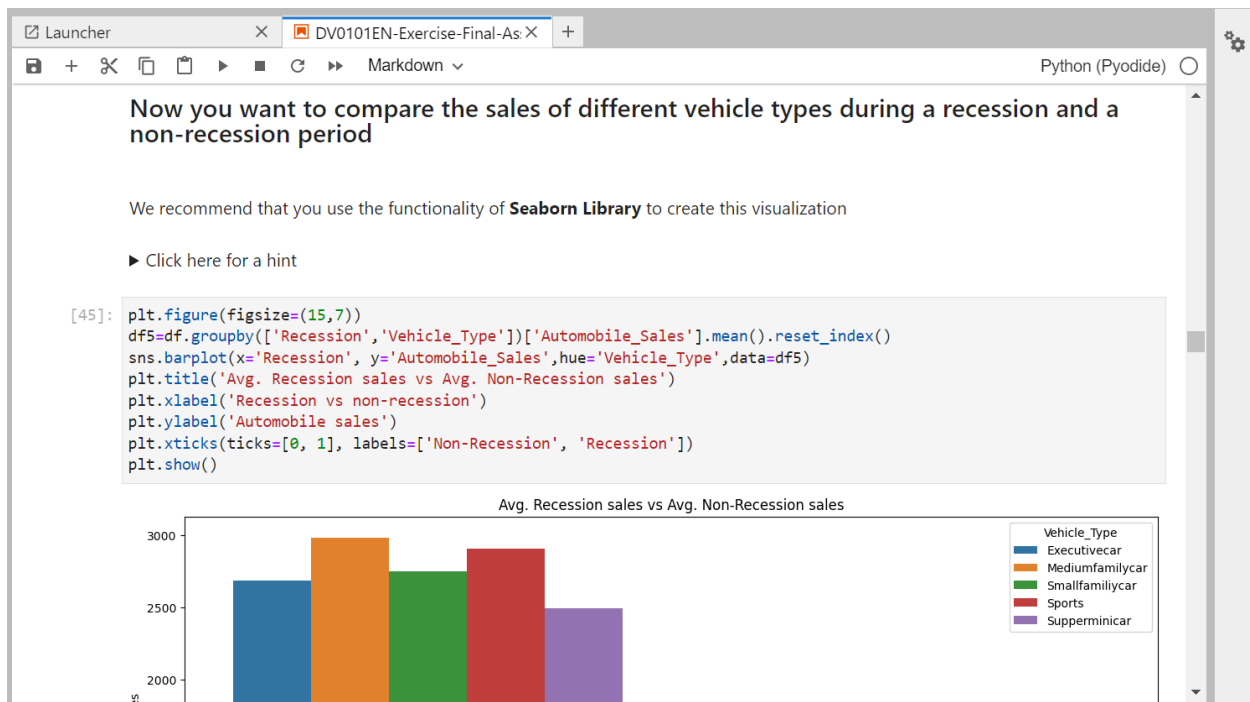
▼ Click here for a hint

```
<p>
You will require:-
<br>create a separate dataframe where the column recession has a value of '1'
<br>to group the year, vehicle_type and calculate the average on the 'Automobile Sales'
<br>one way is to -
<br>use as_index as false else you will endup with multiple-indexed datafame
<br>later set year as index and groupby vehicle over Sales and plot
<br>make use of .plot() with kind = 'line'
<br>do not forget to include labels and title
```

```python
[29]: plt.figure(figsize=(15,7))
      df2=df[df['Recession']== 1]
      df3=df2.groupby(['Year','Vehicle_Type'])['Automobile_Sales'].sum().reset_index()
      df3.set_index('Year',inplace=True)
      df3=df3.groupby('Vehicle_Type')['Automobile_Sales']
      df3.plot(kind='line')
      plt.title('Vehicle type: sales during recessions')
      plt.xlabel('Years')
      plt.ylabel('Automobile sales')
      plt.legend(labels=df.Vehicle_Type.unique(),loc='upper left')
      plt.show
```

```
[29]: <function matplotlib.pyplot.show(close=None, block=None)>
```

```
[29]: <function matplotlib.pyplot.show(close=None, block=None)>
```



```
<details>
    <summary>Click here for Solution template</summary>
```

TASK 1.3: Use the functionality of **Seaborn Library** to create a visualization to compare the sales trend per vehicle type for a recession period with a non-recession period.

