

NAME: SAMADHAN SANJAY KENDHALE

ROLL no: cs2-38

BATCH: C22

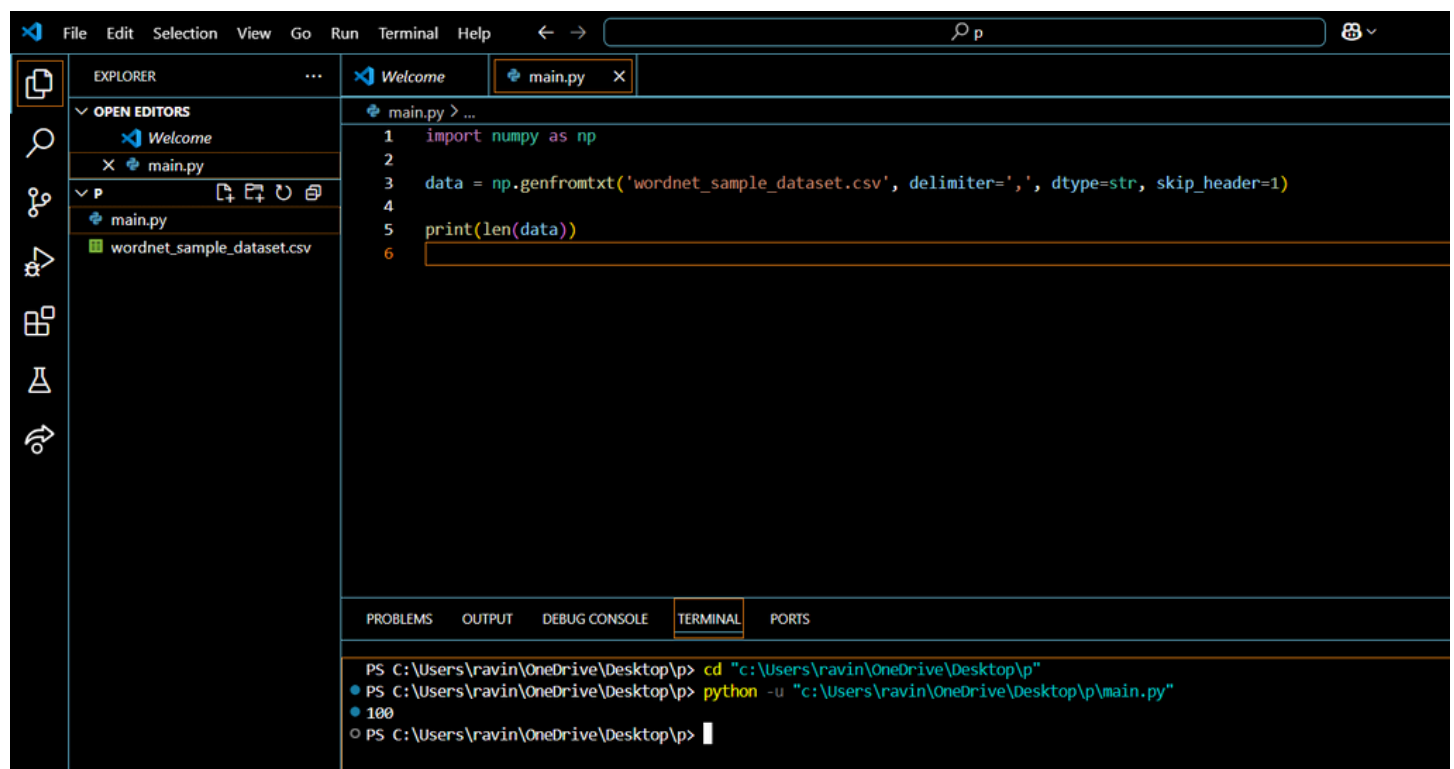
PRN: 202401040243

dataset: wordnet

numpy:

1. How many rows are there in the dataset?

solution:



The screenshot shows a Visual Studio Code editor window. The Explorer pane on the left shows the file structure with 'main.py' and 'wordnet_sample_dataset.csv'. The main editor area shows the following Python code in 'main.py':

```
1 import numpy as np
2
3 data = np.genfromtxt('wordnet_sample_dataset.csv', delimiter=',', dtype=str, skip_header=1)
4
5 print(len(data))
6
```

The bottom panel shows the TERMINAL output:

```
PS C:\Users\ravin\OneDrive\Desktop\p> cd "c:\Users\ravin\OneDrive\Desktop\p"
PS C:\Users\ravin\OneDrive\Desktop\p> python -u "c:\Users\ravin\OneDrive\Desktop\p\main.py"
100
PS C:\Users\ravin\OneDrive\Desktop\p>
```

2. Print the first 5 Synset Names.

solution

The screenshot shows the Visual Studio Code interface. The Explorer sidebar on the left shows the file structure with 'main.py' and 'wordnet_sample_dataset.csv' open. The main editor displays the following Python code in 'main.py':

```
1 import numpy as np
2
3 data = np.genfromtxt('wordnet_sample_dataset.csv', delimiter=',', dtype=str, skip_header=1)
4
5 print(data[:5, 0])
```

The bottom panel shows the TERMINAL output:

```
PS C:\Users\ravin\OneDrive\Desktop> cd "c:\Users\ravin\OneDrive\Desktop\p"
PS C:\Users\ravin\OneDrive\Desktop\p> python -u "c:\Users\ravin\OneDrive\Desktop\p\main.py"
['n0001' 'n0002' 'n0003' 'n0004' 'n0005']
PS C:\Users\ravin\OneDrive\Desktop\p>
```

`print(data[:5, 0])`

3. Print all POS tags.

solution:

The screenshot shows the Visual Studio Code interface. The Explorer sidebar on the left shows the file structure with 'main.py' and 'wordnet_sample_dataset.csv' open. The main editor displays the following Python code in 'main.py':

```
1 import numpy as np
2
3 data = np.genfromtxt('wordnet_sample_dataset.csv', delimiter=',', dtype=str, skip_header=1)
4
5 print(data[:, 6])
6
```

The bottom panel shows the TERMINAL output:

```
PS C:\Users\ravin\OneDrive\Desktop\p> cd "c:\Users\ravin\OneDrive\Desktop\p"
PS C:\Users\ravin\OneDrive\Desktop\p> python -u "c:\Users\ravin\OneDrive\Desktop\p\main.py"
['r' 'n' 'n' 'v' 'v' 'r' 'a' 'r' 'a' 'r' 'r' 'n' 'n' 'n' 'v' 'a' 'n'
'r' 'v' 'n' 'a' 'a' 'r' 'n' 'a' 'a' 'n' 'n' 'a' 'r' 'v' 'r' 'r' 'v' 'n'
'a' 'r' 'a' 'r' 'a' 'v' 'n' 'a' 'v' 'a' 'r' 'n' 'r' 'r' 'v' 'r' 'a' 'n'
'v' 'r' 'a' 'v' 'n' 'r' 'n' 'n' 'r' 'v' 'v' 'v' 'r' 'v' 'r' 'a' 'v' 'v'
'a' 'a' 'n' 'a' 'n' 'n' 'v' 'r' 'v' 'r' 'v' 'a' 'a' 'v' 'a' 'v' 'a' 'v'
'r' 'n' 'v' 'r' 'a' 'n' 'r' 'n' 'n' 'v']
PS C:\Users\ravin\OneDrive\Desktop\p>
```

`print(data[:, 6])`

4. Find how many times "n" appears in the POS column.

solution:

