

Thumati Pavan Venkata Narendra Kumar

EMP-ID-289219

PYTHON-PROJECT-2

1. Password Generator.

```
passwordGenerator.py X
pavan > Python > passwordGenerator.py
1 import random
2 import string
3 def generate_password(length=12):
4     characters = string.ascii_letters + string.digits + string.punctuation
5     password = ''.join(random.choice(characters) for _ in range(length))
6     return password
7 print("Generated Password:", generate_password(12))
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS powershell - Python
PS C:\Users\Administrator\Desktop\DevOps-Training> cd .\pavan\
PS C:\Users\Administrator\Desktop\DevOps-Training\pavan> cd .\Python\
PS C:\Users\Administrator\Desktop\DevOps-Training\pavan\Python> python .\passwordGenerator.py
Generated Password: gP\p^[T'U3dW
PS C:\Users\Administrator\Desktop\DevOps-Training\pavan\Python> python .\passwordGenerator.py
Generated Password: HK"ornQ9rSFQ
PS C:\Users\Administrator\Desktop\DevOps-Training\pavan\Python> python .\passwordGenerator.py
Generated Password: bP0]<i&QuLK@
PS C:\Users\Administrator\Desktop\DevOps-Training\pavan\Python>
PS C:\Users\Administrator\Desktop\DevOps-Training\pavan\Python> python .\passwordGenerator.py
Generated Password: :Oxf{&ht*D5<
PS C:\Users\Administrator\Desktop\DevOps-Training\pavan\Python> python .\passwordGenerator.py
Generated Password: lz78m!)P7pf*
PS C:\Users\Administrator\Desktop\DevOps-Training\pavan\Python> █
```

2. To-Do List (CLI).

```
toDoList.py ×
pavan > Python > toDoList.py
1  l=[]
2  flag = True
3  while flag:
4
5      print("\n1. To Add Task\n2. To View Tasks\n3. To Remove Task\n4. To Exit")
6      print("Please select any one option from the above list :: ")
7      ch = int(input())
8      match ch:
9          case 1:
10         newTask = input("Enter task to add : ")
11         l.append(newTask)
12         print("Task added!")
13         case 2:
14             i=1
15             print("\nThe To Do list is : ")
16             for j in l:
17                 print(f"{i}-->{j}")
18                 i=i+1
19             print("\n")
20         case 3:
21             newTask = input("Enter task to view : ")
22             l.remove(newTask)
23             print("Task removed!")
24         case 4:
25             flag=False
26             break
27         case default:
28             print("Choice valid option")
29
```

```
~~~~~
PS C:\Users\Administrator\Desktop\DevOps-Training\pavan\Python> python .\toDoList.py

1. To Add Task
2. To View Tasks
3. To Remove Task
4. To Exit
Please select any one option from the above list ::
1
Enter task to add : Running
Task added!

1. To Add Task
2. To View Tasks
3. To Remove Task
4. To Exit
Please select any one option from the above list ::
1
Enter task to add : GYM
Task added!

1. To Add Task
2. To View Tasks
3. To Remove Task
4. To Exit
Please select any one option from the above list ::
2
```

```
Please select any one option from the above list ::  
2
```

```
The To Do list is :  
1-->Running  
2-->GYM
```

1. To Add Task
2. To View Tasks
3. To Remove Task
4. To Exit

```
Please select any one option from the above list ::  
3
```

```
Enter task to view : Running  
Task removed!
```

1. To Add Task
2. To View Tasks
3. To Remove Task
4. To Exit

```
Please select any one option from the above list ::  
2
```

```
The To Do list is :  
1-->GYM
```

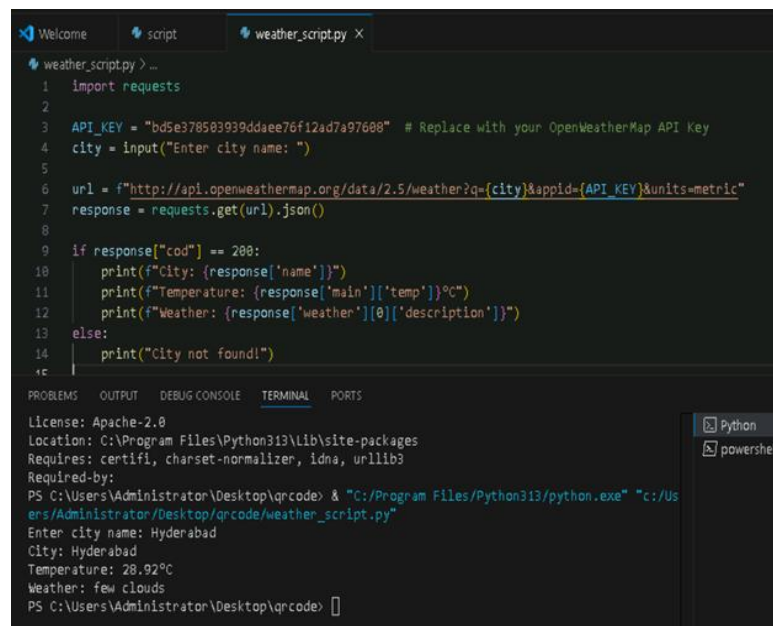
```
The To Do list is :  
1-->GYM
```

1. To Add Task
2. To View Tasks
3. To Remove Task
4. To Exit

