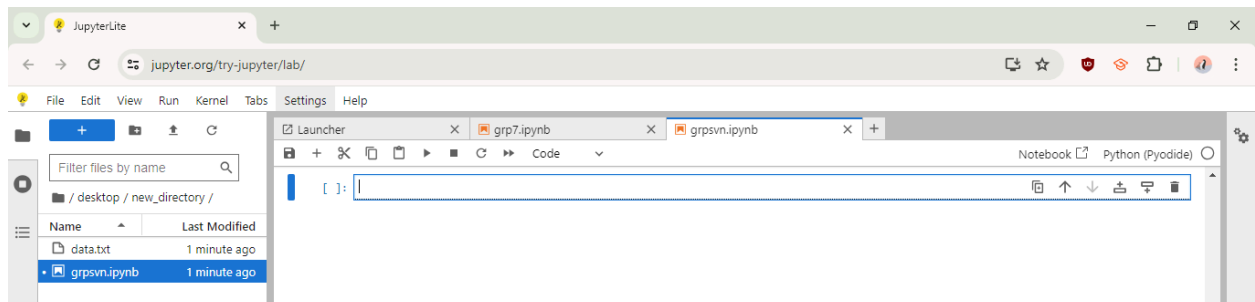
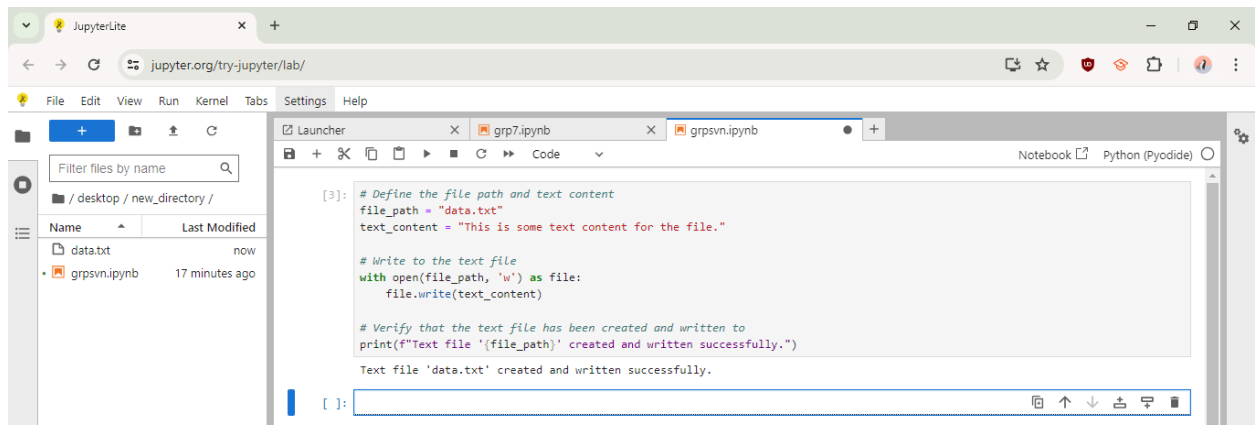