▼ Click here for a hint

To visualize the average number of vehicles sold during recession and non-recession periods, you can use a bar chart

You will need to group recession average Automobile_Sales and then plot it

Make use of sns.barplot(x=x,y=y, data = df)

```python
[36]: plt.figure(figsize=(15,7))
df4=df.groupby('Recession')['Automobile_Sales'].mean().reset_index()
sns.barplot(x='Recession', y='Automobile_Sales', data=df4,)
plt.title('Avg. Recession sales vs Avg. Non-Recession sales')
plt.xlabel('Recession vs non-recession')
plt.ylabel('Automobile sales')
plt.xticks(ticks=[0, 1], labels=['Non-Recession', 'Recession'])
plt.legend(labels=['Non-Recession','Recession'],loc='upper left')
plt.show()
```


Avg. Recession sales vs Avg. Non-Recession sales

Avg. Recession sales vs Avg. Non-Recession sales

▼ Click here for Solution template

```python
new_df = df.groupby('Recession')['Automobile_Sales'].mean().reset_index()
```

Now you want to compare the sales of different vehicle types during a recession and a non-recession period

We recommend that you use the functionality of **Seaborn Library** to create this visualization

▶ Click here for a hint

```python
plt.figure(figsize=(15,7))
df5=df.groupby(['Recession','Vehicle_Type'])['Automobile_Sales'].mean().reset_index()
sns.barplot(x='Recession', y='Automobile_Sales',hue='Vehicle_Type',data=df5)
plt.title('Avg. Recession sales vs Avg. Non-Recession sales')
plt.xlabel('Recession vs non-recession')
plt.ylabel('Automobile sales')
plt.xticks(ticks=[0, 1], labels=['Non-Recession', 'Recession'])
plt.show()
```


▼ Click here for Solution template

```python
# Filter the data for recessionary periods
recession_data = df[df['Recession'] == 1]
```

TASK 1.4: Use sub plotting to compare the variations in GDP during recession and non-recession period by developing line plots for each period.

Now, you want to find more insights from the data to understand the reason.
Plot a two line charts using subplotting to answer:-

How did the GDP vary over time during recession and non-recession periods?