The screenshot shows the Visual Studio Code interface. The Explorer sidebar on the left shows the file structure with 'main.py' and 'wordnet_sample_dataset.csv' open. The main editor area displays the following Python code in 'main.py':

```
1 import numpy as np
2
3 data = np.genfromtxt('wordnet_sample_dataset.csv', delimiter=',', dtype=str, skip_header=1)
4
5 pos = data[:, 6]
6 print(np.sum(pos == "n"))
7
```

The bottom panel shows the TERMINAL tab with the following commands and output:

```
PS C:\Users\ravin\OneDrive\Desktop> cd "c:\Users\ravin\OneDrive\Desktop\p"
PS C:\Users\ravin\OneDrive\Desktop\p> python -u "c:\Users\ravin\OneDrive\Desktop\p\main.py"
24
PS C:\Users\ravin\OneDrive\Desktop\p>
```

pos = data[:, 6]

print(np.sum(pos == "n"))

5. Find how many Lemmas contain the word "car".

solution:

The screenshot shows the Visual Studio Code interface. The Explorer sidebar on the left shows the file structure with 'main.py' and 'wordnet_sample_dataset.csv' open. The main editor area displays the following Python code in 'main.py':

```
1 import numpy as np
2
3 data = np.genfromtxt('wordnet_sample_dataset.csv', delimiter=',', dtype=str, skip_header=1)
4
5 lemmas = data[:, 1]
6 count = np.sum(np.char.find(lemmas, "car") != -1)
7 print(count)
8
```

The bottom panel shows the TERMINAL tab with the following commands and output:

```
PS C:\Users\ravin\OneDrive\Desktop\p> cd "c:\Users\ravin\OneDrive\Desktop\p"
PS C:\Users\ravin\OneDrive\Desktop\p> python -u "c:\Users\ravin\OneDrive\Desktop\p\main.py"
32
PS C:\Users\ravin\OneDrive\Desktop\p>
```

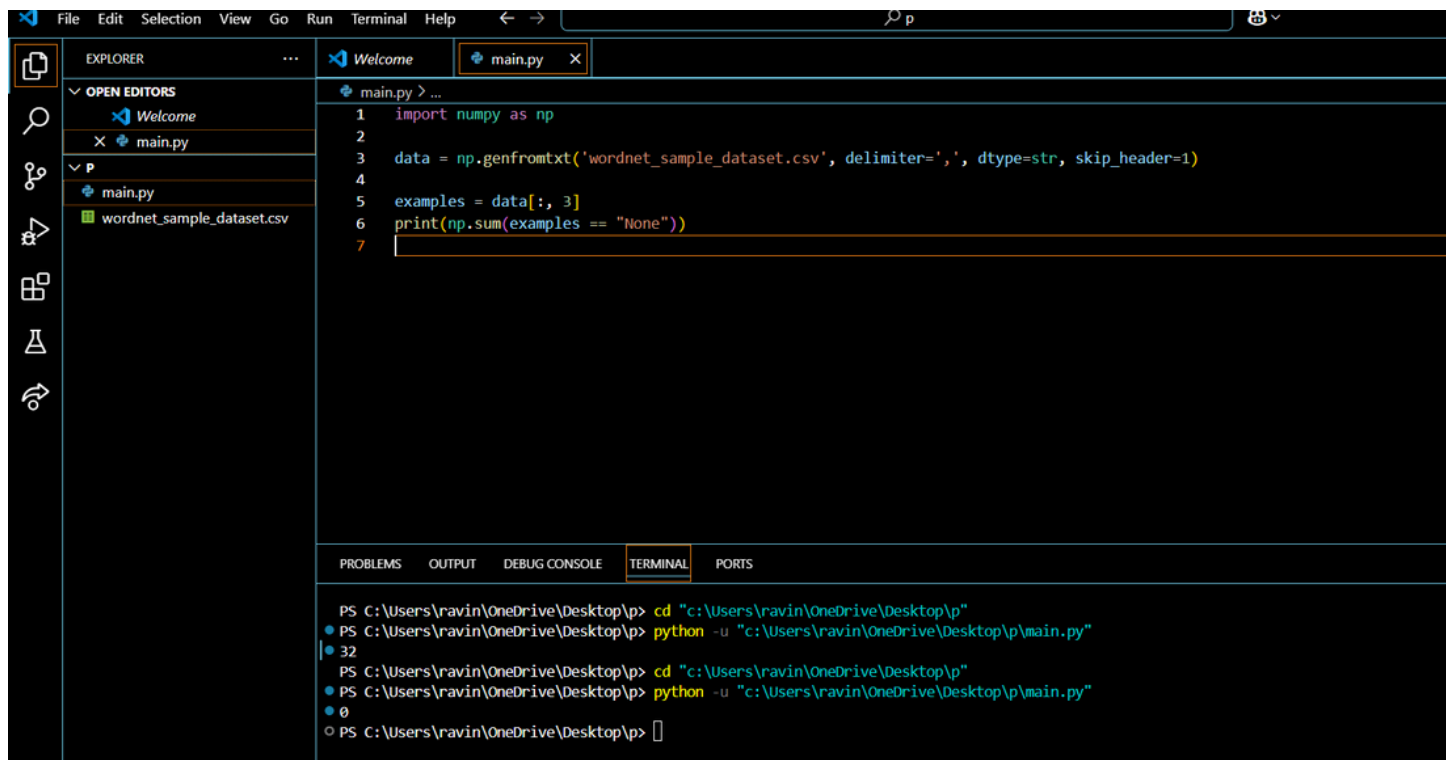
lemmas = data[:, 1]

count = np.sum(np.char.find(lemmas, "car") != -1)

print(count)

6. Find how many rows have "None" in Examples.

solution:



The screenshot shows the Visual Studio Code interface. The Explorer pane on the left shows the file structure with 'main.py' and 'wordnet_sample_dataset.csv' open. The main editor shows the following Python code in 'main.py':

```
1 import numpy as np
2
3 data = np.genfromtxt('wordnet_sample_dataset.csv', delimiter=',', dtype=str, skip_header=1)
4
5 examples = data[:, 3]
6 print(np.sum(examples == "None"))
7
```

The TERMINAL pane at the bottom shows the execution of the script:

