

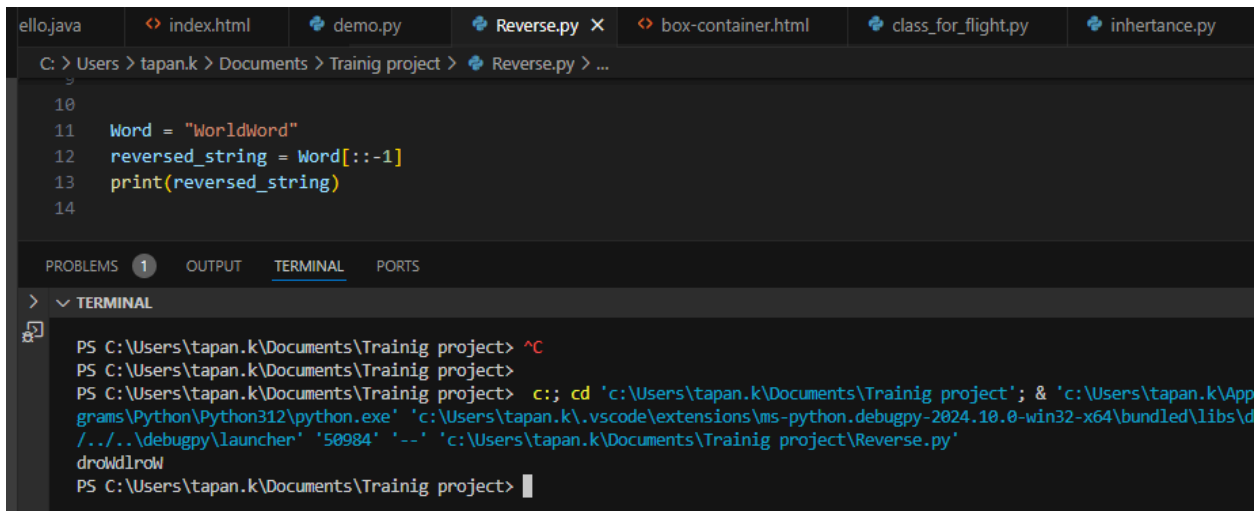
## Python Programs

### 1.Reverse a string “WorldWord”?

#### Code:-

```
Word = "WorldWord"
reversed_string = Word[::-1]
print(reversed_string)
```

#### Output:-

A screenshot of a Visual Studio Code editor window. The top bar shows several open files: 'ello.java', 'index.html', 'demo.py', 'Reverse.py' (which is the active file), 'box-container.html', 'class\_for\_flight.py', and 'inheritance.py'. The editor area shows the following Python code:

```
10
11 Word = "WorldWord"
12 reversed_string = Word[::-1]
13 print(reversed_string)
14
```

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

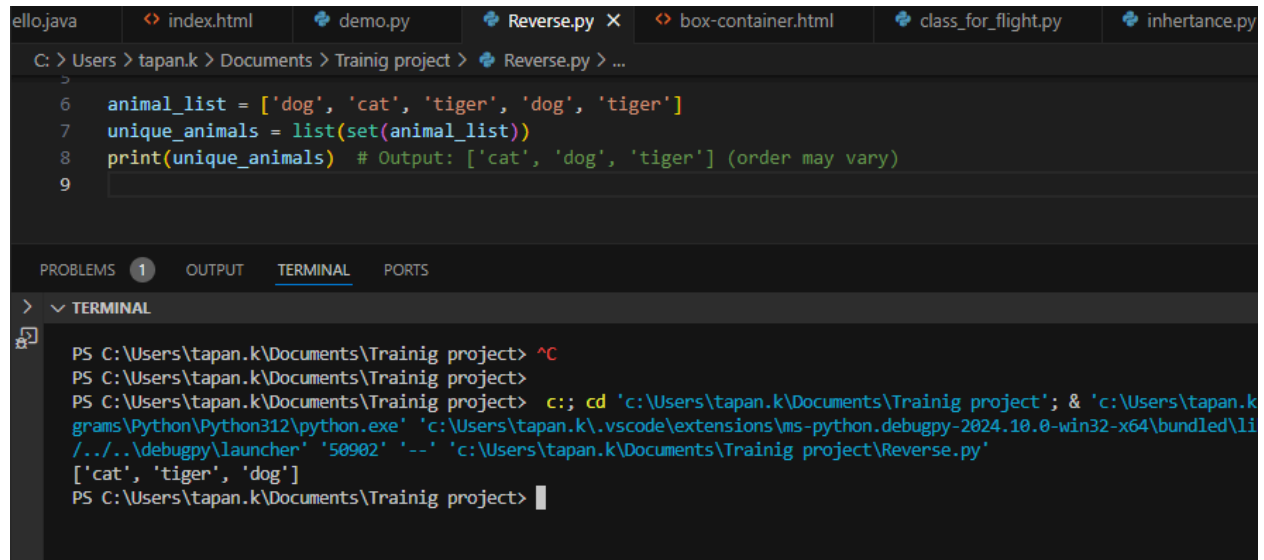
```
PS C:\Users\tapan.k\Documents\Trainig project> ^C
PS C:\Users\tapan.k\Documents\Trainig project>
PS C:\Users\tapan.k\Documents\Trainig project> c:: cd 'c:\Users\tapan.k\Documents\Trainig project'; & 'c:\Users\tapan.k\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\tapan.k\.vscode\extensions\ms-python.debugpy-2024.10.0-win32-x64\bundled\libs\debugpy\launcher' '50984' '--' 'c:\Users\tapan.k\Documents\Trainig project\Reverse.py'
drowdlrow
PS C:\Users\tapan.k\Documents\Trainig project> |
```

