

# PYTHON CASE STUDY

## Password Generator

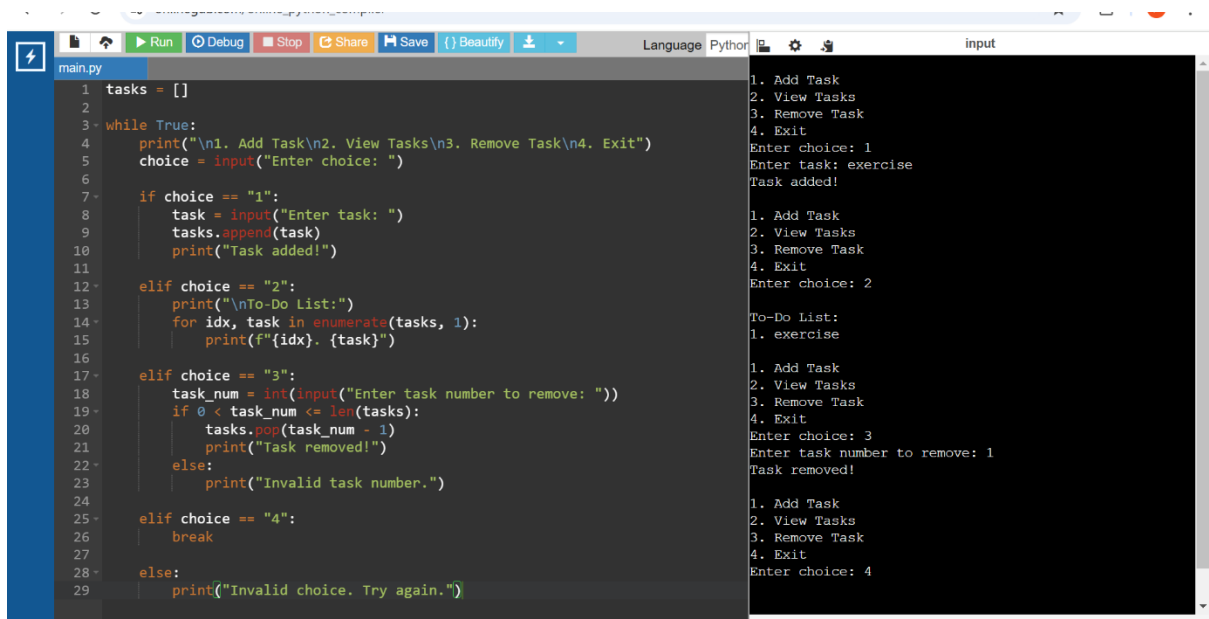


The screenshot shows a Python IDE with a file named `main.py`. The code defines a function `generate_password` that takes a length parameter (default 12) and returns a random password. The password is generated by joining random choices from a set of characters (ASCII letters, digits, and punctuation). The script prints the generated password and then exits with code 0.

```
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))
8
```

The output in the console shows the generated password: `Generated Password: Fq\,qu!7TfH1`. The program then finishes with the message: `...Program finished with exit code 0` and prompts the user to press ENTER to exit the console.