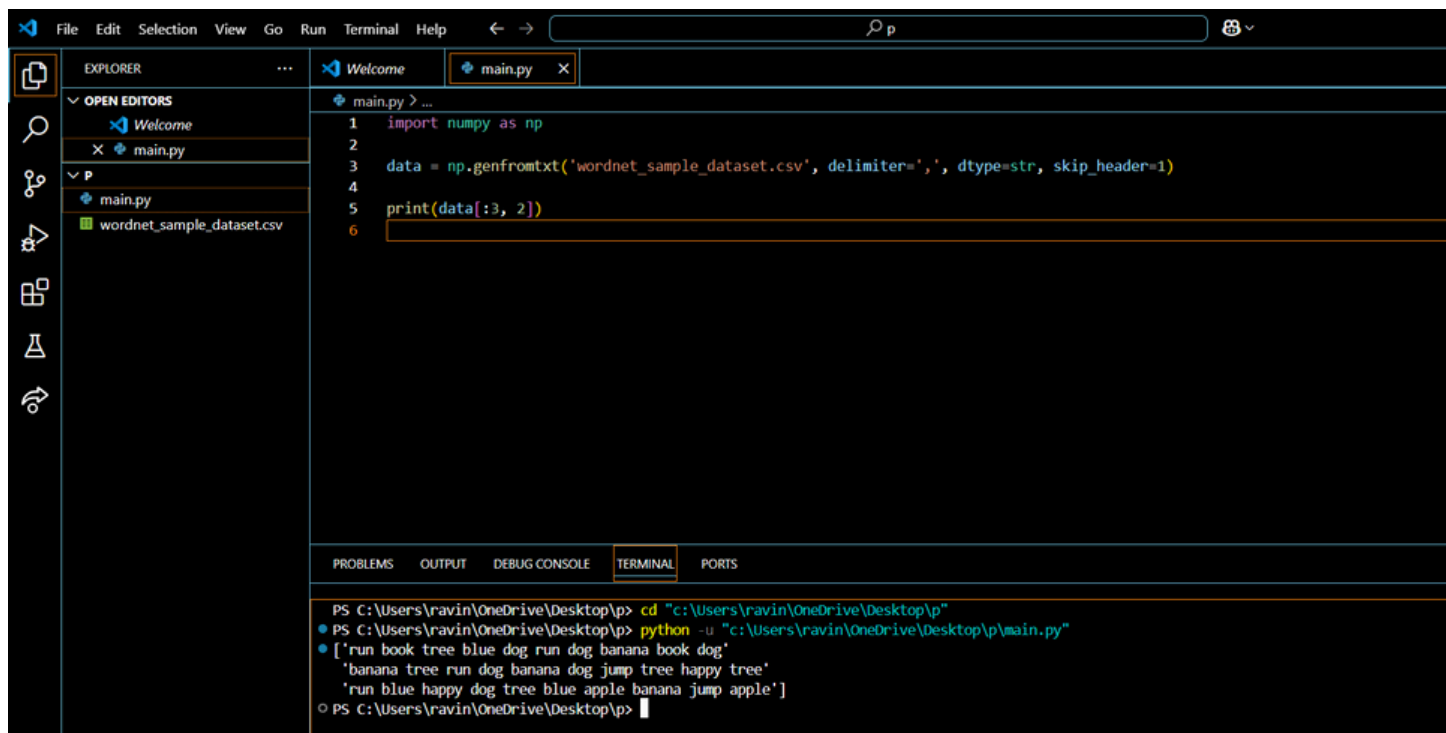
```
PS C:\Users\ravin\OneDrive\Desktop> cd "c:\Users\ravin\OneDrive\Desktop\p"
PS C:\Users\ravin\OneDrive\Desktop\p> python -u "c:\Users\ravin\OneDrive\Desktop\p\main.py"
32
PS C:\Users\ravin\OneDrive\Desktop\p> cd "c:\Users\ravin\OneDrive\Desktop\p"
PS C:\Users\ravin\OneDrive\Desktop\p> python -u "c:\Users\ravin\OneDrive\Desktop\p\main.py"
0
PS C:\Users\ravin\OneDrive\Desktop\p>
```

`examples = data[:, 3]`

`print(np.sum(examples == "None"))`

7. Print the first 3 Definitions.

solution:



The screenshot shows the Visual Studio Code interface. The Explorer pane on the left shows the file structure with 'main.py' and 'wordnet_sample_dataset.csv' open. The main editor shows the following Python code in 'main.py':

```
1 import numpy as np
2
3 data = np.genfromtxt('wordnet_sample_dataset.csv', delimiter=',', dtype=str, skip_header=1)
4
5 print(data[:3, 2])
6
```

The TERMINAL pane at the bottom shows the execution of the script:

```
PS C:\Users\ravin\OneDrive\Desktop\p> cd "c:\Users\ravin\OneDrive\Desktop\p"
PS C:\Users\ravin\OneDrive\Desktop\p> python -u "c:\Users\ravin\OneDrive\Desktop\p\main.py"
['run book tree blue dog run dog banana book dog'
 'banana tree run dog banana dog jump tree happy tree'
 'run blue happy dog tree blue apple banana jump apple']
PS C:\Users\ravin\OneDrive\Desktop\p>
```

`print(data[:3, 2])`

8. Print all unique POS tags.

solution:



The screenshot shows the VS Code interface with a file explorer on the left containing 'main.py' and 'wordnet_sample_dataset.csv'. The main editor window shows the following Python code in 'main.py':

```
1 import numpy as np
2
3 data = np.genfromtxt('wordnet_sample_dataset.csv', delimiter=',', dtype=str, skip_header=1)
4
5 unique_tags = np.unique(data[:, 6])
6 print(unique_tags)
7
```

The bottom panel shows the terminal with the following commands and output:

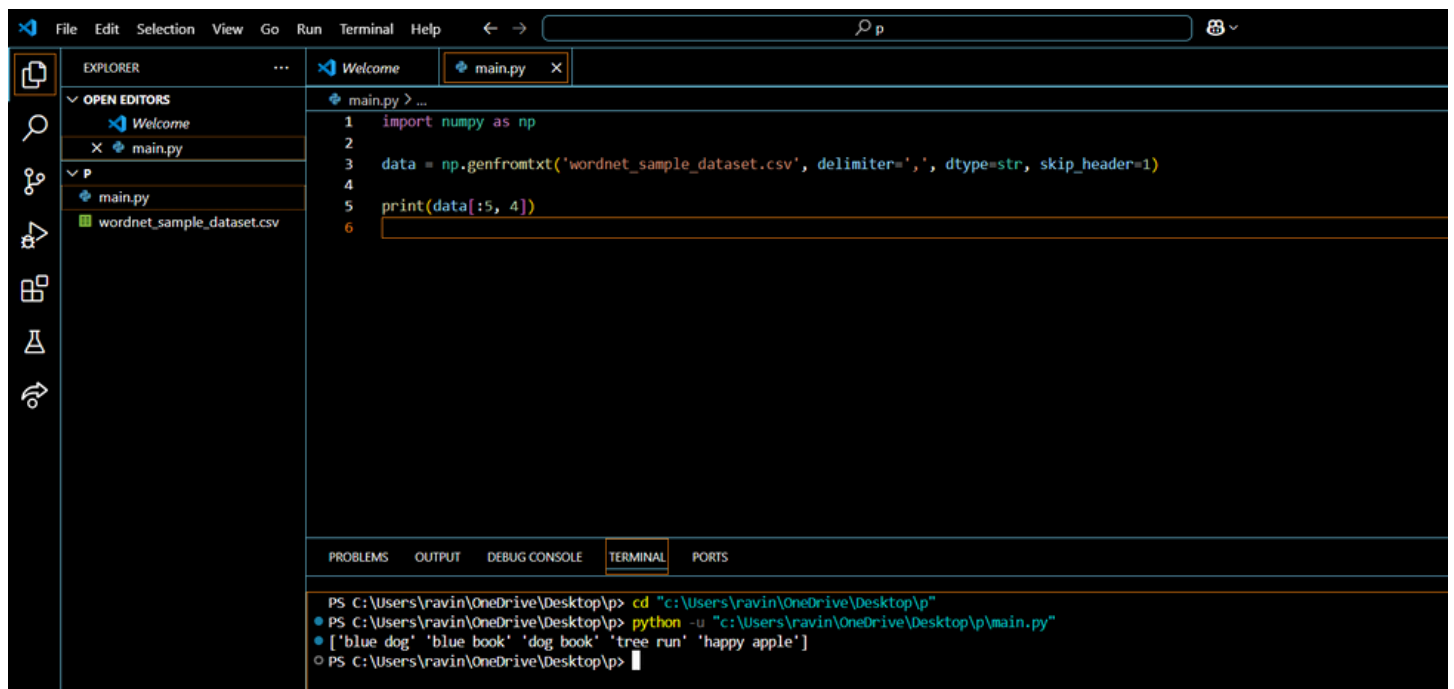
```
PS C:\Users\ravin\OneDrive\Desktop\p> cd "C:\Users\ravin\OneDrive\Desktop\p"
PS C:\Users\ravin\OneDrive\Desktop\p> python -u "C:\Users\ravin\OneDrive\Desktop\p\main.py"
['a' Focus folder in explorer (ctrl + click)
PS C:\Users\ravin\OneDrive\Desktop\p>
```

```
unique_tags = np.unique(data[:, 6])
```

```
print(unique_tags)
```