Make use of add_subplot() from Matplotlib for this comparision.

```python
[62]:  df6=df[df['Recession']==1]
       df7=df[df['Recession']==0]
       df6=df6.groupby('Year')['GDP'].mean().reset_index()
       df7=df7.groupby('Year')['GDP'].mean().reset_index()
       df6.set_index('Year',inplace=True)
       df7.set_index('Year',inplace=True)

       fig=plt.figure(figsize=(15,7))
       ax1=fig.add_subplot(1,2,1)
       ax1.plot(df6)
       ax1.set_title('GDP during recession')
       ax1.set_xlabel('Year')
       ax1.set_ylabel('GDP')
```

```python
       ax2=fig.add_subplot(1,2,2)
       ax2.plot(df7)
       ax2.set_title('GDP during non-recession')
       ax2.set_ylabel('GDP')
       ax2.set_xlabel('Year')

       plt.show()
```
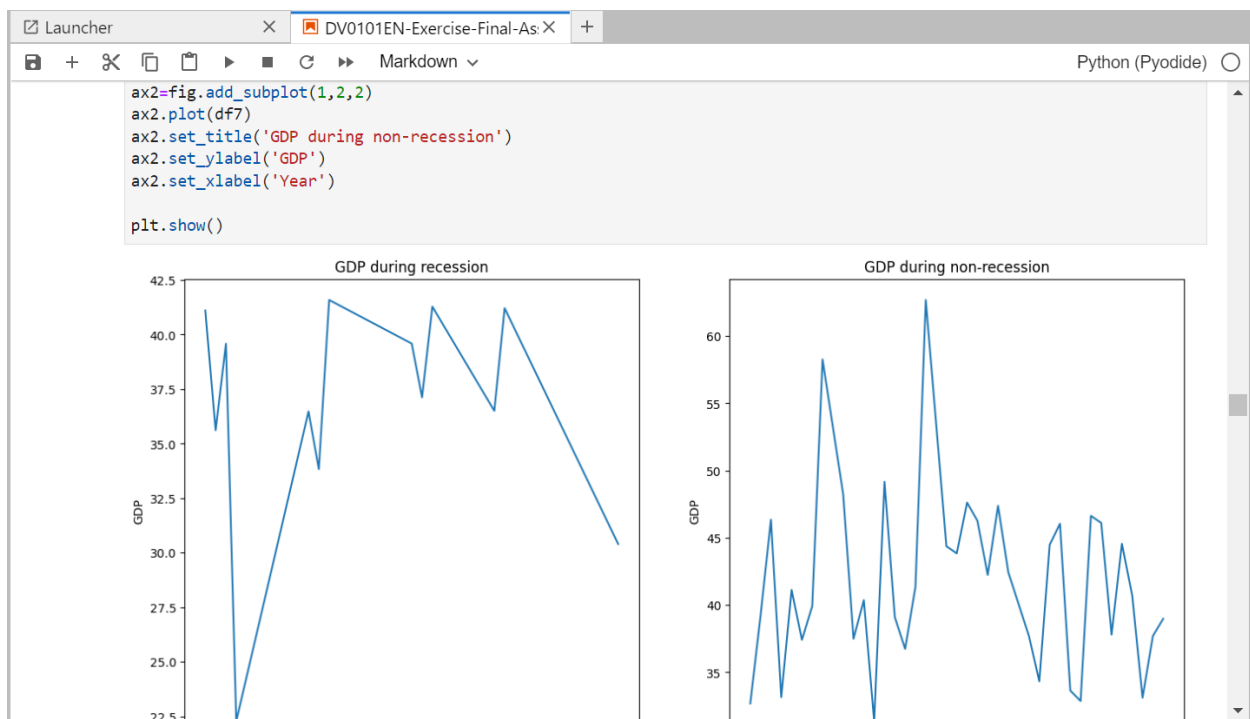
TASK 1.5: Develop a Bubble plot for displaying the impact of seasonality on Automobile Sales.

How has seasonality impacted the sales, in which months the sales were high or low? Check it for non-recession years to understand the trend

Develop a Bubble plot for displaying Automobile Sales for every month and use Seasonality Weight for representing the size of each bubble

Title this plot as 'Seasonality impact on Automobile Sales'

▼ Click here for a hint

You can create Bubble Chart by calling the scatter()

Pass the 'Month' and 'Automobile_Sales' to the functions as x and y and then use Seasonality weight for size parameter

```python
[67]: df8=df[df['Recession']==0]
      sns.scatterplot(x='Month', y='Automobile_Sales',data=df8, size=df8.Seasonality_Weight)
      plt.xlabel('Month')
      plt.ylabel('Automobile_Sales')
      plt.title('Seasonality impact on Automobile Sales')
      plt.show()
```