### 2.Remove duplicates in ['dog','cat','tiger','dog', 'tiger'] ?

#### Code:-

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

#### Output:-



The screenshot shows a VS Code editor with several tabs open: `ello.java`, `index.html`, `demo.py`, `Reverse.py` (active), `box-container.html`, `class_for_flight.py`, and `inheritance.py`. The `Reverse.py` file contains the following code:

```
6 animal_list = ['dog', 'cat', 'tiger', 'dog', 'tiger']
7 unique_animals = list(set(animal_list))
8 print(unique_animals) # Output: ['cat', 'dog', 'tiger'] (order may vary)
9
```

The terminal at the bottom shows the execution of the script:

```
PS C:\Users\tapan.k\Documents\Trainig project> ^C
PS C:\Users\tapan.k\Documents\Trainig project>
PS C:\Users\tapan.k\Documents\Trainig project> c:: cd 'c:\Users\tapan.k\Documents\Trainig project'; & 'c:\Users\tapan.k\Documents\Python\Python312\python.exe' 'c:\Users\tapan.k\.vscode\extensions\ms-python.debugpy-2024.10.0-win32-x64\bundle\lib\..\..\debugpy\launcher' '50902' '--' 'c:\Users\tapan.k\Documents\Trainig project\Reverse.py'
['cat', 'tiger', 'dog']
PS C:\Users\tapan.k\Documents\Trainig project>
```

