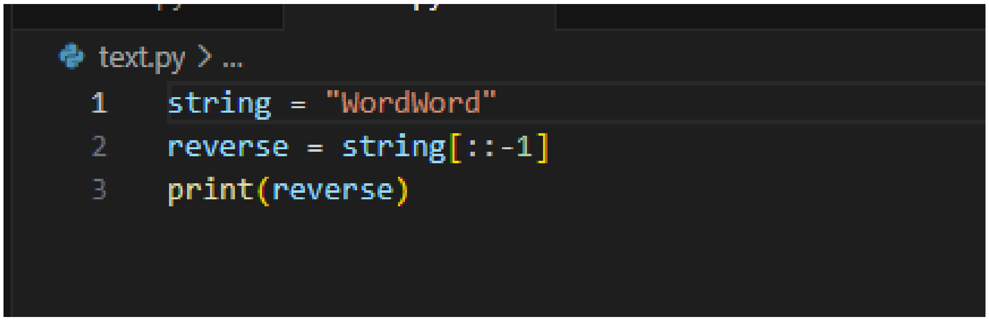
Name: Sownthari R P

Date: 12.08.2024

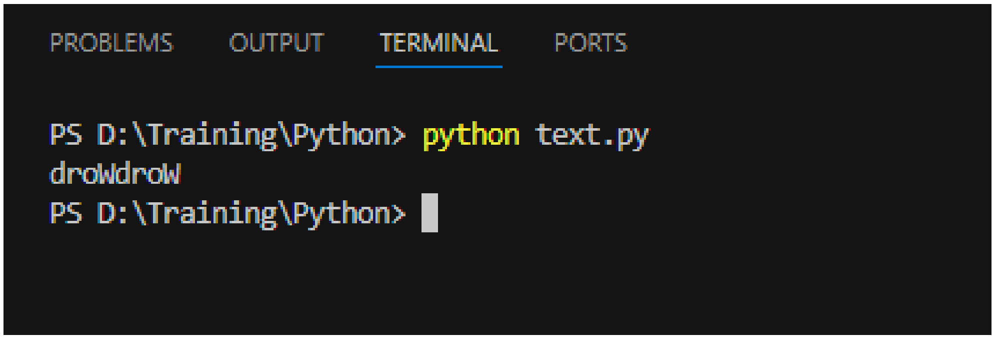
1.Reverse a string “WorldWord”.

Program:



```python
string = "WordWord"
reverse = string[::-1]
print(reverse)
```

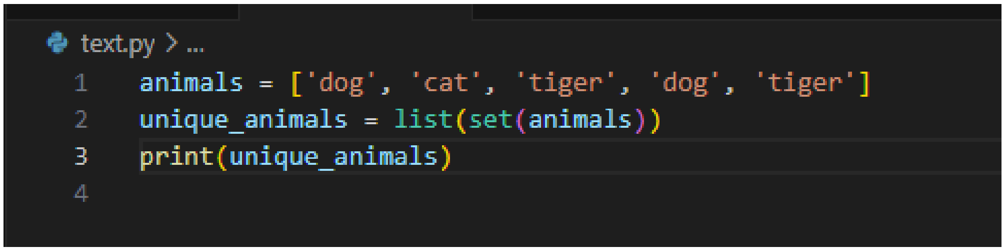
Output:



```
PS D:\Training\Python> python text.py
droWdroW
PS D:\Training\Python>
```

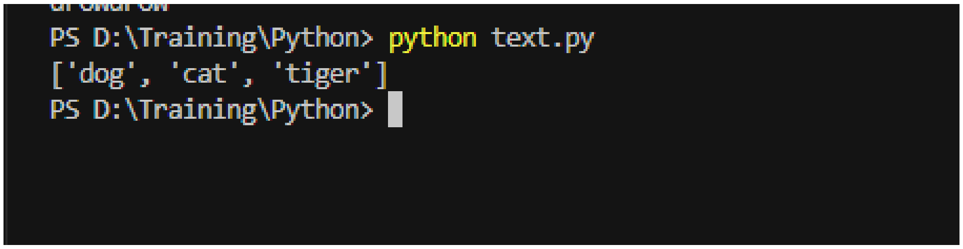
2.Remove duplicates in [’dog’,’cat’,’tiger’,’dog’, ‘tiger’]