Jupyter notebook Activity

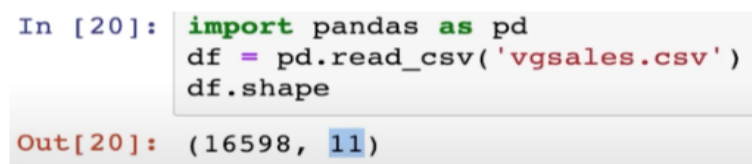
- Adding Folder



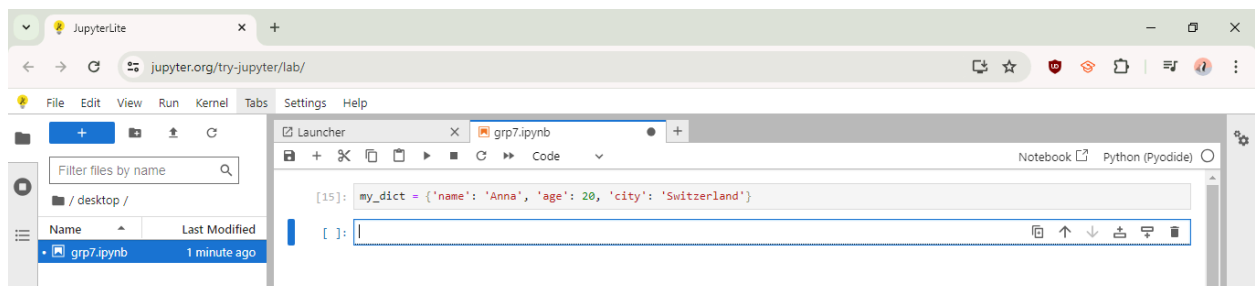
- Adding Text file



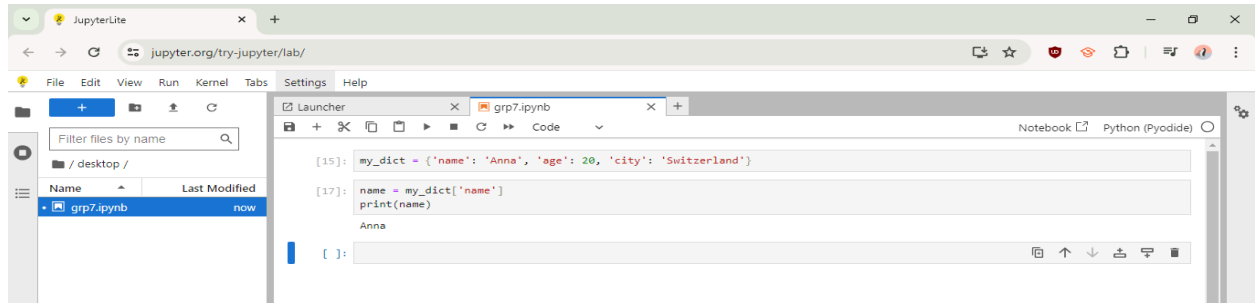
- CSV file for data analysis and visualization



- To write and call dictionary methods

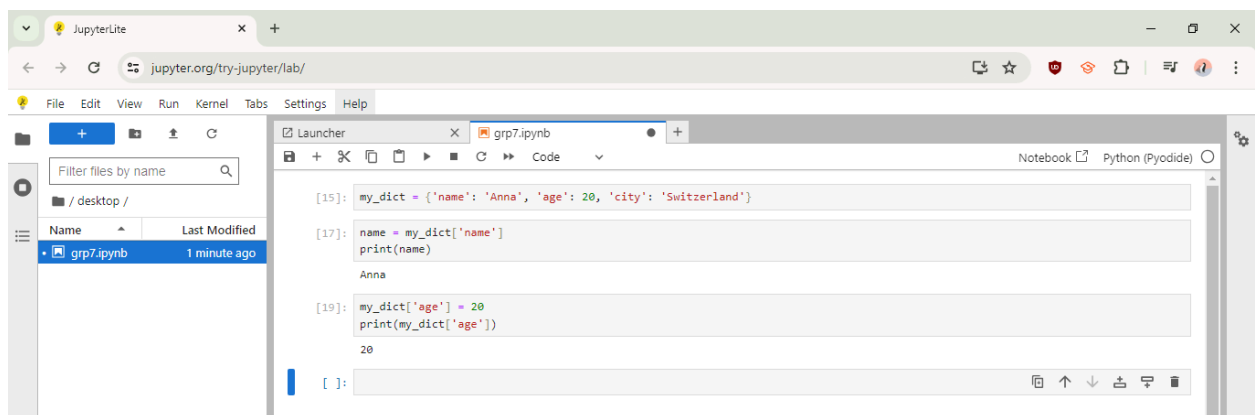


- **Accessing Items in the Dictionary:**



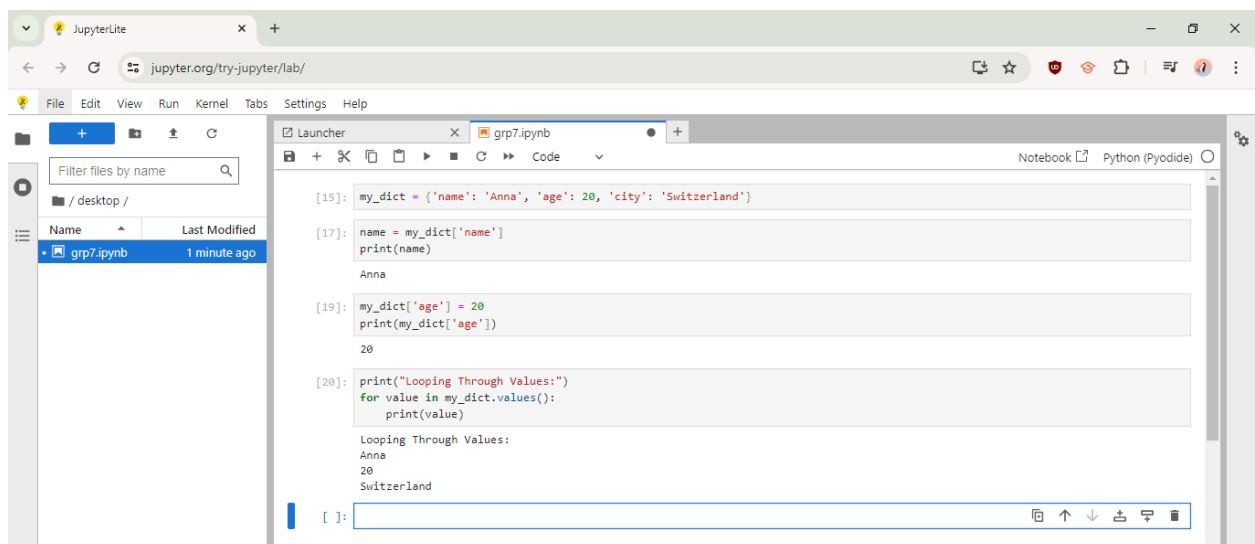
The screenshot shows the JupyterLab interface with a file browser on the left and a code editor on the right. The file browser shows a file named 'grp7.ipynb' on the desktop. The code editor has two cells. The first cell (index 15) contains the code to create a dictionary: `my_dict = {'name': 'Anna', 'age': 20, 'city': 'Switzerland'}`. The second cell (index 17) contains the code to access the 'name' value: `name = my_dict['name']; print(name)`. The output of the second cell is 'Anna'.