9. Print the first 5 Hypernyms.

solution:



The screenshot shows the VS Code interface with a file explorer on the left containing 'main.py' and 'wordnet_sample_dataset.csv'. The main editor window shows the following Python code in 'main.py':

```
1 import numpy as np
2
3 data = np.genfromtxt('wordnet_sample_dataset.csv', delimiter=',', dtype=str, skip_header=1)
4
5 print(data[:5, 4])
6
```

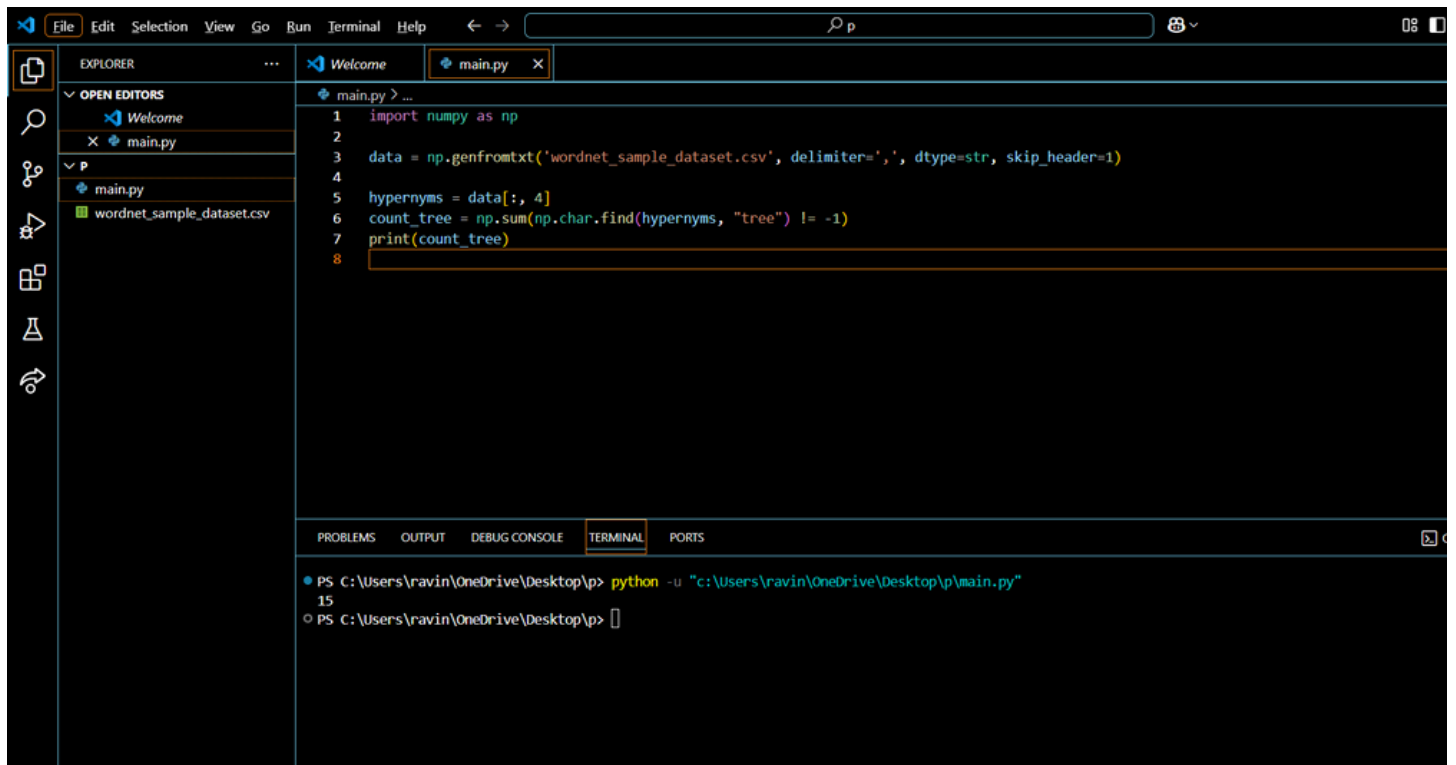
The bottom panel shows the terminal with the following commands and output:

```
PS C:\Users\ravin\OneDrive\Desktop\p> cd "C:\Users\ravin\OneDrive\Desktop\p"
PS C:\Users\ravin\OneDrive\Desktop\p> python -u "C:\Users\ravin\OneDrive\Desktop\p\main.py"
['blue dog' 'blue book' 'dog book' 'tree run' 'happy apple']
PS C:\Users\ravin\OneDrive\Desktop\p>
```

```
print(data[:5, 4])
```

10. How many rows have the word "tree" anywhere in Hypernyms?

solution:



The screenshot shows a Visual Studio Code editor window. The Explorer pane on the left shows a project with two files: `main.py` and `wordnet_sample_dataset.csv`. The main editor area displays the contents of `main.py`, which is a Python script using NumPy to read a CSV file and count occurrences of the word "tree" in the hypernym column. The script is as follows:

```
1 import numpy as np
2
3 data = np.genfromtxt('wordnet_sample_dataset.csv', delimiter=',', dtype=str, skip_header=1)
4
5 hypernoms = data[:, 4]
6 count_tree = np.sum(np.char.find(hypernoms, "tree") != -1)
7 print(count_tree)
8
```

Below the editor, the TERMINAL pane shows the command to run the script and its output:

```
PS C:\Users\ravin\OneDrive\Desktop\p> python -u "c:\Users\ravin\OneDrive\Desktop\p\main.py"
15
PS C:\Users\ravin\OneDrive\Desktop\p>
```

```
hypernoms = data[:, 4]
```

```
count_tree = np.sum(np.char.find(hypernoms, "tree") != -1)
```

```
print(count_tree)
```

pandas:

1. Show the column names of the dataset.

solution:

The screenshot shows a Visual Studio Code editor window. The Explorer sidebar on the left shows a file named `wordnet_sample_dataset.csv` under a folder `P`. The main editor area displays a Python file `main.py` with the following code:

```
1 import pandas as pd
2
3 df = pd.read_csv('wordnet_sample_dataset.csv')
4
5 print(df.columns)
6
```

Below the editor, the TERMINAL panel is active, showing the command prompt output:

```
PS C:\Users\ravin\OneDrive\Desktop\p> cd "c:\Users\ravin\OneDrive\Desktop\p"
PS C:\Users\ravin\OneDrive\Desktop\p> python -u "c:\Users\ravin\OneDrive\Desktop\p\main.py"
Index(['Synset Name', 'Lemmas', 'Definition', 'Examples', 'Hypernyms',
      'Hyponyms', 'POS'],
      dtype='object')
PS C:\Users\ravin\OneDrive\Desktop\p>
```

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

```
print(df.columns)
```

2. Show the datatype (dtype) of each column.

solution:

The screenshot shows the VS Code interface with a Python file named `main.py` open in the editor. The file contains the following code:

```
1 import pandas as pd
2
3 df = pd.read_csv('wordnet_sample_dataset.csv')
4
5 print(df.dtypes)
6
```