## To-Do List (CLI)

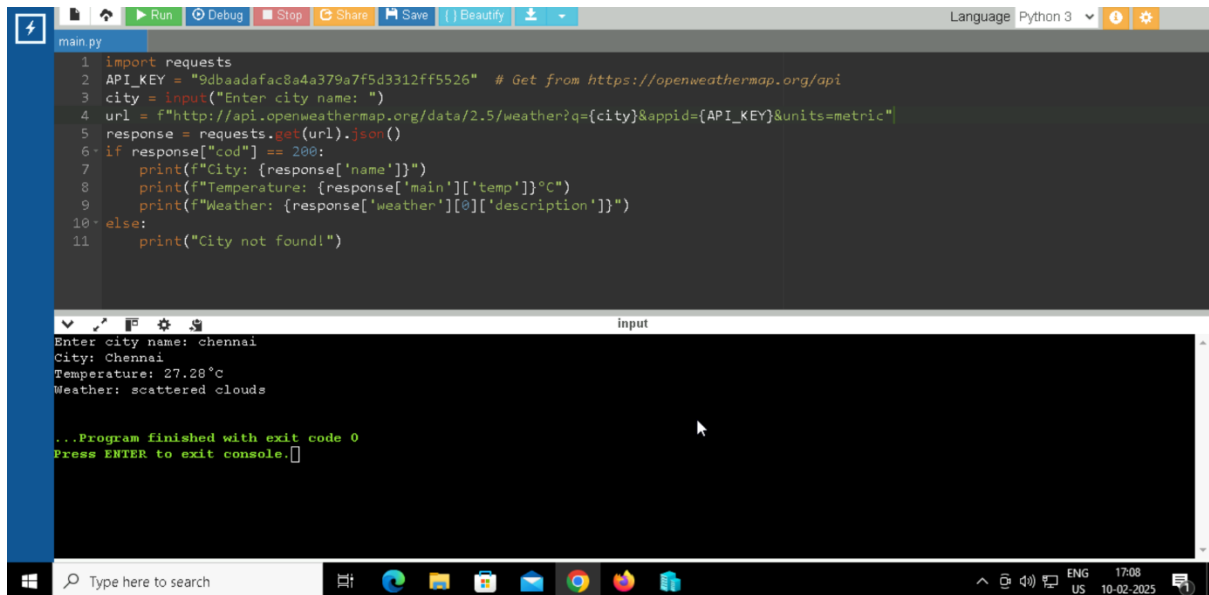


The screenshot shows a Python IDE with a file named `main.py`. The code implements a To-Do List CLI application. It starts with an empty list of tasks. The user can add tasks, view the list, remove tasks, or exit. The application uses a loop to handle user input and a menu to guide the user.

```
1 tasks = []
2
3 while True:
4     print("\n1. Add Task\n2. View Tasks\n3. Remove Task\n4. Exit")
5     choice = input("Enter choice: ")
6
7     if choice == "1":
8         task = input("Enter task: ")
9         tasks.append(task)
10        print("Task added!")
11
12    elif choice == "2":
13        print("\nTo-Do List:")
14        for idx, task in enumerate(tasks, 1):
15            print(f"{idx}. {task}")
16
17    elif choice == "3":
18        task_num = int(input("Enter task number to remove: "))
19        if 0 < task_num <= len(tasks):
20            tasks.pop(task_num - 1)
21            print("Task removed!")
22        else:
23            print("Invalid task number.")
24
25    elif choice == "4":
26        break
27
28    else:
29        print("Invalid choice. Try again.")
```

The output in the console shows the menu and user interactions. The user enters '1' to add a task, 'exercise', and the task is added. The user then enters '2' to view the list, and the output shows the To-Do List with the task 'exercise'. The user then enters '3' to remove a task, and the output shows the task removed. Finally, the user enters '4' to exit the application.

## Weather App (API-based)



The screenshot shows a Python script named `main.py` running in a terminal window. The script uses the `requests` library to fetch weather data from the OpenWeatherMap API. The user has entered `chennai` as the city name. The output shows the city name, temperature, and weather description.

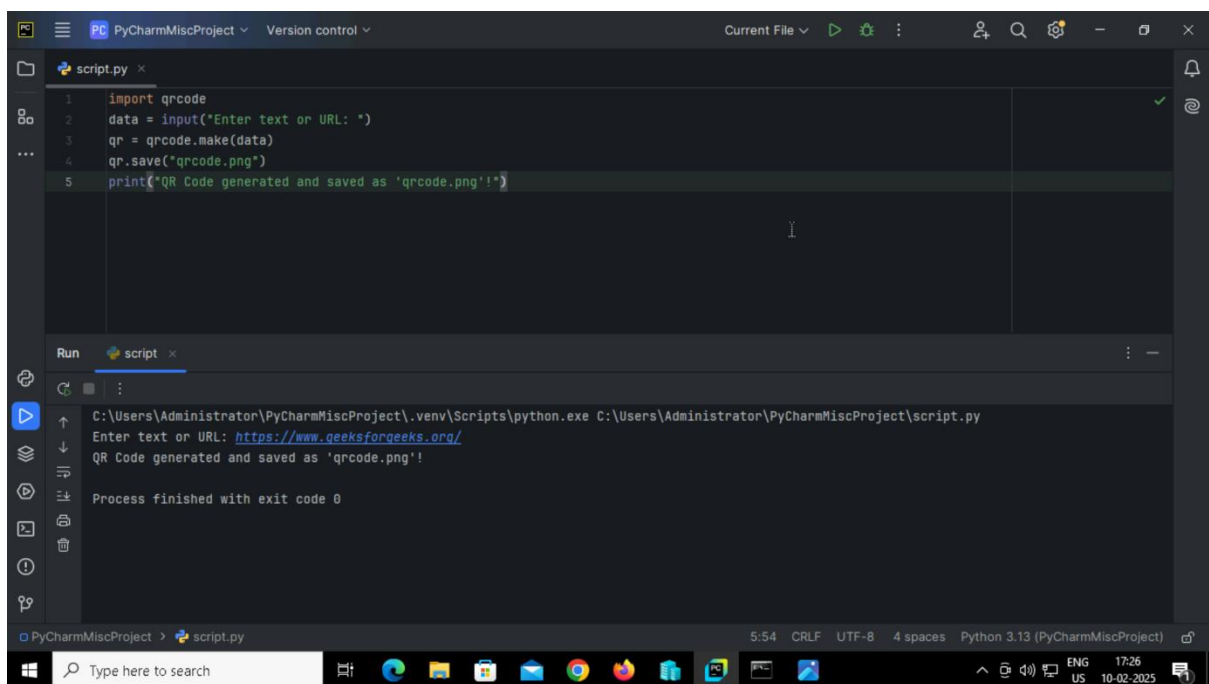
```
1 import requests
2 API_KEY = "9dbaadafac8a4a379a7f5d3312ff5526" # Get from https://openweathermap.org/api
3 city = input("Enter city name: ")
4 url = f"http://api.openweathermap.org/data/2.5/weather?q={city}&appid={API_KEY}&units=metric"
5 response = requests.get(url).json()
6 if response["cod"] == 200:
7     print(f"City: {response['name']}")
8     print(f"Temperature: {response['main']['temp']}°C")
9     print(f"Weather: {response['weather'][0]['description']}")
10 else:
11     print("City not found!")
```