- **Change Values in the Dictionary:**



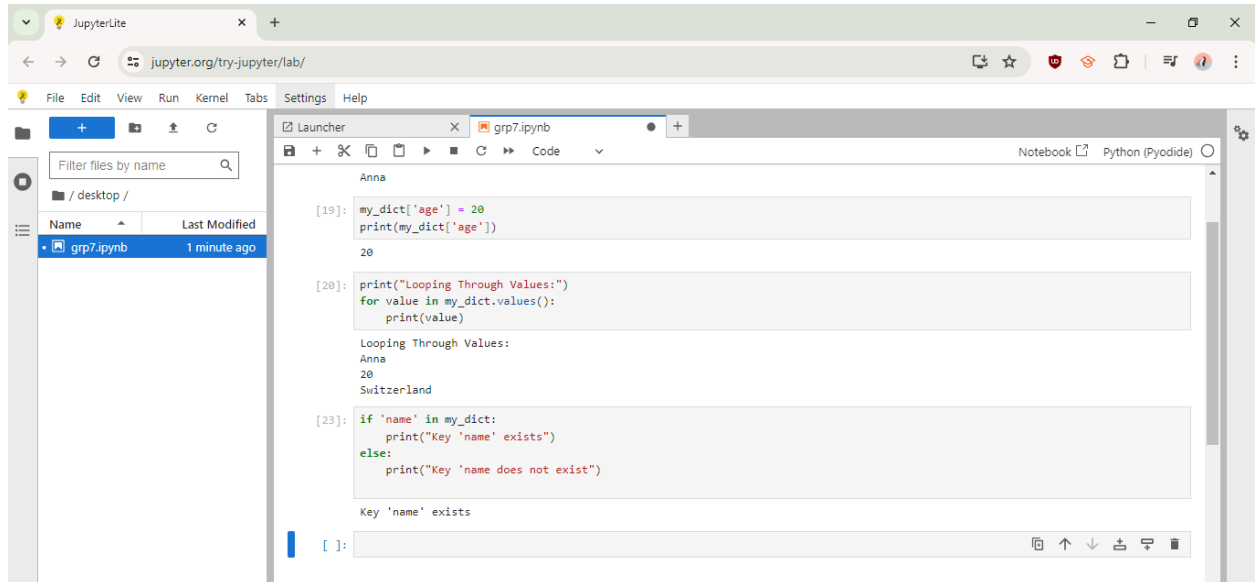
The screenshot shows the JupyterLab interface with the same file browser and code editor. The code editor now has three cells. The first cell (index 15) is the same as before. The second cell (index 17) is the same as before. The third cell (index 19) contains the code to change the 'age' value: `my_dict['age'] = 20; print(my_dict['age'])`. The output of the third cell is '20'.

- **Loop Through Dictionary Values:**



The screenshot shows the JupyterLab interface with the same file browser and code editor. The code editor now has four cells. The first cell (index 15) is the same as before. The second cell (index 17) is the same as before. The third cell (index 19) is the same as before. The fourth cell (index 20) contains the code to loop through the dictionary values: `print("Looping Through Values:"); for value in my_dict.values(): print(value)`. The output of the fourth cell is 'Looping Through Values:', 'Anna', '20', and 'Switzerland'.

- **Check if Key Exists in the Dictionary:**



The screenshot shows a JupyterLab interface with a notebook named 'grp7.ipynb'. The code in the notebook is as follows:

```
[19]: my_dict['age'] = 20
      print(my_dict['age'])

20

[20]: print("Looping Through Values:")
      for value in my_dict.values():
          print(value)

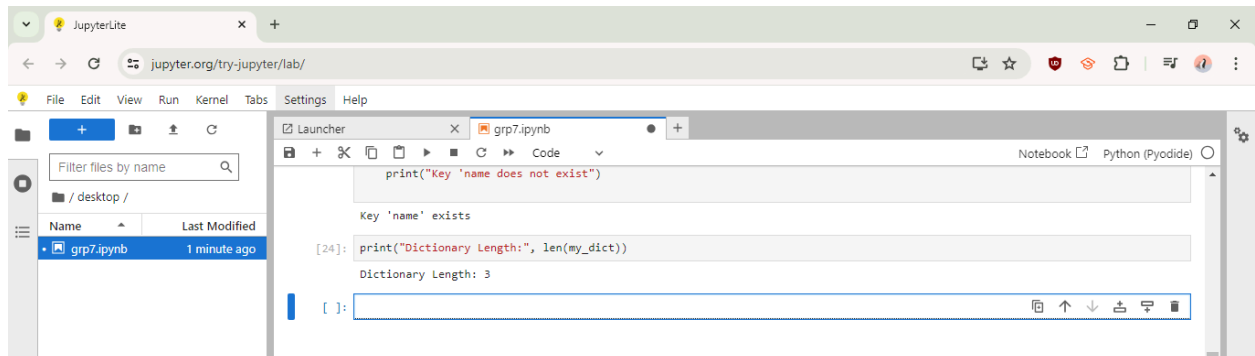
Looping Through Values:
Anna
20
Switzerland

[23]: if 'name' in my_dict:
      print("Key 'name' exists")
      else:
          print("Key 'name' does not exist")

Key 'name' exists

[ ]:
```