Seasonality impact on Automobile Sales

Seasonality impact on Automobile Sales
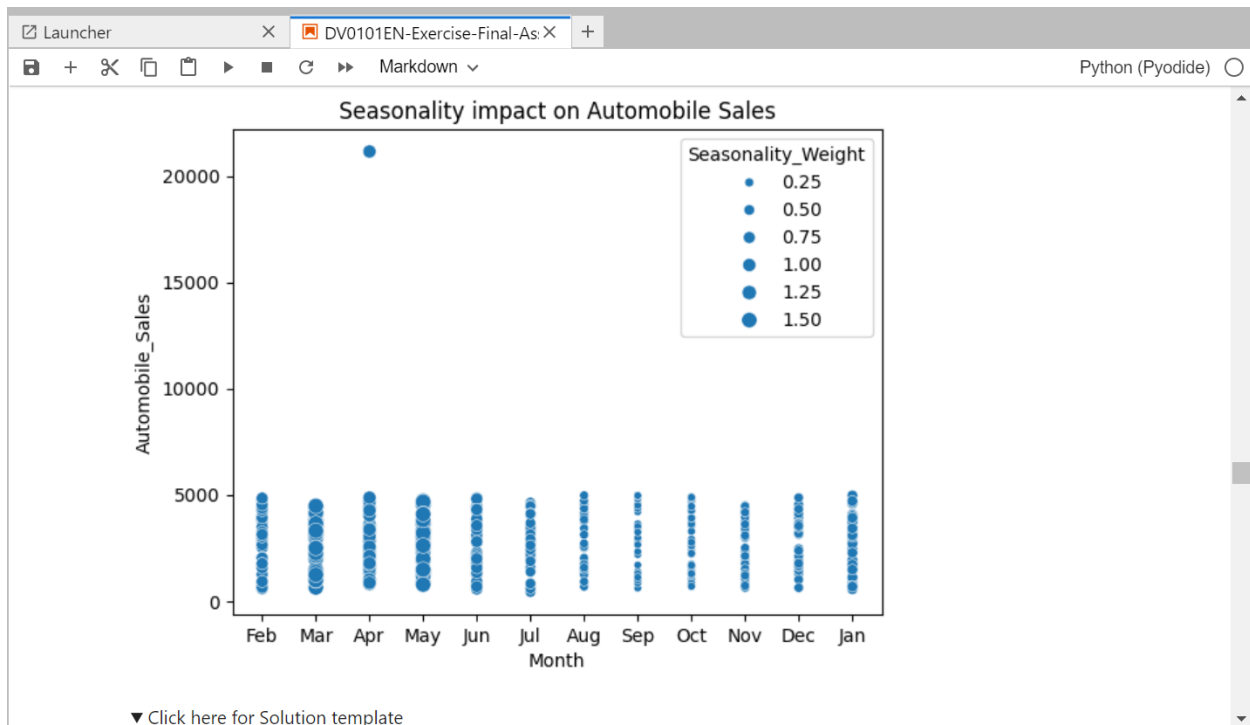
▼ Click here for Solution template

TASK 1.6: Use the functionality of Matplotlib to develop a scatter plot to identify the correlation between average vehicle price relate to the sales volume during recessions.

From the data, develop a scatter plot to identify if there a correlation between consumer confidence and automobile sales during recession period?