Program:



```python
animals = ['dog', 'cat', 'tiger', 'dog', 'tiger']
unique_animals = list(set(animals))
print(unique_animals)

```

Output:

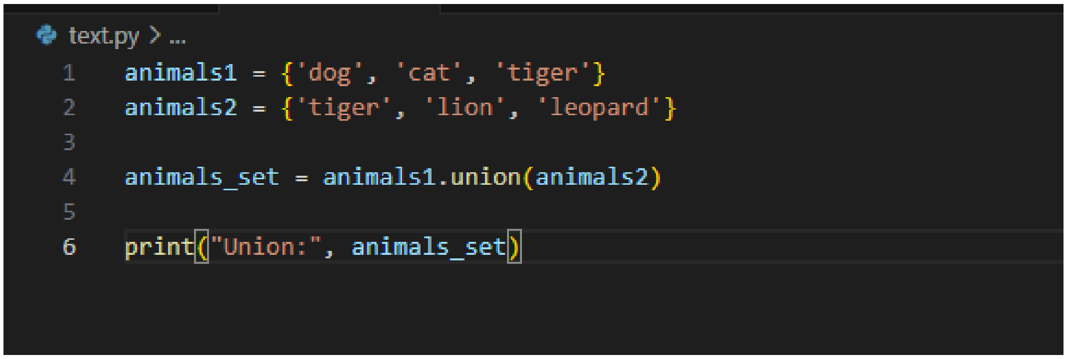


```
PS D:\Training\Python> python text.py
['dog', 'cat', 'tiger']
PS D:\Training\Python>
```

3.Perform union and intersection using Set

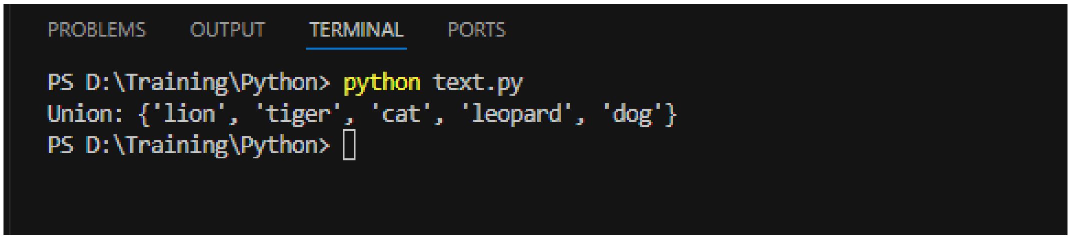
Union:

Program:



```python
animals1 = {'dog', 'cat', 'tiger'}
animals2 = {'tiger', 'lion', 'leopard'}

animals_set = animals1.union(animals2)

print("Union:", animals_set)
```

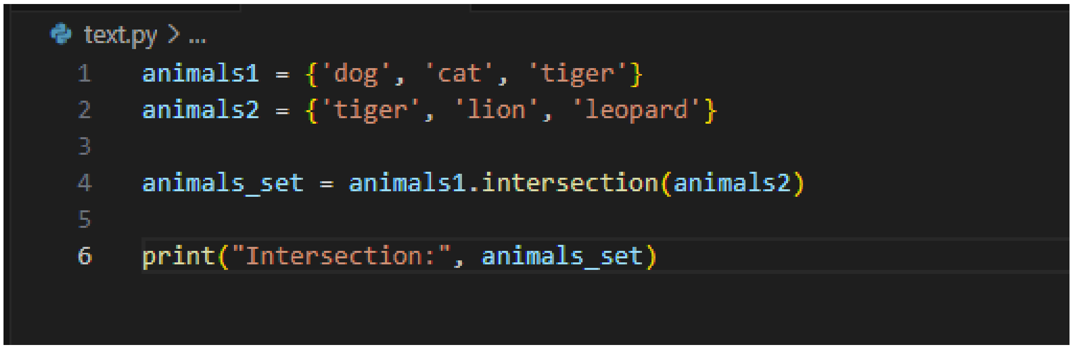
Output:



```
PS D:\Training\Python> python text.py
Union: {'lion', 'tiger', 'cat', 'leopard', 'dog'}
PS D:\Training\Python>
```