- **Checking for Dictionary Length:**



The screenshot shows a JupyterLab interface with a notebook named 'grp7.ipynb'. The code in the notebook is as follows:

```
print("Key 'name' does not exist")

Key 'name' exists

[24]: print("Dictionary Length:", len(my_dict))

Dictionary Length: 3

[ ]:
```

- **Adding Items in the Dictionary:**



The screenshot shows a JupyterLab interface with a notebook named 'grp7.ipynb'. The code in the notebook is as follows:

```
Key 'name' exists

[24]: print("Dictionary Length:", len(my_dict))

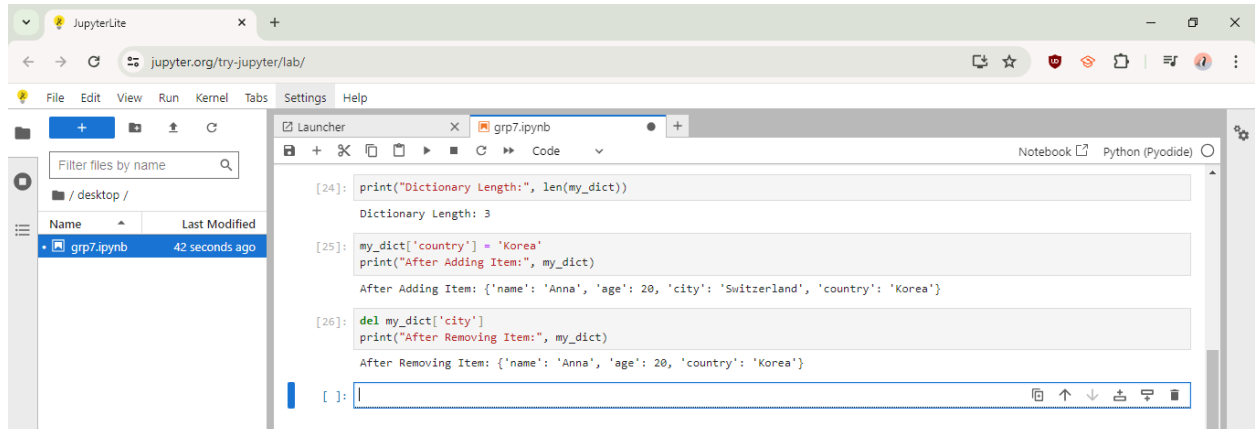
Dictionary Length: 3

[25]: my_dict['country'] = 'Korea'
      print("After Adding Item:", my_dict)

After Adding Item: {'name': 'Anna', 'age': 20, 'city': 'Switzerland', 'country': 'Korea'}

[ ]:
```

- **Removing Items in the Dictionary:**



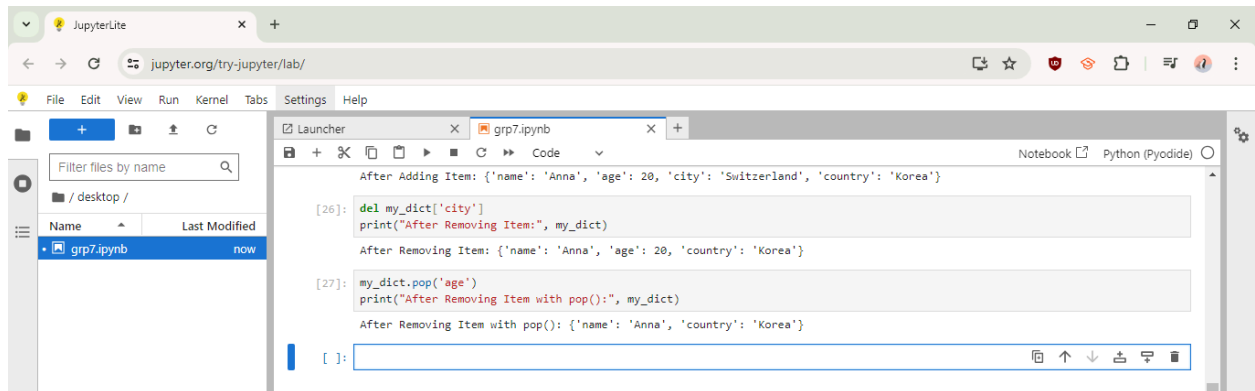
The screenshot shows a JupyterLab interface with a file browser on the left and a code editor on the right. The file browser shows a file named 'grp7.ipynb' with a last modified time of '42 seconds ago'. The code editor displays the following code:

```
[24]: print("Dictionary Length:", len(my_dict))
      Dictionary Length: 3

[25]: my_dict['country'] = 'Korea'
      print("After Adding Item:", my_dict)
      After Adding Item: {'name': 'Anna', 'age': 20, 'city': 'Switzerland', 'country': 'Korea'}

[26]: del my_dict['city']
      print("After Removing Item:", my_dict)
      After Removing Item: {'name': 'Anna', 'age': 20, 'country': 'Korea'}
```