### 3. Perform union and intersection using Set ?

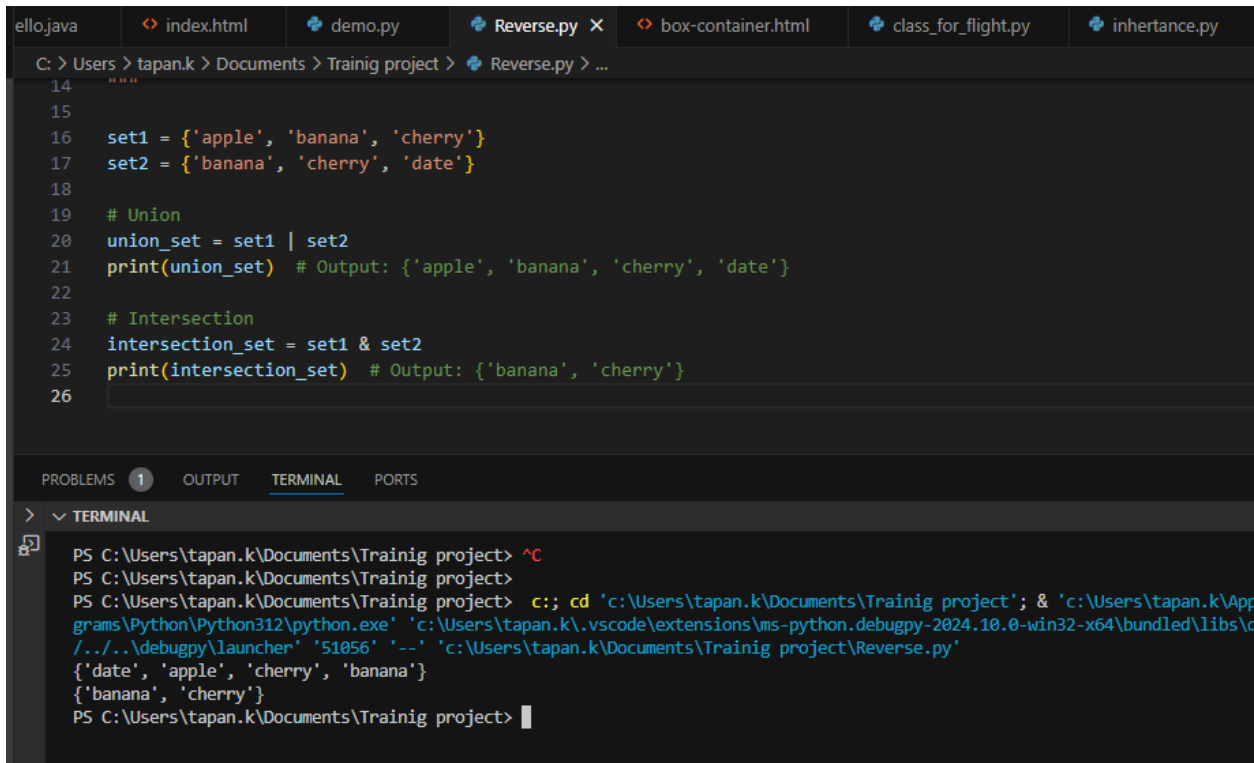
#### Code:-

```
set1 = {'apple', 'banana', 'cherry'}
set2 = {'banana', 'cherry', 'date'}

# Union
union_set = set1 | set2
print(union_set) # Output: {'apple', 'banana', 'cherry', 'date'}

# Intersection
intersection_set = set1 & set2
print(intersection_set)
```

#### Output:-



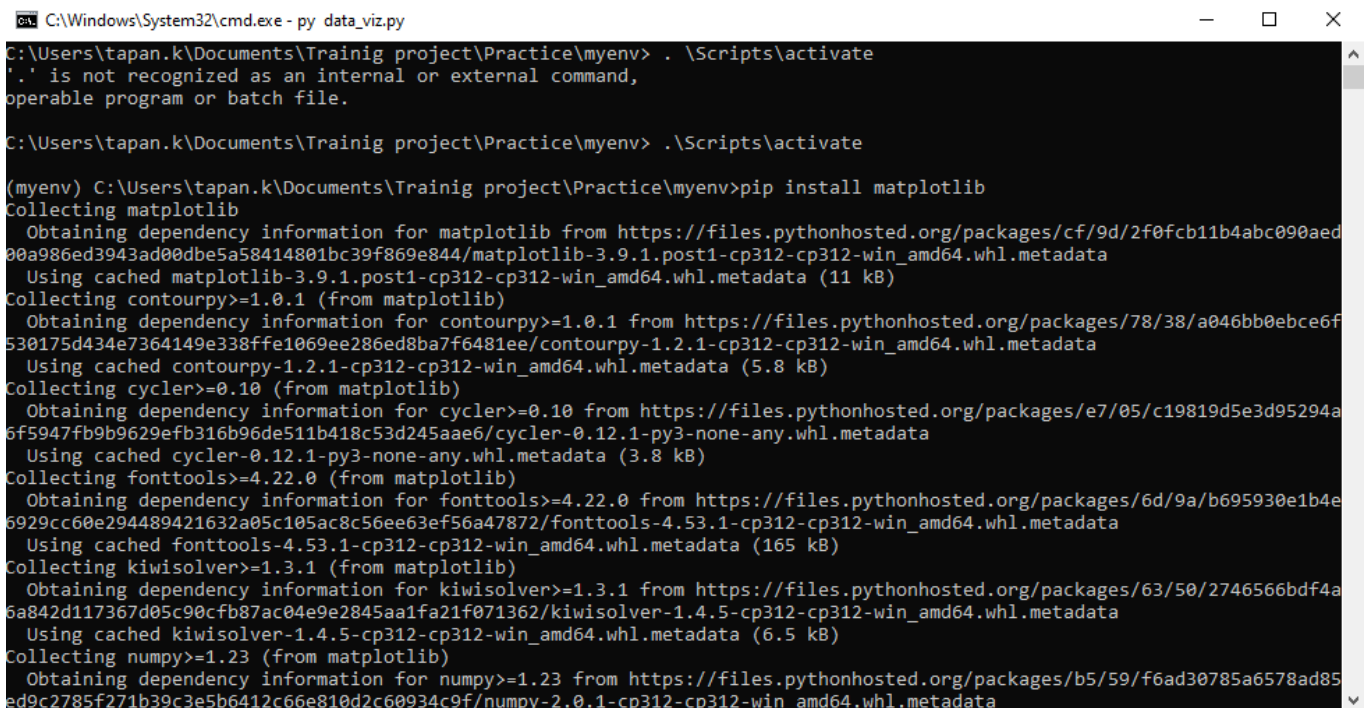
The screenshot shows the VS Code editor with a file named `Reverse.py` open. The file contains Python code that creates two sets, `set1` and `set2`, and calculates their union and intersection. The terminal output shows the execution of the script, displaying the union set as `{'apple', 'banana', 'cherry', 'date'}` and the intersection set as `{'banana', 'cherry'}`.

```
14 """
15
16 set1 = {'apple', 'banana', 'cherry'}
17 set2 = {'banana', 'cherry', 'date'}
18
19 # Union
20 union_set = set1 | set2
21 print(union_set) # Output: {'apple', 'banana', 'cherry', 'date'}
22
23 # Intersection
24 intersection_set = set1 & set2
25 print(intersection_set) # Output: {'banana', 'cherry'}
26
```