Title this plot as 'Consumer Confidence and Automobile Sales during Recessions'
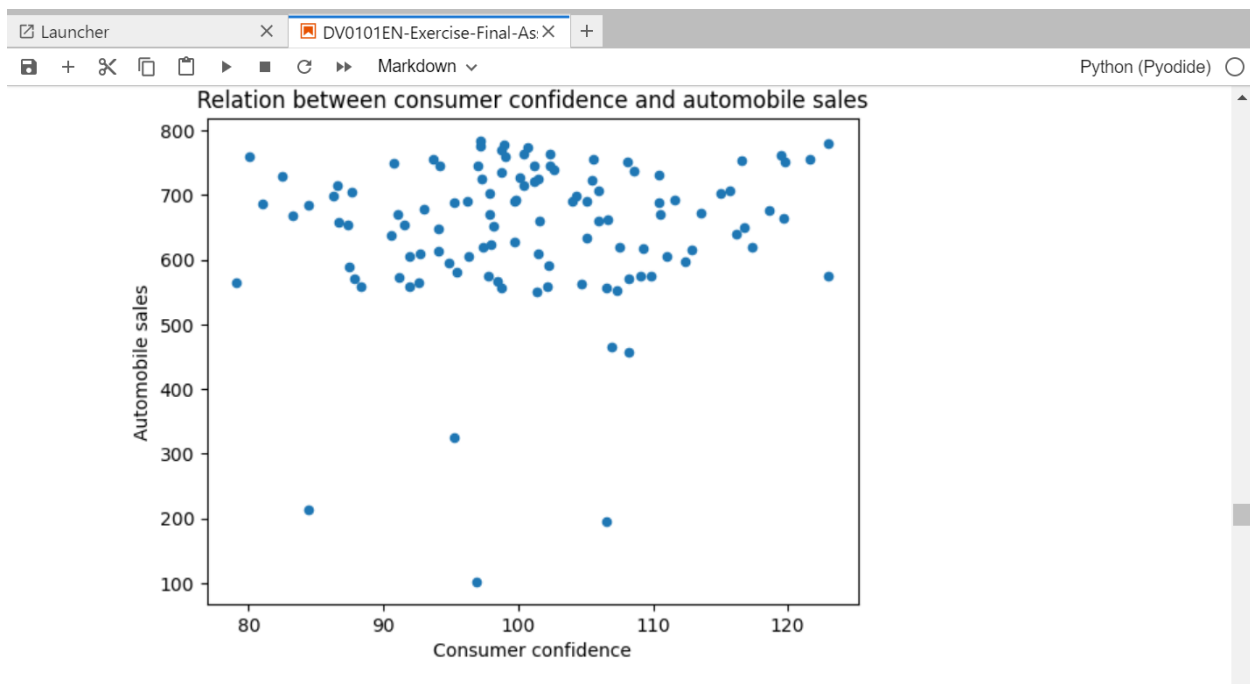
▼ Click here for a hint
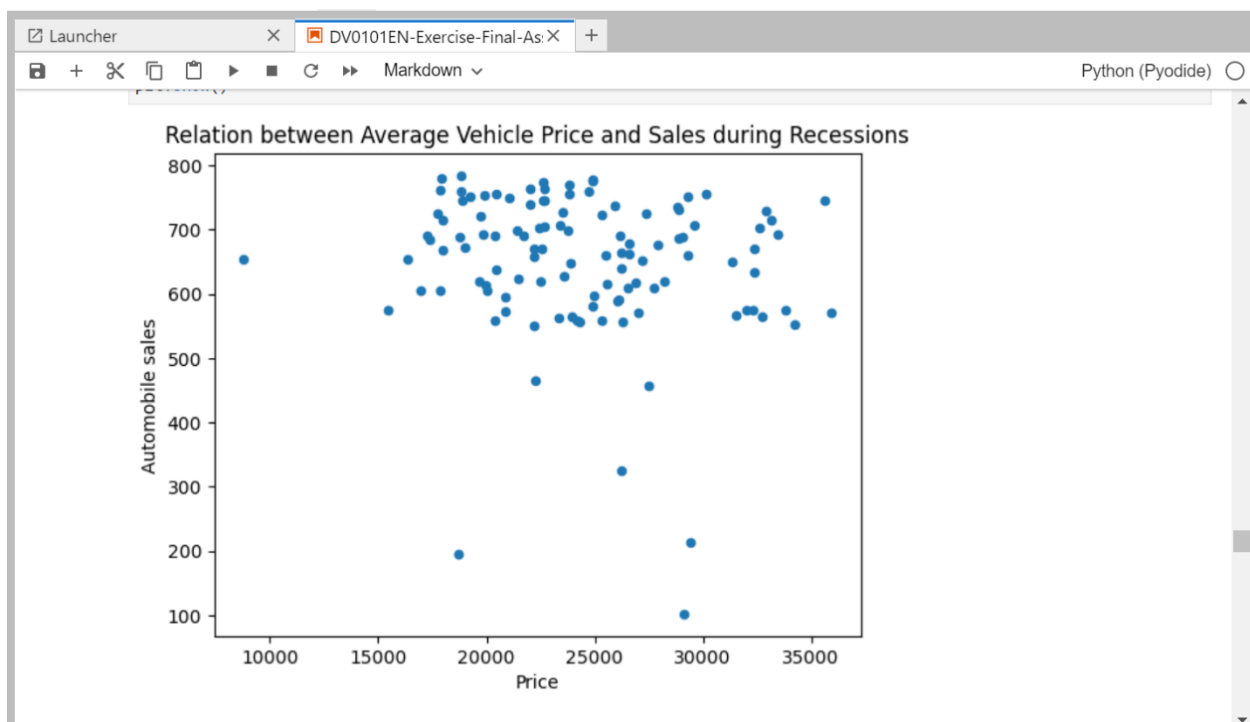
You can create dataframe where recession is '1'.