The Explorer sidebar on the left shows the file structure with `main.py` and `wordnet_sample_dataset.csv`. The Terminal panel at the bottom shows the output of running the script:

```
PS C:\Users\ravin\OneDrive\Desktop> cd "c:\Users\ravin\OneDrive\Desktop\p"
PS C:\Users\ravin\OneDrive\Desktop\p> python -u "c:\Users\ravin\OneDrive\Desktop\p\main.py"
Index(['Synset Name', 'Lemmas', 'Definition', 'Examples', 'Hypernyms',
      'Hyponyms', 'POS'],
      dtype='object')
PS C:\Users\ravin\OneDrive\Desktop\p> cd "c:\Users\ravin\OneDrive\Desktop\p"
PS C:\Users\ravin\OneDrive\Desktop\p> python -u "c:\Users\ravin\OneDrive\Desktop\p\main.py"
Synset Name    object
Lemmas         object
Definition     object
Examples       object
Hypernyms      object
Hyponyms       object
POS            object
dtype: object
PS C:\Users\ravin\OneDrive\Desktop\p>
```

`print(df.dtypes)`

3. Find the number of missing (empty) values in each column.

solution:

The screenshot shows the VS Code interface with a Python file named `main.py` open in the editor. The file contains the following code:

```
1 import pandas as pd
2
3 df = pd.read_csv('wordnet_sample_dataset.csv')
4
5 print(df.isnull().sum())
6
```

The Explorer sidebar on the left shows the file structure with `main.py` and `wordnet_sample_dataset.csv`. The Terminal panel at the bottom shows the output of running the script:

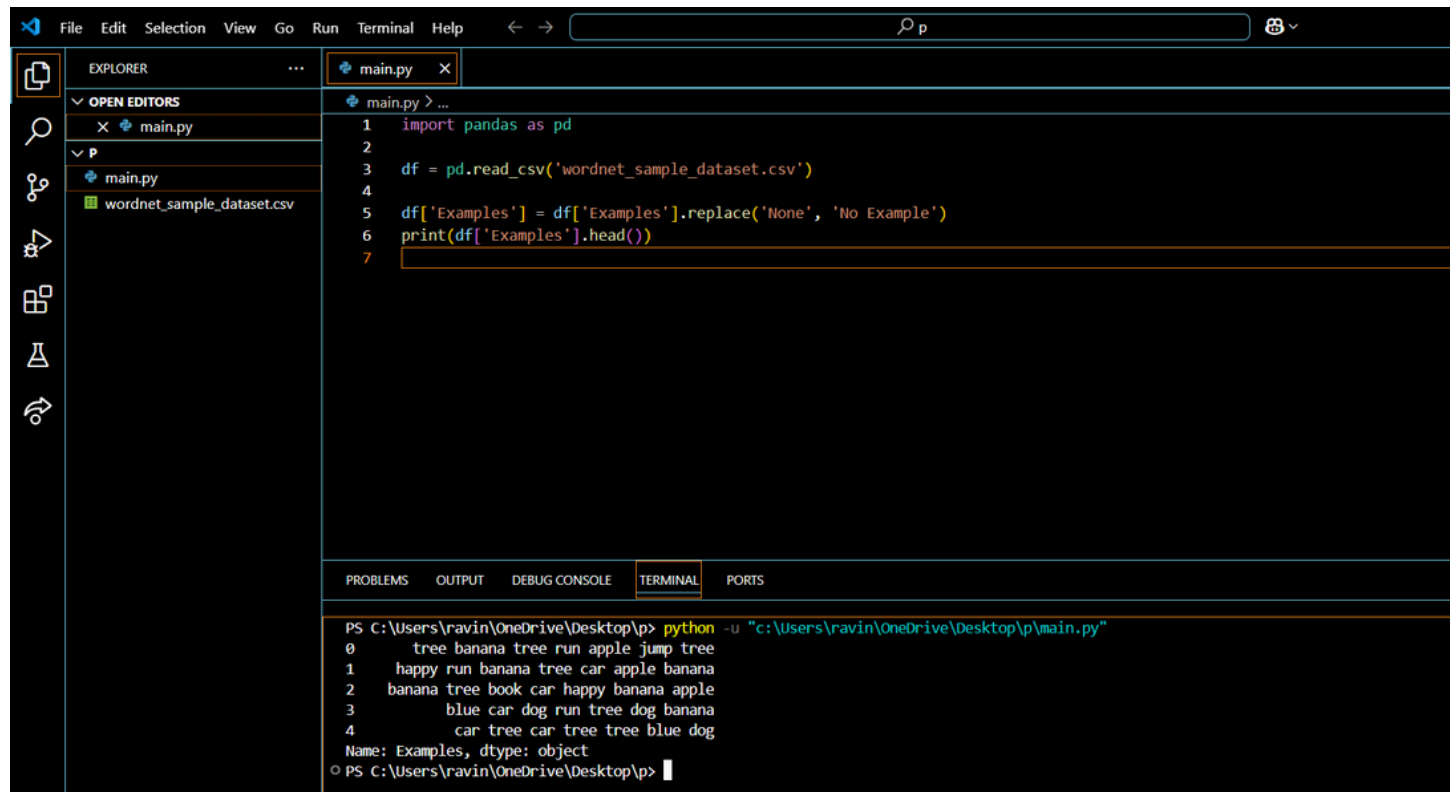
```
PS C:\Users\ravin\OneDrive\Desktop\p> python -u "c:\Users\ravin\OneDrive\Desktop\p\main.py"
Synset Name    0
Lemmas         0
Definition     0
Examples       0
Hypernyms      0
Hyponyms       0
POS            0
dtype: int64
PS C:\Users\ravin\OneDrive\Desktop\p>
```



```
print(df.isnull().sum())
```

4. Replace all "None" values in Examples with "No Example".

solution:



The screenshot shows a Visual Studio Code editor window. The Explorer pane on the left shows a project with two files: `main.py` and `wordnet_sample_dataset.csv`. The main editor displays the contents of `main.py`, which is a Python script that imports pandas, reads a CSV file, replaces 'None' values in the 'Examples' column with 'No Example', and prints the first five rows of the resulting DataFrame. The Terminal pane at the bottom shows the command `python -u "c:\Users\ravin\OneDrive\Desktop\p\main.py"` being executed, followed by the output of the script, which displays five rows of data from the 'Examples' column.

```
1 import pandas as pd
2
3 df = pd.read_csv('wordnet_sample_dataset.csv')
4
5 df['Examples'] = df['Examples'].replace('None', 'No Example')
6 print(df['Examples'].head())
7
```