Terminal Output:

```
PS C:\Users\tapan.k\Documents\Trainig project> ^C
PS C:\Users\tapan.k\Documents\Trainig project>
PS C:\Users\tapan.k\Documents\Trainig project> c:; cd 'c:\Users\tapan.k\Documents\Trainig project'; & 'c:\Users\tapan.k\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\tapan.k\.vscode\extensions\ms-python.debugpy-2024.10.0-win32-x64\bundled\libs\debugpy\launcher' '51056' '--' 'c:\Users\tapan.k\Documents\Trainig project\Reverse.py'
{'date', 'apple', 'cherry', 'banana'}
{'banana', 'cherry'}
PS C:\Users\tapan.k\Documents\Trainig project>
```

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



The screenshot shows a command prompt window with the following commands and output:

```
C:\Windows\System32\cmd.exe - py data_viz.py
C:\Users\tapan.k\Documents\Trainig project\Practice\myenv> . \Scripts\activate
'.' is not recognized as an internal or external command,
operable program or batch file.
C:\Users\tapan.k\Documents\Trainig project\Practice\myenv> . \Scripts\activate
(myenv) C:\Users\tapan.k\Documents\Trainig project\Practice\myenv> pip install matplotlib
Collecting matplotlib
  Obtaining dependency information for matplotlib from https://files.pythonhosted.org/packages/cf/9d/2f0fcb11b4abc090aed00a986ed3943ad00dbe5a58414801bc39f869e844/matplotlib-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/cycler-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
```

```
C:\Windows\System32\cmd.exe - py data_viz.py

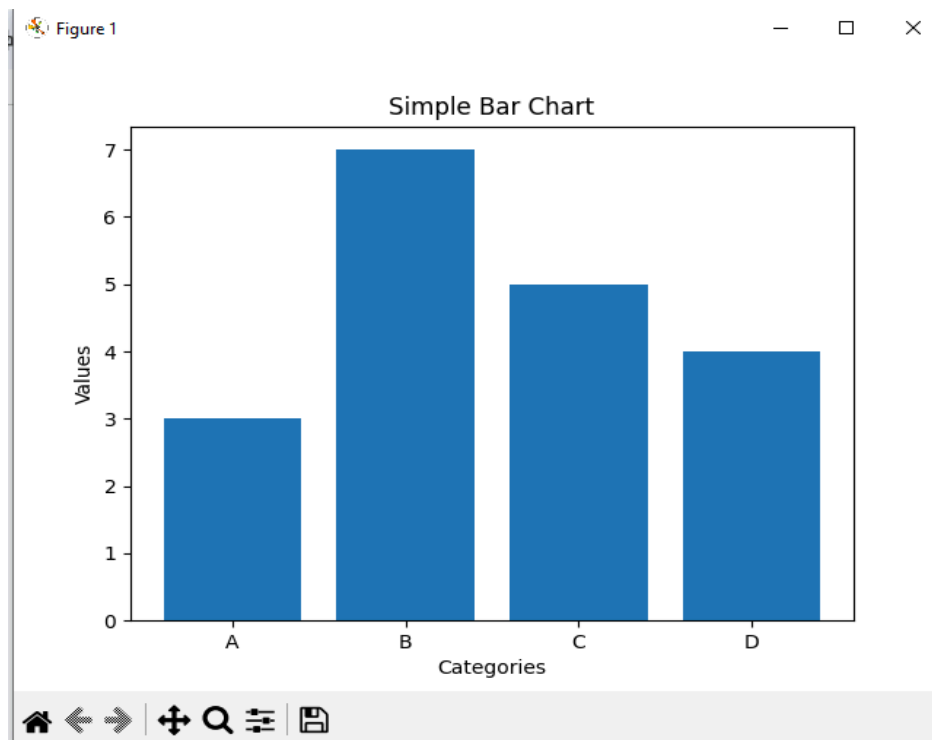
Using cached python_dateutil-2.9.0.post0-py2.py3-none-any.whl.metadata (8.4 kB)
Collecting six>=1.5 (from python-dateutil>=2.7->matplotlib)
Obtaining dependency information for six>=1.5 from https://files.pythonhosted.org/packages/d9/5a/e7c31adbe875f2abbb91b
d84cf2dc52d792b5a01506781dbcf25c91daf11/six-1.16.0-py2.py3-none-any.whl.metadata
Using cached six-1.16.0-py2.py3-none-any.whl.metadata (1.8 kB)
Using cached matplotlib-3.9.1.post1-cp312-cp312-win_amd64.whl (8.0 MB)
Using cached contourpy-1.2.1-cp312-cp312-win_amd64.whl (189 kB)
Using cached cyclor-0.12.1-py3-none-any.whl (8.3 kB)
Using cached fonttools-4.53.1-cp312-cp312-win_amd64.whl (2.2 MB)
Using cached kiwisolver-1.4.5-cp312-cp312-win_amd64.whl (56 kB)
Using cached numpy-2.0.1-cp312-cp312-win_amd64.whl (16.3 MB)
Using cached packaging-24.1-py3-none-any.whl (53 kB)
Using cached pillow-10.4.0-cp312-cp312-win_amd64.whl (2.6 MB)
Using cached pyparsing-3.1.2-py3-none-any.whl (103 kB)
Using cached python_dateutil-2.9.0.post0-py2.py3-none-any.whl (229 kB)
Using cached six-1.16.0-py2.py3-none-any.whl (11 kB)
Installing collected packages: six, pyparsing, pillow, packaging, numpy, kiwisolver, fonttools, cyclor, python-dateutil,
contourpy, matplotlib
Successfully installed contourpy-1.2.1 cyclor-0.12.1 fonttools-4.53.1 kiwisolver-1.4.5 matplotlib-3.9.1.post1 numpy-2.0.
1 packaging-24.1 pillow-10.4.0 pyparsing-3.1.2 python-dateutil-2.9.0.post0 six-1.16.0

[notice] A new release of pip is available: 23.2.1 -> 24.2
[notice] To update, run: python.exe -m pip install --upgrade pip

(myenv) C:\Users\tapan.k\Documents\Trainig project\Practice\myenv>cd
C:\Users\tapan.k\Documents\Trainig project\Practice\myenv

(myenv) C:\Users\tapan.k\Documents\Trainig project\Practice\myenv>py data_viz.py
```