Pass the 'Consumer_Confidence' and 'Automobile_Sales' to the plt.scatter()

```python
[70]: df9=df[df['Recession']==1]
      df9.plot(kind='scatter',x='Consumer_Confidence',y='Automobile_Sales')
      plt.xlabel('Consumer confidence')
      plt.ylabel('Automobile sales')
      plt.title('Relation between consumer confidence and automobile sales')
      plt.show()
```

Relation between consumer confidence and automobile sales

800 ⌐

Relation between consumer confidence and automobile sales

How does the average vehicle price relate to the sales volume during recessions?

Plot another scatter plot and title it as 'Relationship between Average Vehicle Price and Sales during Recessions'

```
[73]: df10=df[df['Recession']==1]
      df10.plot(kind='scatter',x='Price',y='Automobile_Sales')
      plt.xlabel('Price')
      plt.ylabel('Automobile sales')
      plt.title('Relation between Average Vehicle Price and Sales during Recessions')
      plt.show()
```


Relation between Average Vehicle Price and Sales during Recessions

Relation between Average Vehicle Price and Sales during Recessions

TASK 1.7: Create a pie chart to display the portion of advertising expenditure of XYZAutomotives during recession and non-recession periods.

How did the advertising expenditure of XYZAutomotives change during recession and non-recession periods?

▼ Click here for a hint

You can create two dataframe for recession and nonreccession period.

Calculate the sum of Advertising_Expenditure for both dataframes

Pass these total values to plt.pie(). May include labels as ['Recession', 'Non-Recession']

Feel Free to customie the pie further

title this plot as - Advertising Expenditure during Recession and Non-Recession Periods

```python
[78]: df10=df[df['Recession']==1]
rec_adv=df10.Advertising_Expenditure.sum()
df11=df[df['Recession']==0]
nonrec_adv=df11.Advertising_Expenditure.sum()
tot_adv=[rec_adv,nonrec_adv]
plt.pie(tot_adv, labels=['Exp_during recession','Exp_during non recession'],autopct='%1.1f%%')
plt.title('Advertising Expenditure during Recession and Non-Recession Periods')
plt.show()
```