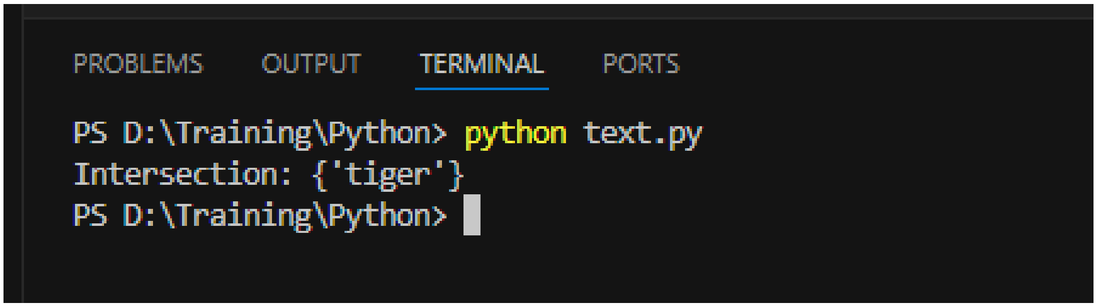
Intersection:

Program:



```python
animals1 = {'dog', 'cat', 'tiger'}
animals2 = {'tiger', 'lion', 'leopard'}

animals_set = animals1.intersection(animals2)

print("Intersection:", animals_set)
```

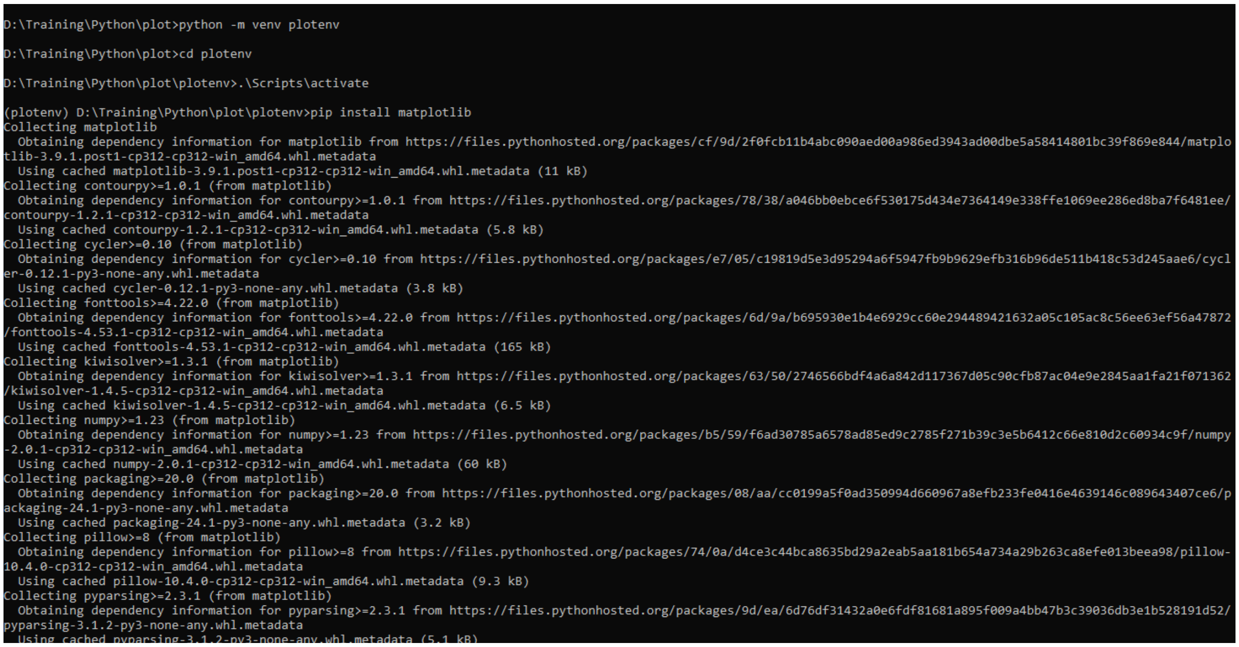
Output:



```
PS D:\Training\Python> python text.py
Intersection: {'tiger'}
PS D:\Training\Python> 
```