- **Remove an Item Using del Statement:**



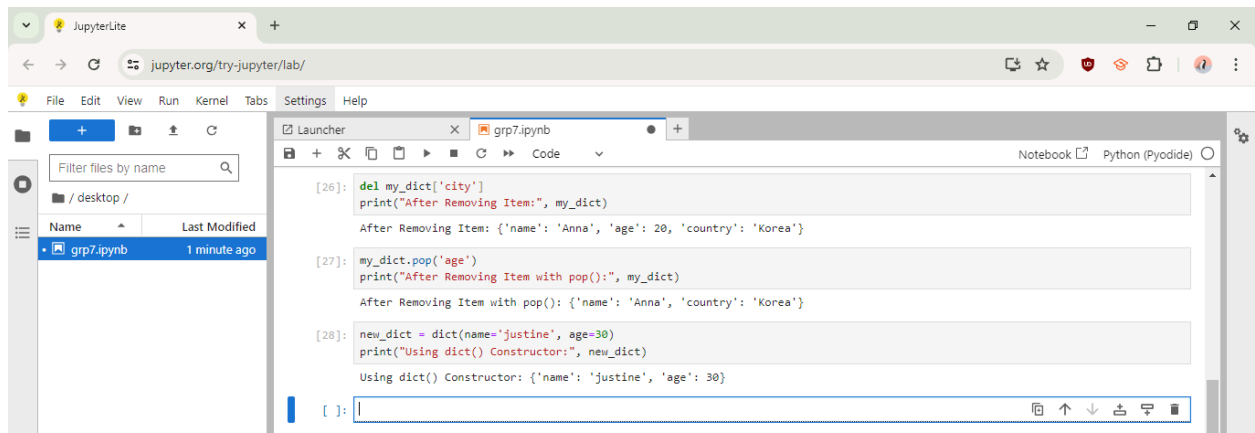
The screenshot shows a JupyterLab interface with a file browser on the left and a code editor on the right. The file browser shows a file named 'grp7.ipynb' with a last modified time of 'now'. The code editor displays the following code:

```
After Adding Item: {'name': 'Anna', 'age': 20, 'city': 'Switzerland', 'country': 'Korea'}

[26]: del my_dict['city']
      print("After Removing Item:", my_dict)
      After Removing Item: {'name': 'Anna', 'age': 20, 'country': 'Korea'}

[27]: my_dict.pop('age')
      print("After Removing Item with pop():", my_dict)
      After Removing Item with pop(): {'name': 'Anna', 'country': 'Korea'}
```

- **The dict() Constructor:**



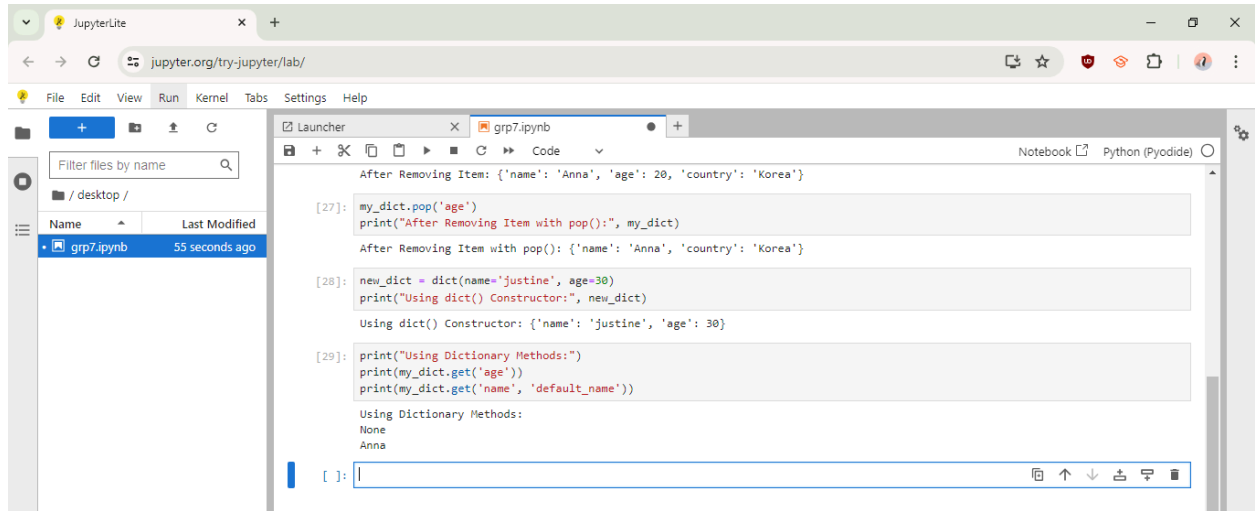
The screenshot shows a JupyterLab interface with a file browser on the left and a code editor on the right. The file browser shows a file named 'grp7.ipynb' with a last modified time of '1 minute ago'. The code editor displays the following code:

```
[26]: del my_dict['city']
      print("After Removing Item:", my_dict)
      After Removing Item: {'name': 'Anna', 'age': 20, 'country': 'Korea'}

[27]: my_dict.pop('age')
      print("After Removing Item with pop():", my_dict)
      After Removing Item with pop(): {'name': 'Anna', 'country': 'Korea'}

[28]: new_dict = dict(name='Justine', age=30)
      print("Using dict() Constructor:", new_dict)
      Using dict() Constructor: {'name': 'Justine', 'age': 30}
```