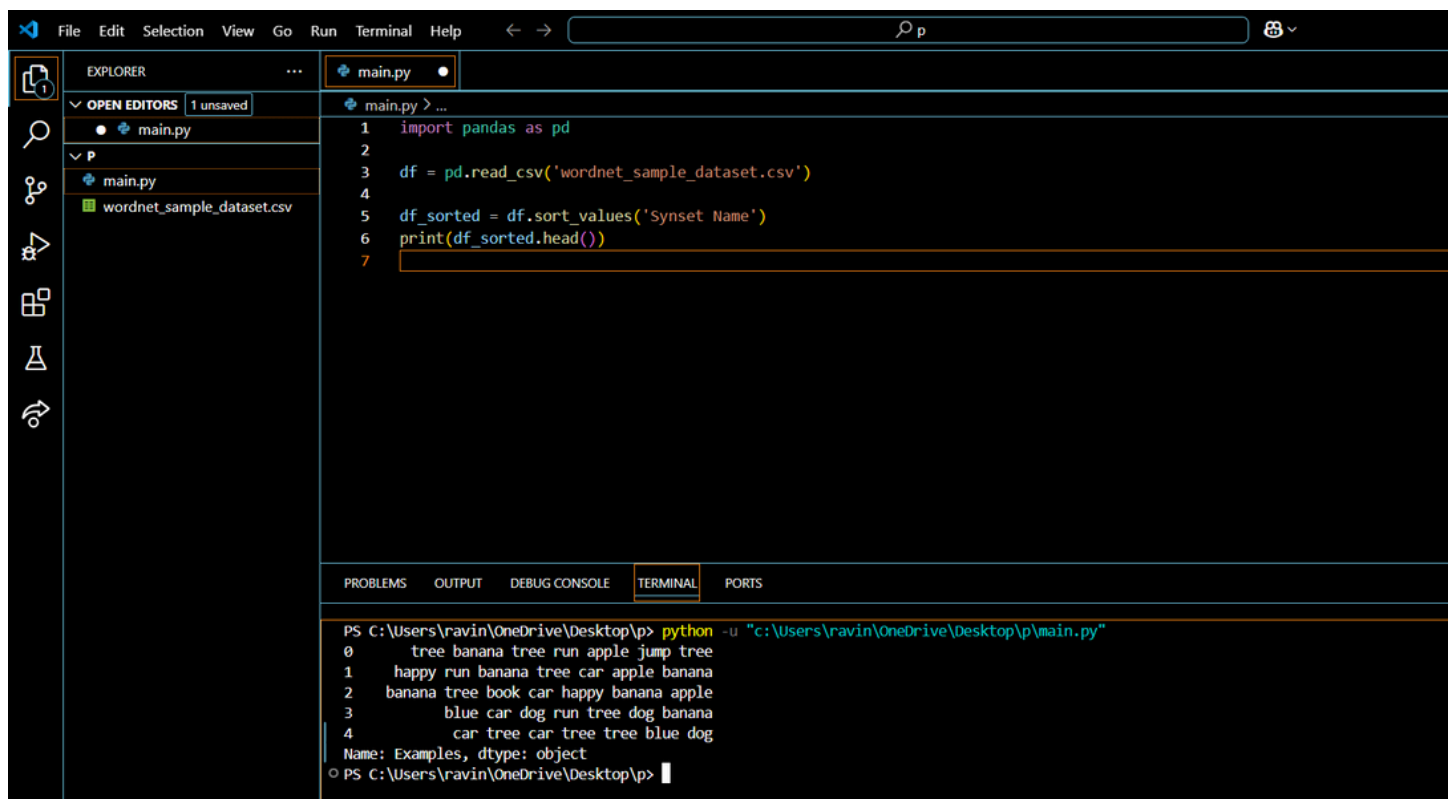
```
PS C:\Users\ravin\OneDrive\Desktop\p> python -u "c:\Users\ravin\OneDrive\Desktop\p\main.py"
0    tree banana tree run apple jump tree
1    happy run banana tree car apple banana
2    banana tree book car happy banana apple
3    blue car dog run tree dog banana
4    car tree car tree tree blue dog
Name: Examples, dtype: object
PS C:\Users\ravin\OneDrive\Desktop\p>
```

```
df['Examples'] = df['Examples'].replace('None', 'No Example')
```

```
print(df["Examples"].head())
```

5. Sort the dataset by "Synset Name" alphabetically.

solution:



The screenshot shows the Visual Studio Code interface. The Explorer pane on the left shows the file structure with 'main.py' and 'wordnet_sample_dataset.csv'. The main editor displays the following Python code in 'main.py':

```
1 import pandas as pd
2
3 df = pd.read_csv('wordnet_sample_dataset.csv')
4
5 df_sorted = df.sort_values('Synset Name')
6 print(df_sorted.head())
7
```

The TERMINAL pane at the bottom shows the command prompt output:

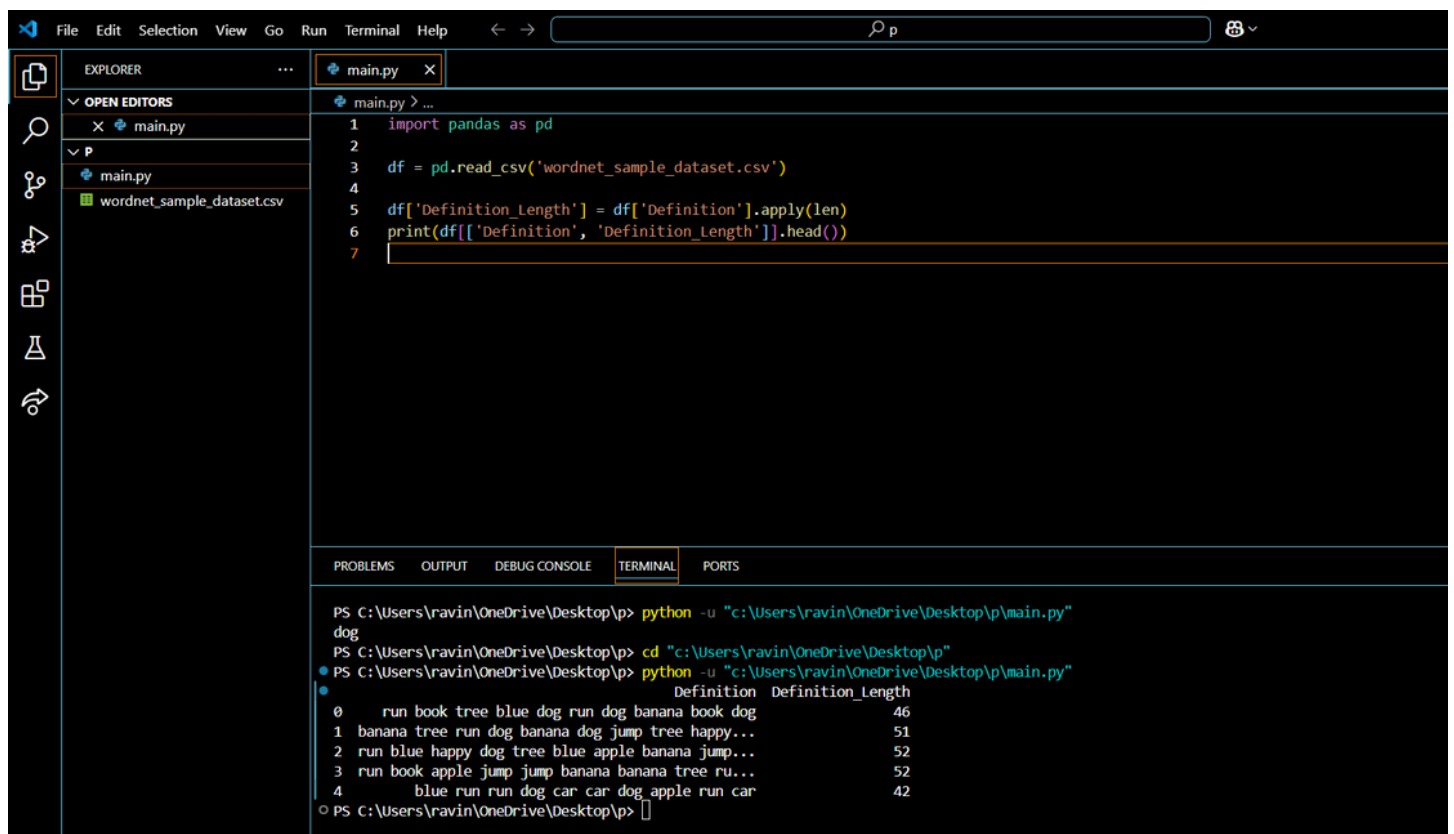
```
PS C:\Users\ravin\OneDrive\Desktop> python -u "c:\Users\ravin\OneDrive\Desktop\p\main.py"
0      tree banana tree run apple jump tree
1    happy run banana tree car apple banana
2    banana tree book car happy banana apple
3      blue car dog run tree dog banana
4      car tree car tree tree blue dog
Name: Examples, dtype: object
PS C:\Users\ravin\OneDrive\Desktop>
```