```
Please select any one option from the above list ::  
4
```

```
> PS C:\Users\Administrator\Desktop\DevOps-Training\pavan\Python> █
```

3. Weather App (API-based).

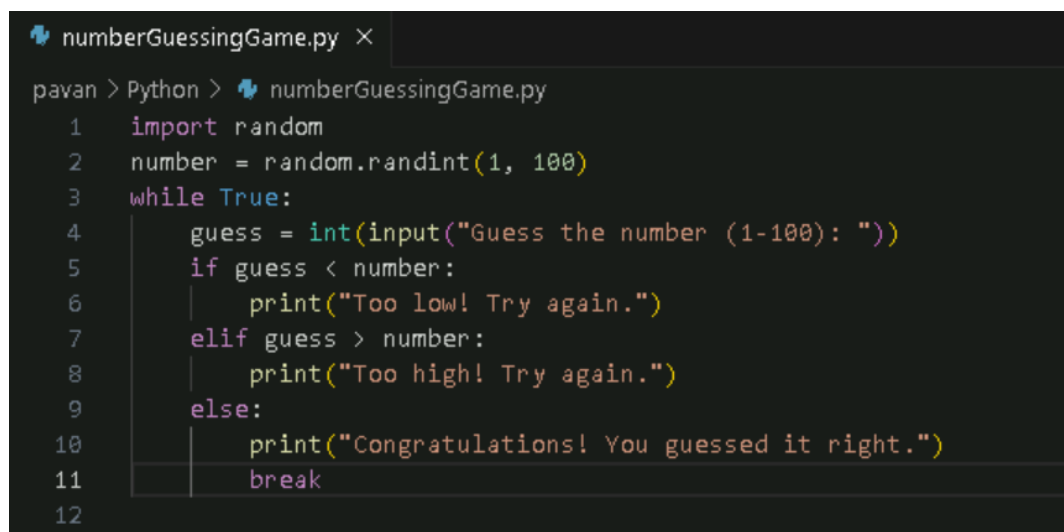


The screenshot shows a VS Code editor with a file named `weather_script.py`. The script uses the `requests` library to fetch weather data from the OpenWeatherMap API. The terminal output shows the script being executed, with the user entering 'Hyderabad' as the city name. The output displays the city name, temperature (28.92°C), and weather (few clouds).

```
1 import requests
2
3 API_KEY = "bd5e378583939ddae76f12ad7a97688" # Replace with your OpenWeatherMap API Key
4 city = input("Enter city name: ")
5
6 url = f"http://api.openweathermap.org/data/2.5/weather?q={city}&appid={API_KEY}&units=metric"
7 response = requests.get(url).json()
8
9 if response["cod"] == 200:
10     print(f"City: {response['name']}")
11     print(f"Temperature: {response['main']['temp']}°C")
12     print(f"Weather: {response['weather'][0]['description']}")
13 else:
14     print("City not found!")
```

License: Apache-2.0
Location: C:\Program Files\Python313\Lib\site-packages
Requires: certifi, charset-normalizer, idna, urllib3
Required-by:
PS C:\Users\Administrator\Desktop\qrcode> & "C:/Program Files/Python313/python.exe" "c:/Users/Administrator/Desktop/qrcode/weather_script.py"
Enter city name: Hyderabad
City: Hyderabad
Temperature: 28.92°C
Weather: few clouds
PS C:\Users\Administrator\Desktop\qrcode>

4. Number Guessing Game.



The screenshot shows a VS Code editor with a file named `numberGuessingGame.py`. The script uses the `random` library to generate a random number between 1 and 100. It then enters a `while` loop where the user is prompted to guess the number. If the guess is too low or too high, the user is prompted to try again. If the guess is correct, the user is congratulated and the loop is broken.

```
1 import random
2 number = random.randint(1, 100)
3 while True:
4     guess = int(input("Guess the number (1-100): "))
5     if guess < number:
6         print("Too low! Try again.")
7     elif guess > number:
8         print("Too high! Try again.")
9     else:
10        print("Congratulations! You guessed it right.")
11        break
12
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS powershell - Python

PS C:\Users\Administrator\Desktop\DevOps-Training\pavan\Python> python .\numberGuessingGame.py
Guess the number (1-100): 18
Too low! Try again.
Guess the number (1-100): 56
Too low! Try again.
Guess the number (1-100): 89
Too low! Try again.
Guess the number (1-100): 90
Too low! Try again.
Guess the number (1-100): 98
Too high! Try again.
Guess the number (1-100): 92
Congratulations! You guessed it right.
PS C:\Users\Administrator\Desktop\DevOps-Training\pavan\Python> 
```

5. QR Code Generator.

```
QRCodeGenerator.py X
pavan > Python > QRCodeGenerator.py
1 import qrcode
2 data = input("Enter text or URL: ")
3 qr = qrcode.make(data)
4 qr.save("qrcode.png")
5 print("QR Code generated and saved as 'qrcode.png'!")

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS powershell - Pyt

PS C:\Users\Administrator\Desktop\DevOps-Training\pavan\Python> python .\QRCodeGenerator.py
Enter text or URL: Hi welcome..
QR Code generated and saved as 'qrcode.png'!
PS C:\Users\Administrator\Desktop\DevOps-Training\pavan\Python> 
```