Advertising Expenditure during Recession and Non-Recession Periods

---

Advertising Expenditure during Recession and Non-Recession Periods

▼ Click here for Solution template
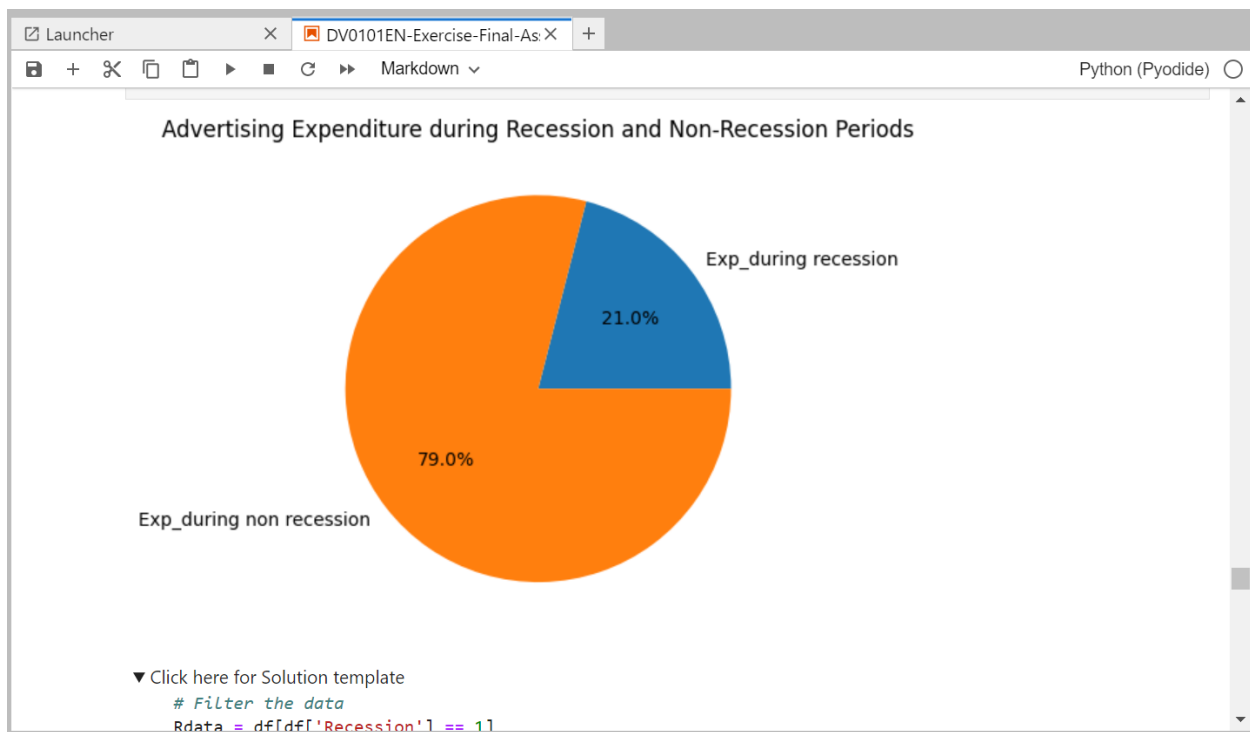
```python
# Filter the data
Rdata = df[df['Recession'] == 1]
```

TASK 1.8: Develop a pie chart to display the total Advertisement expenditure for each vehicle type during recession period.

Can we observe the share of each vehicle type in total sales during recessions?

Create another pie plot to display the total advertisement expenditure for each vehicle type

▼ Click here for a hint

You will be required to group vehicle type for sum of advertisement expenditure.

the plot a pie with the data, May include relevant labels

title this plot as - Share of Each Vehicle Type in Total Sales during Recessions

```
[85]:  df12=df[df['Recession']==1]
       df12=df12.groupby('Vehicle_Type')['Advertising_Expenditure'].sum().reset_index()
       plt.pie(df12.Advertising_Expenditure, labels=df12.Vehicle_Type,autopct='%1.1f%%')
       plt.title('Share of Each Vehicle Type in Total Sales during Recessions')
       plt.show()
```

Share of Each Vehicle Type in Total Sales during Recessions
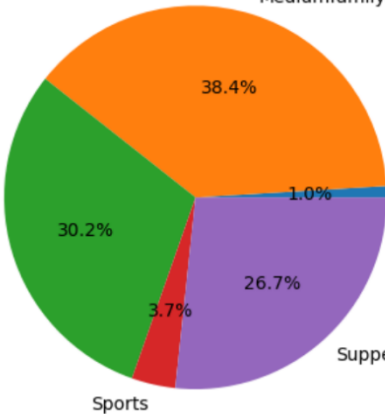
Mediumfamilycar

Share of Each Vehicle Type in Total Sales during Recessions

▶ Click here for Solution template

Inference

TASK 1.9: Develop a countplot to analyse the effect of the unemployment rate on vehicle type and sales during the Recession Period.

Analyze the effect of the unemployment rate on vehicle type and sales during the Recession Period

You can create a countplot and title the plot as 'Effect of Unemployment Rate on Vehicle Type and Sales'

▼ Click here for a hint

Filter out the data for recession period

Make use of countplot() from seaborn and pass the relavent data

```python
[91]: plt.figure(figsize=(20,7))
df13=df[df['Recession']==1]
sns.countplot(x='unemployment_rate',hue='Vehicle_Type',data=df13)
plt.xlabel('Unemployment Rate')
plt.ylabel('Count')
plt.title('Effect of Unemployment Rate on Vehicle Type and Sales')
plt.legend(loc='upper right')
plt.xticks(rotation=90)
plt.show()
```