- **Dictionary Methods:**



The screenshot shows a JupyterLab interface with a file browser on the left and a code editor on the right. The file browser shows a file named 'grp7.ipynb' on the desktop. The code editor contains the following Python code:

```
After Removing Item: {'name': 'Anna', 'age': 20, 'country': 'Korea'}

[27]: my_dict.pop('age')
      print("After Removing Item with pop():", my_dict)

After Removing Item with pop(): {'name': 'Anna', 'country': 'Korea'}

[28]: new_dict = dict(name='justine', age=30)
      print("Using dict() Constructor:", new_dict)

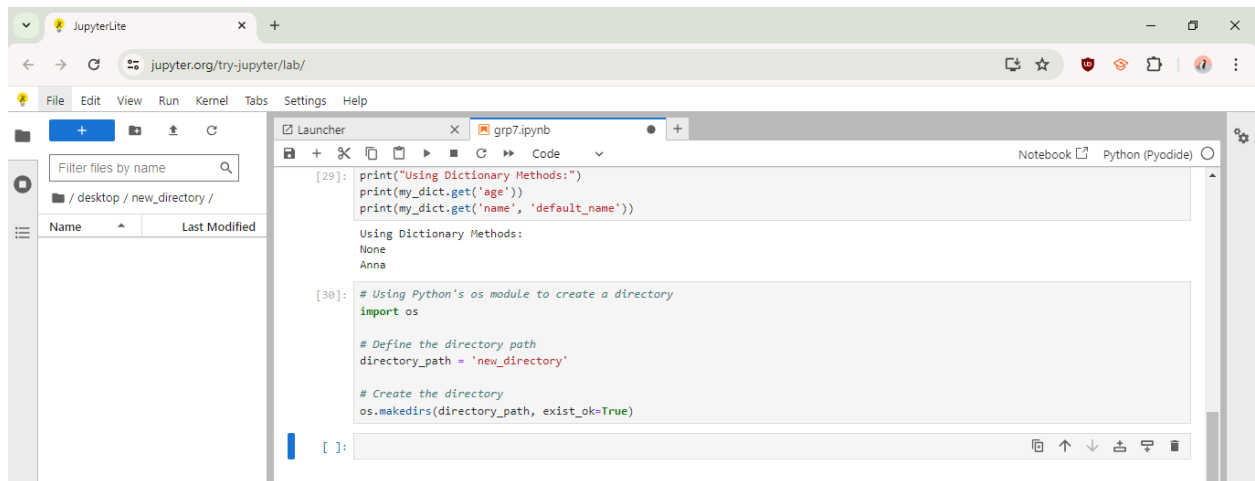
Using dict() Constructor: {'name': 'justine', 'age': 30}

[29]: print("Using Dictionary Methods:")
      print(my_dict.get('age'))
      print(my_dict.get('name', 'default_name'))

Using Dictionary Methods:
None
Anna

[ ]: |
```

- **To create a directory using jupyter notebook:**



The screenshot shows a JupyterLab interface with a file browser on the left and a code editor on the right. The file browser shows a file named 'grp7.ipynb' on the desktop. The code editor contains the following Python code:

```
[29]: print("Using Dictionary Methods:")
      print(my_dict.get('age'))
      print(my_dict.get('name', 'default_name'))

Using Dictionary Methods:
None
Anna

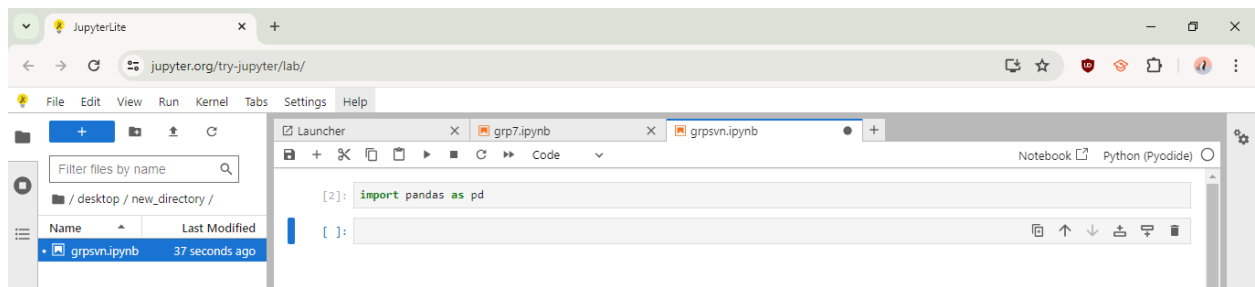
[30]: # Using Python's os module to create a directory
      import os

      # Define the directory path
      directory_path = 'new_directory'

      # Create the directory
      os.makedirs(directory_path, exist_ok=True)

[ ]: |
```