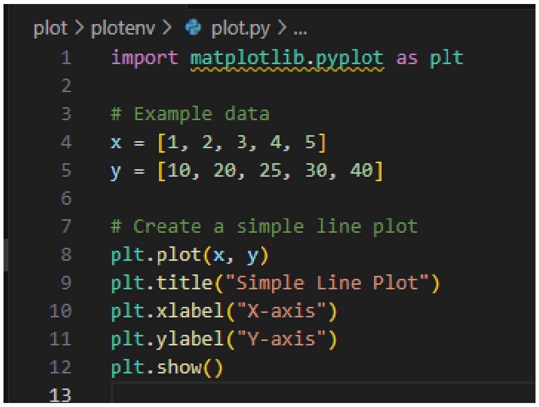
4.Create virtual environment and show installation of package matplotlib and import of modules for visualization.

i.Created environment plotenv and installed matplotlib



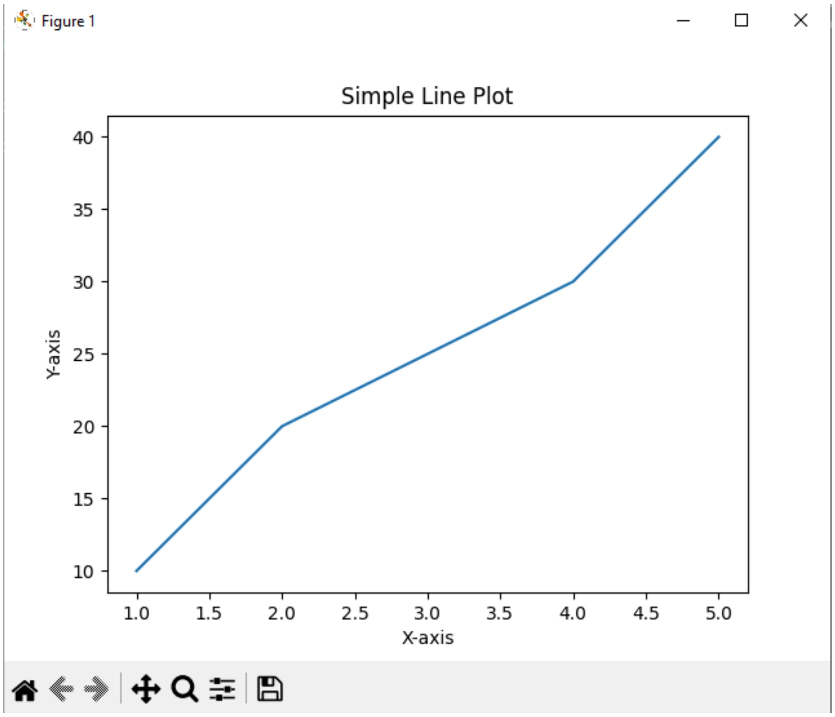
```
D:\Training\Python\plot>python -m venv plotenv

D:\Training\Python\plot>cd plotenv

D:\Training\Python\plot\plotenv>.\Scripts\activate

(plotenv) D:\Training\Python\plot\plotenv>pip install matplotlib
Collecting matplotlib
  Obtaining dependency information for matplotlib from https://files.pythonhosted.org/packages/cf/9d/2f0fcb11b4abc090aed00a986ed3943ad00dbe5a58414801bc39f869e844/matplo
tlib-3.9.1.post1-cp312-cp312-win_amd64.whl.metadata
  Using cached matplotlib-3.9.1.post1-cp312-cp312-win_amd64.whl.metadata (11 kB)
Collecting contourpy>=1.0.1 (from matplotlib)
  Obtaining dependency information for contourpy>=1.0.1 from https://files.pythonhosted.org/packages/78/38/a046bb0ebce6f530175d434e7364149e338ffe1069ee286ed8ba7f6481ee/
contourpy-1.2.1-cp312-cp312-win_amd64.whl.metadata
  Using cached contourpy-1.2.1-cp312-cp312-win_amd64.whl.metadata (5.8 kB)
Collecting cycler>=0.10 (from matplotlib)
  Obtaining dependency information for cycler>=0.10 from https://files.pythonhosted.org/packages/e7/05/c19819d5e3d95294a6f5947fb9b9629efb316b96de511b418c53d245aae6/cycl
er-0.12.1-py3-none-any.whl.metadata
  Using cached cycler-0.12.1-py3-none-any.whl.metadata (3.8 kB)
Collecting fonttools>=4.22.0 (from matplotlib)
  Obtaining dependency information for fonttools>=4.22.0 from https://files.pythonhosted.org/packages/6d/9a/b695930e1b4e6929cc60e294489421632a05c105ac8c56ee63ef56a47872
/fonttools-4.53.1-cp312-cp312-win_amd64.whl.metadata