`df_sorted = df.sort_values('Synset Name')`

`print(df_sorted.head())`

6. Create a new column called "Definition_Length" that stores the number of characters in each definition.

solution:



The screenshot shows the Visual Studio Code interface with the updated Python script in 'main.py':

```
1 import pandas as pd
2
3 df = pd.read_csv('wordnet_sample_dataset.csv')
4
5 df['Definition_Length'] = df['Definition'].apply(len)
6 print(df[['Definition', 'Definition_Length']].head())
7
```

The TERMINAL pane shows the command prompt output:

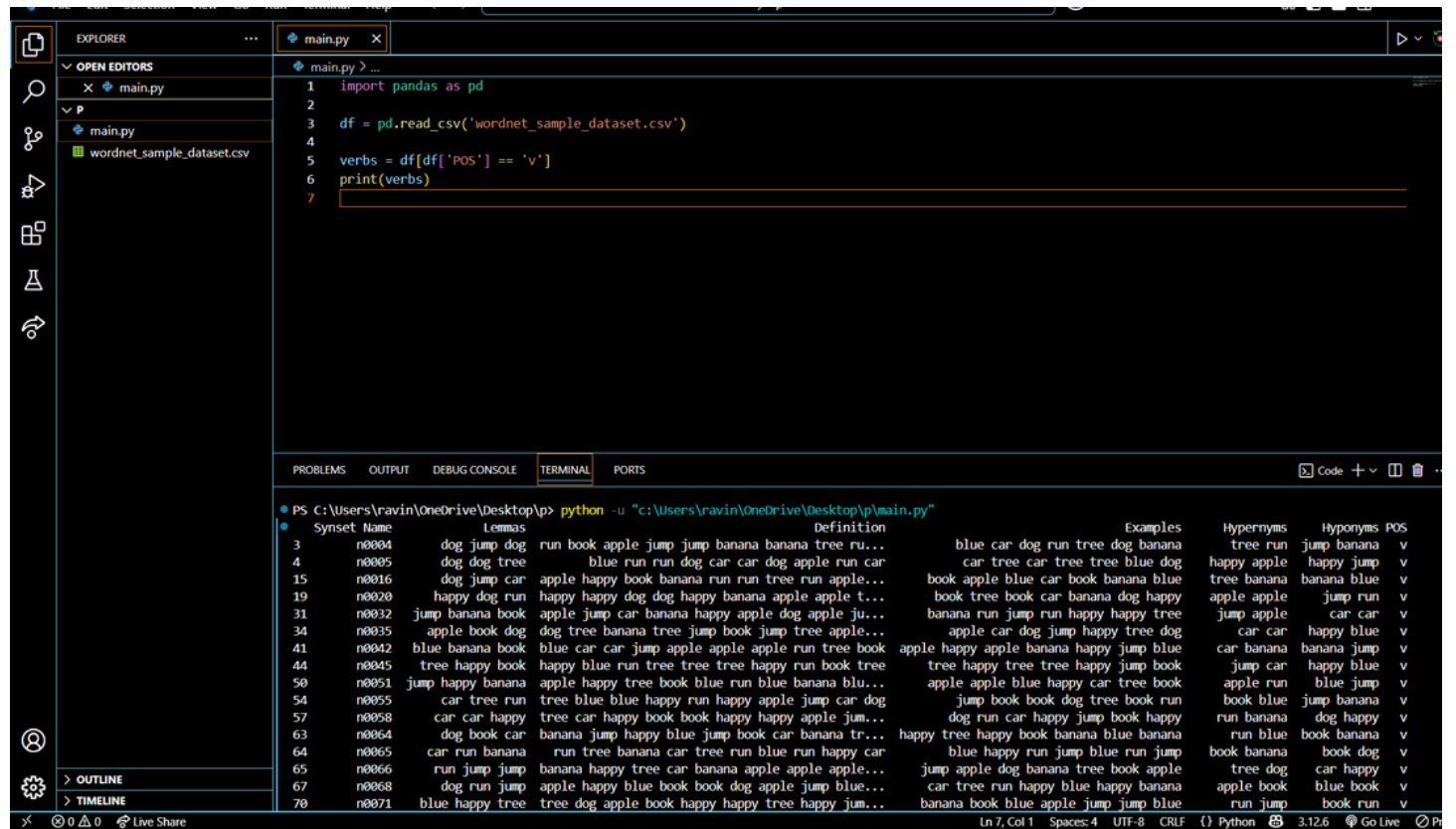
```
PS C:\Users\ravin\OneDrive\Desktop> python -u "c:\Users\ravin\OneDrive\Desktop\p\main.py"
dog
PS C:\Users\ravin\OneDrive\Desktop> cd "c:\Users\ravin\OneDrive\Desktop\p"
PS C:\Users\ravin\OneDrive\Desktop> python -u "c:\Users\ravin\OneDrive\Desktop\p\main.py"
0      Definition  Definition_Length
0    run book tree blue dog run dog banana book dog          46
1  banana tree run dog banana dog jump tree happy...         51
2  run blue happy dog tree blue apple banana jump...         52
3  run book apple jump jump banana banana tree ru...         52
4    blue run run dog car car dog apple run car           42
PS C:\Users\ravin\OneDrive\Desktop>
```

```
df['Definition_Length'] = df['Definition'].apply(len)
```

```
print(df[['Definition', 'Definition_Length']].head())
```

7. Filter and show all rows where POS is "v" (verb).

solution:



The screenshot shows a VS Code editor with a file named `main.py` open. The script reads a CSV file `wordnet_sample_dataset.csv` and filters rows where the POS is 'v'. The terminal output shows the first 70 rows of the filtered dataset, which are verbs.

```
1 import pandas as pd
2
3 df = pd.read_csv('wordnet_sample_dataset.csv')
4
5 verbs = df[df['POS'] == 'v']
6 print(verbs)
7
```