- **To import libraries:**



The screenshot shows a JupyterLab interface with a file browser on the left and a code editor on the right. The file browser shows a file named 'grp7.ipynb' on the desktop. The code editor contains the following Python code:

```
[2]: import pandas as pd

[ ]: |
```

- To use CSV file for data:

```
group.ipynb
Group7.txt

+ ✂ 📄 📄 ▶ ■ ↺ ▶▶ Code

[24]: import pandas as pd

      df = pd.read_csv('group.csv')
```

- Analysis and visualization

```
nb
Group7.txt X +

📄 📄 ▶ ■ ↺ ▶▶ Code ▼

import pandas as pd

df = pd.read_csv('vgsales.csv')

print(df.head())
print(df.info())
print(df.describe())

df.dropna(inplace=True)

mean_global_sales = df['Global_Sales'].mean()
print("Mean Global Sales:", mean_global_sales)

import matplotlib.pyplot as plt

plt.figure(figsize=(10, 6))
plt.hist(df['Global_Sales'], bins=20, color='skyblue', edgecolor='black', alpha=0.7)
plt.title('Histogram of Global Sales')
plt.xlabel('Global Sales')
plt.ylabel('Frequency')
plt.grid(True)
plt.show()
```

	Rank	Name	Platform	Year	Genre	Publisher
0	1	Wii Sports	Wii	2006.0	Sports	Nintendo
1	2	Super Mario Bros.	NES	1985.0	Platform	Nintendo
2	3	Mario Kart Wii	Wii	2008.0	Racing	Nintendo
3	4	Wii Sports Resort	Wii	2009.0	Sports	Nintendo
4	5	Pokemon Red/Pokemon Blue	GB	1996.0	Role-Playing	Nintendo
	NA_Sales	EU_Sales	JP_Sales	Other_Sales	Global_Sales	
0	41.40	29.02	3.77	8.46	82.74	
1	29.08	3.58	0.81	0.77	40.24	
2	15.85	12.88	3.79	3.31	35.82	
3	15.75	11.01	3.28	2.96	33.00	
4	11.27	8.89	10.22	1.00	31.37	

```

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 16598 entries, 0 to 16597
Data columns (total 11 columns):
#   Column      Non-Null Count  Dtype
---  -
0   Rank         16598 non-null  int64
1   Name         16598 non-null  object
2   Platform     16598 non-null  object
3   Year         16327 non-null  float64
4   Genre        16598 non-null  object
5   Publisher    16540 non-null  object
6   NA_Sales     16598 non-null  float64
7   EU_Sales     16598 non-null  float64
8   JP_Sales     16598 non-null  float64
9   Other_Sales  16598 non-null  float64
10  Global_Sales 16598 non-null  float64
dtypes: float64(6), int64(1), object(4)
memory usage: 1.1+ MB

```

```

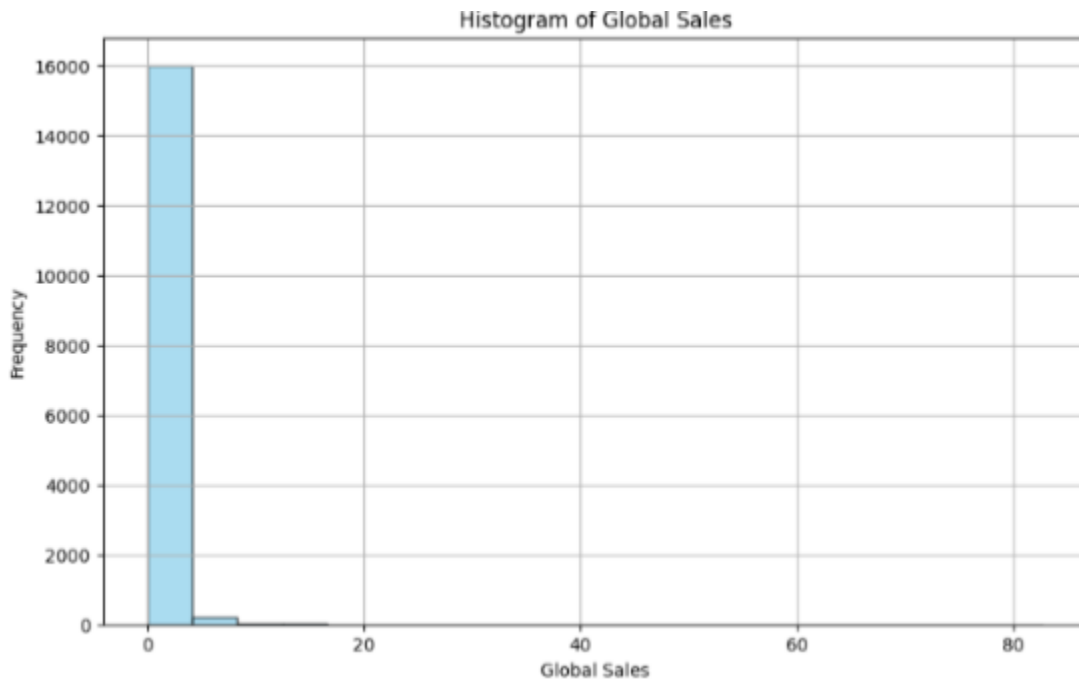
None
Rank      Year      NA_Sales    EU_Sales    JP_Sales \
count 16598.000000 16327.000000 16598.000000 16598.000000 16598.000000
mean   8300.605254   2006.406443    0.264667    0.146652    0.077782
std    4791.853933    5.828981    0.816683    0.505351    0.309291
min     1.000000    1980.000000    0.000000    0.000000    0.000000
25%    4151.250000    2003.000000    0.000000    0.000000    0.000000
50%    8300.500000    2007.000000    0.080000    0.020000    0.000000
75%   12449.750000    2010.000000    0.240000    0.110000    0.040000
max   16600.000000    2020.000000   41.490000   29.020000   10.220000