  Using cached fonttools-4.53.1-cp312-cp312-win_amd64.whl.metadata (165 kB)
Collecting kiwisolver>=1.3.1 (from matplotlib)
  Obtaining dependency information for kiwisolver>=1.3.1 from https://files.pythonhosted.org/packages/63/50/2746566bdf4a6a842d117367d05c90cfb87ac04e9e2845aa1fa21f071362
/kiwisolver-1.4.5-cp312-cp312-win_amd64.whl.metadata
  Using cached kiwisolver-1.4.5-cp312-cp312-win_amd64.whl.metadata (6.5 kB)
Collecting numpy>=1.23 (from matplotlib)
  Obtaining dependency information for numpy>=1.23 from https://files.pythonhosted.org/packages/b5/59/f6ad30785a6578ad85ed9c2785f271b39c3e5b6412c66e810d2c60934c9f/numpy
-2.0.1-cp312-cp312-win_amd64.whl.metadata
  Using cached numpy-2.0.1-cp312-cp312-win_amd64.whl.metadata (60 kB)
Collecting packaging>=20.0 (from matplotlib)
  Obtaining dependency information for packaging>=20.0 from https://files.pythonhosted.org/packages/08/aa/cc0199a5f0ad350994d660967a8efb233fe0416e4639146c089643407ce6/p
ackaging-24.1-py3-none-any.whl.metadata
  Using cached packaging-24.1-py3-none-any.whl.metadata (3.2 kB)
Collecting pillow>=8 (from matplotlib)
  Obtaining dependency information for pillow>=8 from https://files.pythonhosted.org/packages/74/0a/d4ce3c44bca8635bd29a2eab5aa181b654a734a29b263ca8efe013beea98/pillow-
10.4.0-cp312-cp312-win_amd64.whl.metadata
  Using cached pillow-10.4.0-cp312-cp312-win_amd64.whl.metadata (9.3 kB)
Collecting pyparsing>=2.3.1 (from matplotlib)
  Obtaining dependency information for pyparsing>=2.3.1 from https://files.pythonhosted.org/packages/9d/ea/6d76df31432a0e6fdf81681a895f009a4bb47b3c39036db3e1b528191d52/
pyparsing-3.1.2-py3-none-any.whl.metadata
  Using cached pyparsing-3.1.2-py3-none-any.whl.metadata (5.1 kB)
```