Synset Name	Lemmas	Definition	Examples	Hypernyms	Hyponyms	POS
n0004	dog jump dog	run book apple jump jump banana banana tree ru...	blue car dog run tree dog banana	tree run	jump banana	v
n0005	dog dog tree	blue run run dog car car dog apple run car	car tree car tree tree blue dog	happy apple	happy jump	v
n0016	dog jump car	apple happy book banana run run tree run apple...	book apple blue car book banana blue	tree banana	banana blue	v
n0020	happy dog run	happy happy dog dog happy banana apple apple t...	book tree book car banana dog happy	apple apple	jump run	v
n0032	jump banana book	apple jump car banana happy apple dog apple ju...	banana run jump run happy happy tree	jump apple	car car	v
n0035	apple book dog	dog tree banana tree jump book jump tree apple...	apple car dog jump happy tree dog	car car	happy blue	v
n0042	blue banana book	blue car car jump apple apple apple run tree book	apple happy apple banana happy jump blue	car banana	banana jump	v
n0045	tree happy book	happy blue run tree tree tree happy run book tree	tree happy tree tree happy jump book	jump car	happy blue	v
n0051	jump happy banana	apple happy tree book blue run blue banana blu...	apple apple blue happy car tree book	apple run	blue jump	v
n0055	car tree run	tree blue blue happy run happy apple jump car dog	jump book book dog tree book run	book blue	jump banana	v
n0058	car car happy	tree car happy book book happy happy apple jum...	dog run car happy jump book happy	run banana	dog happy	v
n0064	dog book car	banana jump happy blue jump book car banana tr...	happy tree happy book banana blue banana	run blue	book banana	v
n0065	car run banana	run tree banana car tree run blue run happy car	blue happy run jump blue run jump	book banana	book dog	v
n0066	run jump jump	banana happy tree car banana apple apple apple...	jump apple dog banana tree book apple	tree dog	car happy	v
n0068	dog run jump	apple happy blue book book dog apple jump blue...	car tree run happy blue happy banana	apple book	blue book	v
n0071	blue happy tree	tree dog apple book happy happy tree happy jum...	banana book blue apple jump jump blue	run jump	book run	v

```
verbs = df[df['POS'] == 'v']
```

```
print(verbs)
```

8. Find the most common word appearing in Lemmas.

solution:

The screenshot shows the Visual Studio Code interface. The Explorer panel on the left shows the file structure with 'main.py' and 'wordnet_sample_dataset.csv'. The main editor area displays the following Python code in 'main.py':

```
1 import pandas as pd
2
3 df = pd.read_csv('wordnet_sample_dataset.csv')
4
5 all_lemmas = " ".join(df['Lemmas']).split()
6 most_common = pd.Series(all_lemmas).value_counts().idxmax()
7 print(most_common)
8
```

The TERMINAL panel at the bottom shows the command prompt output:

```
PS C:\Users\ravin\OneDrive\Desktop\p> python -u "c:\Users\ravin\OneDrive\Desktop\p\main.py"
dog
PS C:\Users\ravin\OneDrive\Desktop\p>
```

`all_lemmas = " ".join(df['Lemmas']).split()`

`most_common = pd.Series(all_lemmas).value_counts().idxmax()`

`print(most_common)`

9. Find the maximum length of the Hypernyms field.

solution:

The screenshot shows the Visual Studio Code interface. The Explorer panel on the left shows the file structure with 'main.py' and 'wordnet_sample_dataset.csv'. The main editor area displays the following Python code in 'main.py':

```
1 import pandas as pd
2
3 df = pd.read_csv('wordnet_sample_dataset.csv')
4
5 max_hypernym_length = df['Hypernyms'].apply(len).max()
6 print(max_hypernym_length)
7
```

The TERMINAL panel at the bottom shows the command prompt output:

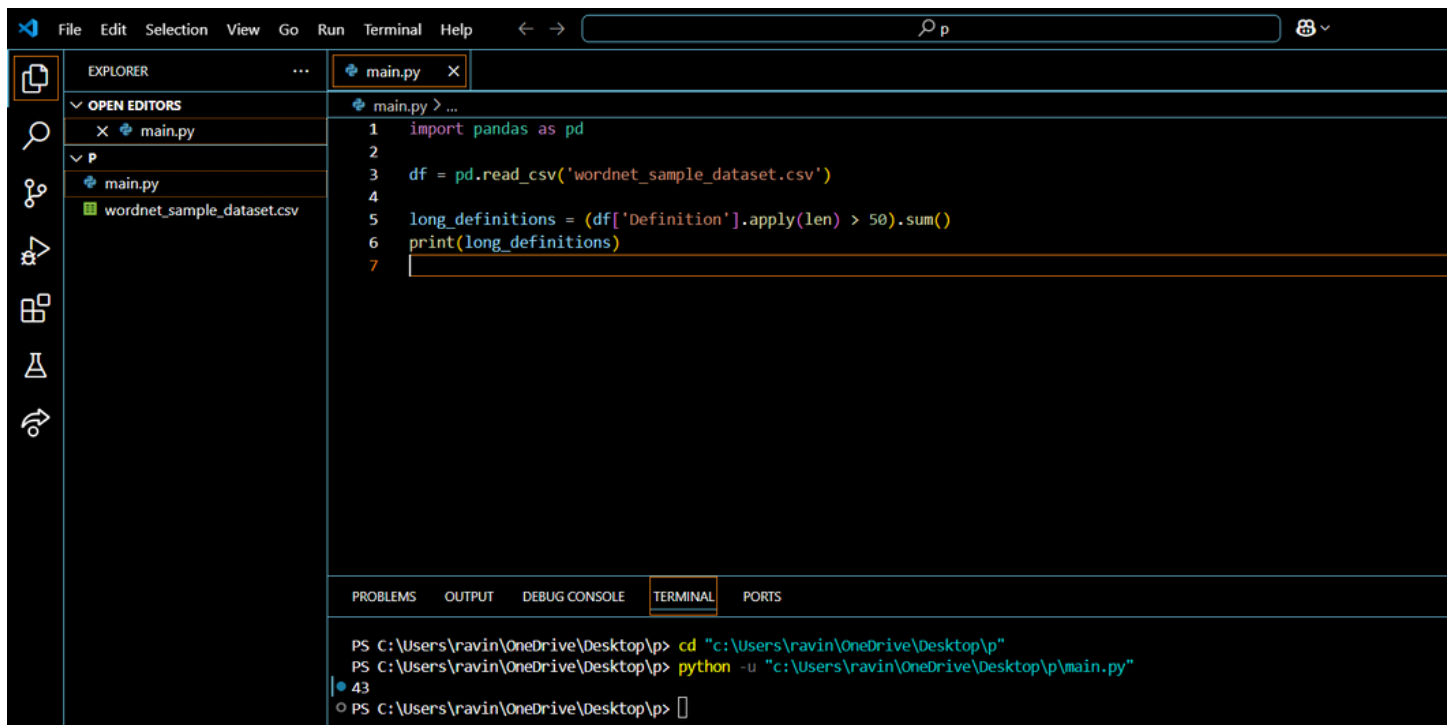
```
PS C:\Users\ravin\OneDrive\Desktop\p> cd "c:\Users\ravin\OneDrive\Desktop\p"
PS C:\Users\ravin\OneDrive\Desktop\p> python -u "c:\Users\ravin\OneDrive\Desktop\p\main.py"
13
PS C:\Users\ravin\OneDrive\Desktop\p>
```

`max_hypernym_length = df['Hypernyms'].apply(len).max()`

`print(max_hypernym_length)`

10. Find how many Definitions have more than 50 characters.

solution:



The screenshot shows the Visual Studio Code interface. The Explorer sidebar on the left shows a file named `wordnet_sample_dataset.csv` and a Python file named `main.py`. The main editor window displays the following Python code in `main.py`:

```
1 import pandas as pd
2
3 df = pd.read_csv('wordnet_sample_dataset.csv')
4
5 long_definitions = (df['Definition'].apply(len) > 50).sum()
6 print(long_definitions)
7
```

Below the editor, the TERMINAL panel is active, showing the command prompt output:

```
PS C:\Users\ravin\OneDrive\Desktop\p> cd "c:\Users\ravin\OneDrive\Desktop\p"
PS C:\Users\ravin\OneDrive\Desktop\p> python -u "c:\Users\ravin\OneDrive\Desktop\p\main.py"
43
PS C:\Users\ravin\OneDrive\Desktop\p>
```

```
long_definitions = (df['Definition'].apply(len) > 50).sum()
```

```
print(long_definitions)
```

---end---