```

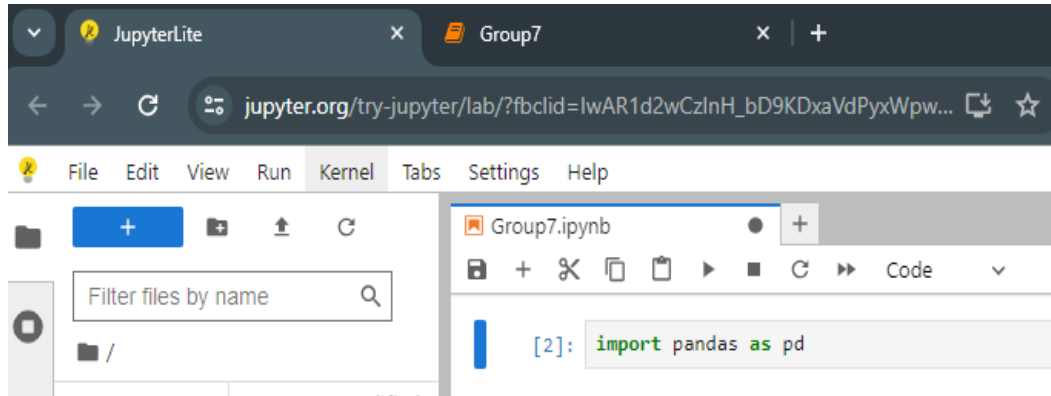
```

Other_Sales  Global_Sales
count 16598.000000 16598.000000
mean    0.048063    0.537441
std     0.188588    1.555028
min     0.000000    0.010000
25%     0.000000    0.060000
50%     0.010000    0.170000
75%     0.040000    0.470000
max    10.570000    82.740000
Mean Global Sales: 0.5409103185808114

```



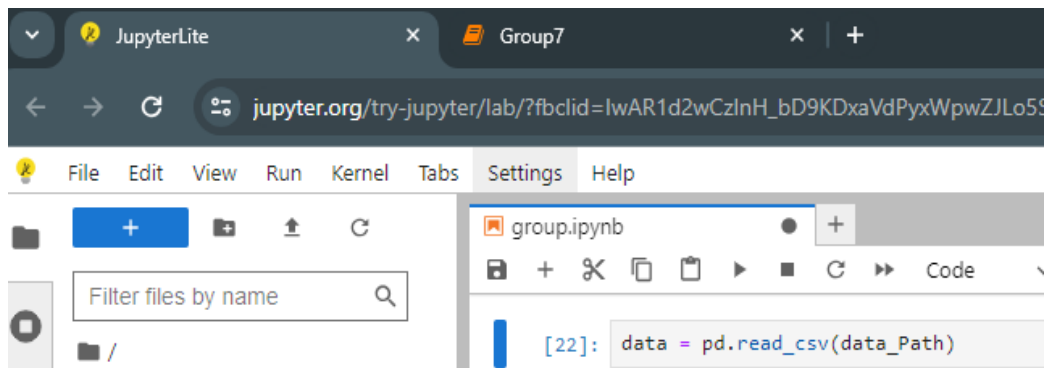
- Import libraries



- Finding data

```
data_path = "data.csv"
```

- Importing data



- Data attributes

```
In [36]: import pandas as pd
          df = pd.read_csv('vgsales.csv')
          df.shape

Out[36]: (16598, 11)
```


Group 7 Members:

Carange, Ansthrea

Leanillo, Herbert

Lopez, Katricia

Madlangsakay, Remo

Tiqui, Liezel