ii.plot.py



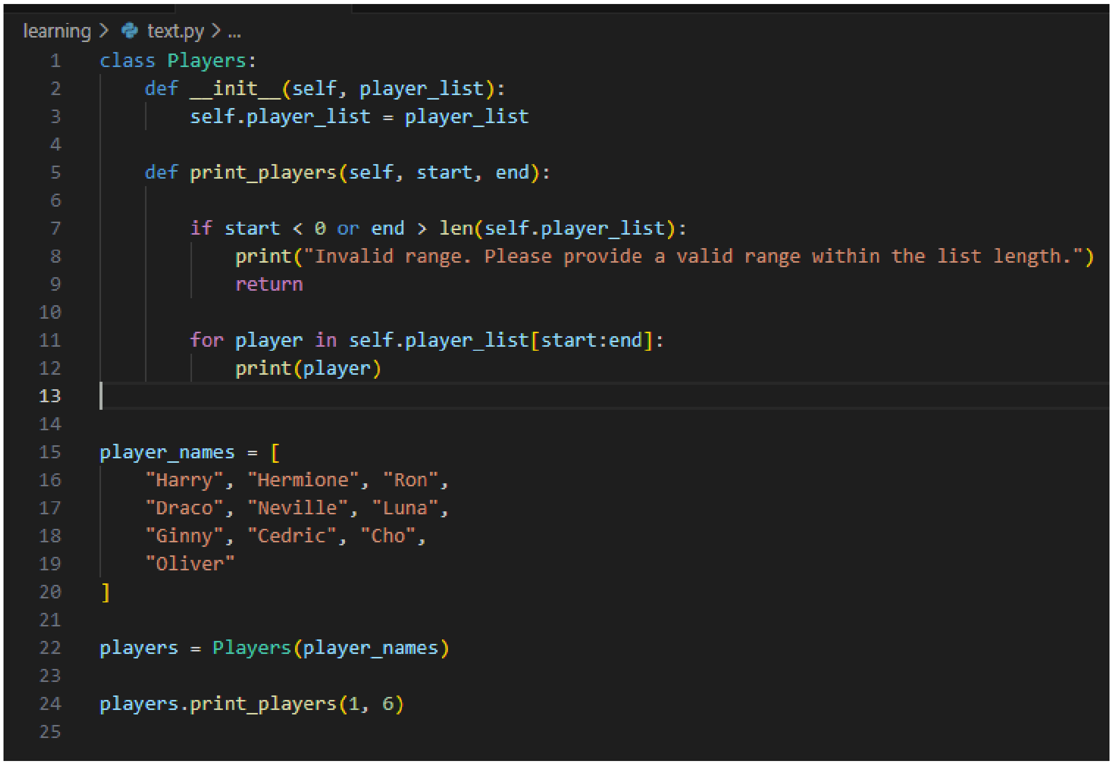
```python
import matplotlib.pyplot as plt

# Example data
x = [1, 2, 3, 4, 5]
y = [10, 20, 25, 30, 40]

# Create a simple line plot
plt.plot(x, y)
plt.title("Simple Line Plot")
plt.xlabel("X-axis")
plt.ylabel("Y-axis")
plt.show()

```

iii. Output



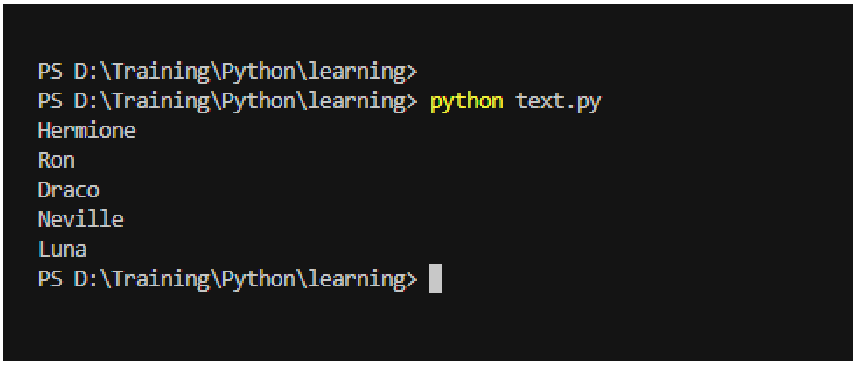
5.Create a range to display players list within Players class.

Program:



```
class Players:
    def __init__(self, player_list):
        self.player_list = player_list

    def print_players(self, start, end):

        if start < 0 or end > len(self.player_list):
            print("Invalid range. Please provide a valid range within the list length.")
            return

        for player in self.player_list[start:end]:
            print(player)


player_names = [
    "Harry", "Hermione", "Ron",
    "Draco", "Neville", "Luna",
    "Ginny", "Cedric", "Cho",
    "Oliver"
]

players = Players(player_names)

players.print_players(1, 6)

```

Output:



```
PS D:\Training\Python\learning>
PS D:\Training\Python\learning> python text.py
Hermione
Ron
Draco
Neville
Luna
PS D:\Training\Python\learning>
```