Output:-



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

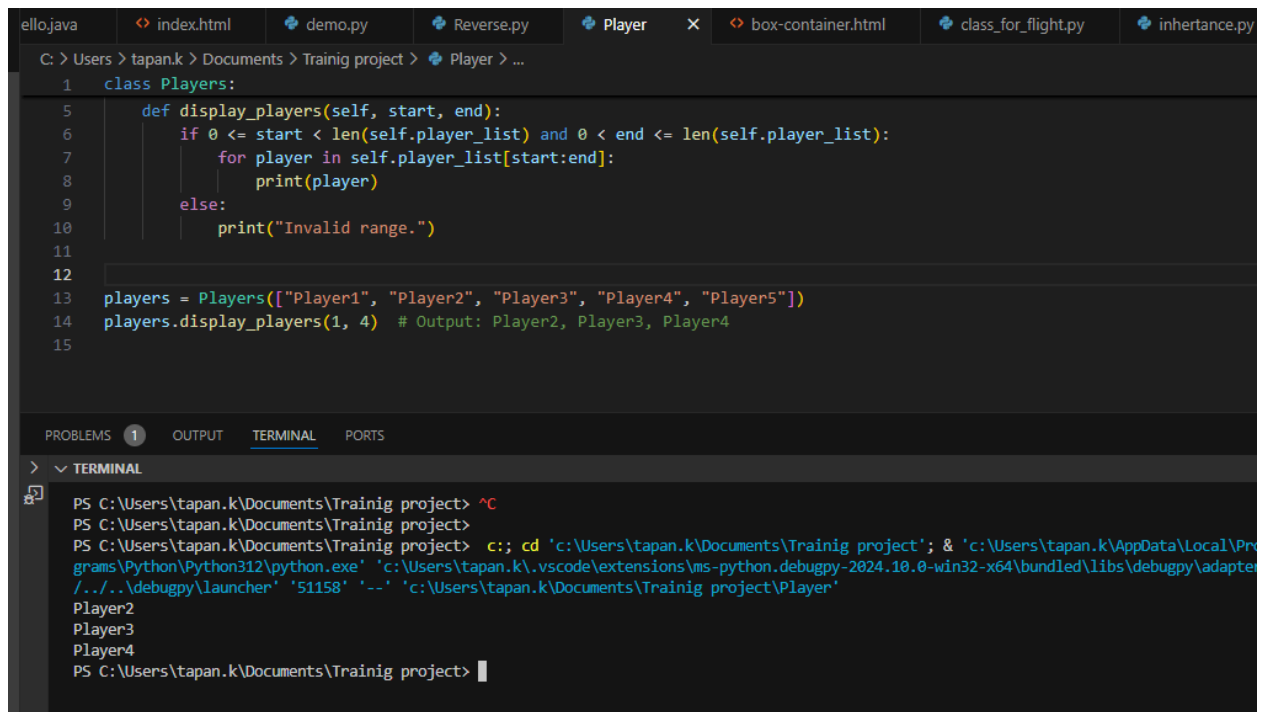
**Code:-**

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

    def display_players(self, start, end):
        if 0 <= start < len(self.player_list) and 0 < end <= len(self.player_list):
            for player in self.player_list[start:end]:
                print(player)
        else:
            print("Invalid range.")

players = Players(["Player1", "Player2", "Player3", "Player4", "Player5"])
players.display_players(1, 4)
```

**Output:-**



The screenshot shows a VS Code editor with a file named 'Player.py' open. The code defines a 'Players' class with an '\_\_init\_\_' method and a 'display\_players' method. The 'display\_players' method checks if the provided start and end indices are within the bounds of the 'player\_list' and then prints the players in that range. Below the code, the terminal output shows the execution of the script, which prints 'Player2', 'Player3', and 'Player4'.

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

    def display_players(self, start, end):
        if 0 <= start < len(self.player_list) and 0 < end <= len(self.player_list):
            for player in self.player_list[start:end]:
                print(player)
        else:
            print("Invalid range.")

players = Players(["Player1", "Player2", "Player3", "Player4", "Player5"])
players.display_players(1, 4)
```

Terminal Output:

```
PS C:\Users\tapan.k\Documents\Trainig project> ^C
PS C:\Users\tapan.k\Documents\Trainig project>
PS C:\Users\tapan.k\Documents\Trainig project> c:: cd 'c:\Users\tapan.k\Documents\Trainig project'; & 'c:\Users\tapan.k\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\tapan.k\.vscode\extensions\ms-python.debugpy-2024.10.0-win32-x64\bundle\libs\debugpy\adapter\..\..\debugpy\launcher' '51158' '--' 'c:\Users\tapan.k\Documents\Trainig project\Player'
Player2
Player3
Player4
PS C:\Users\tapan.k\Documents\Trainig project>
```