input

Enter city name: chennai  
City: Chennai  
Temperature: 27.28°C  
Weather: scattered clouds

...Program finished with exit code 0  
Press ENTER to exit console.

## QR Code Generator



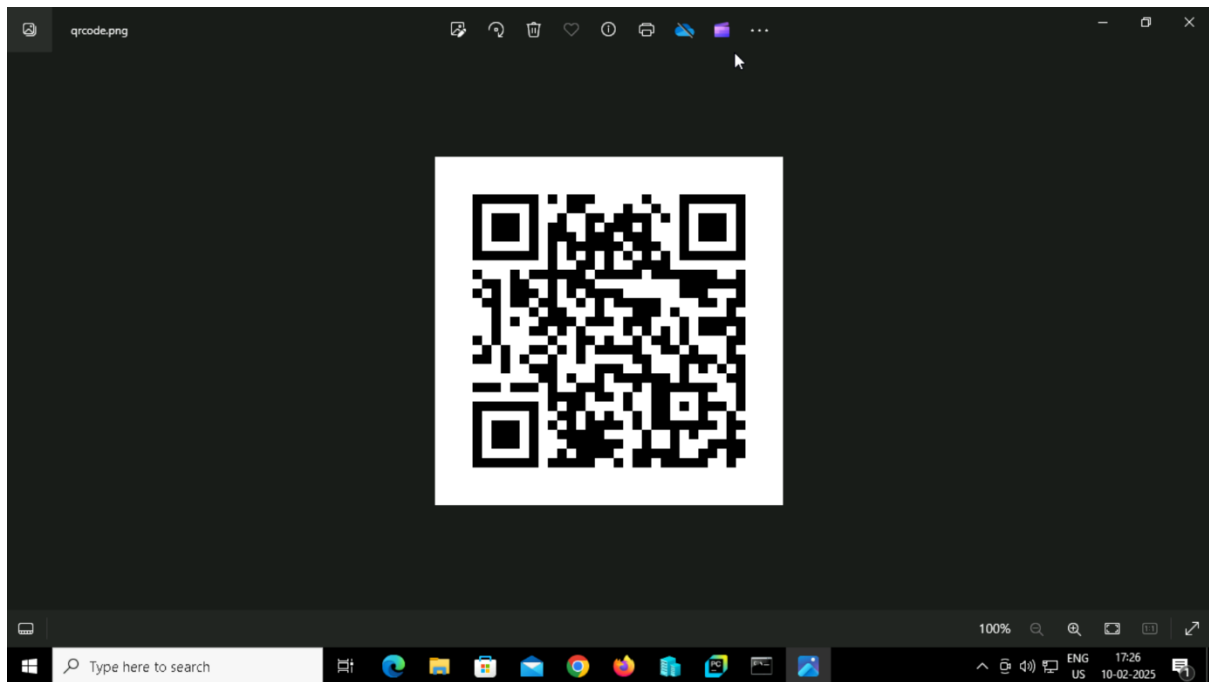
The screenshot shows a Python script named `script.py` running in PyCharm. The script uses the `qrcode` library to generate a QR code from a user-provided text or URL. The user has entered `https://www.geeksforgeeks.org/` as the text or URL. The output shows the QR code generated and saved as `qrcode.png`.

```
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'!")
```

Run script

C:\Users\Administrator\PyCharmMiscProject\.venv\Scripts\python.exe C:\Users\Administrator\PyCharmMiscProject\script.py  
Enter text or URL: <https://www.geeksforgeeks.org/>  
QR Code generated and saved as 'qrcode.png'!

Process finished with exit code 0



## Number Guessing Game

```
main.py
1 import random
2
3 number = random.randint(1, 100)
4
5 while True:
6     guess = int(input("Guess the number (1-100): "))
7
8     if guess < number:
9         print("Too low! Try again.")
10    elif guess > number:
11        print("Too high! Try again.")
12    else:
13        print("Congratulations! You guessed it right.")
14        break
15
```

```
input
Guess the number (1-100): 78
Too high! Try again.
Guess the number (1-100): 36
Too low! Try again.
Guess the number (1-100): 55
Too high! Try again.
Guess the number (1-100): 45
Too low! Try again.
Guess the number (1-100): 50
Too low! Try again.
Guess the number (1-100): 51
Too low! Try again.
Guess the number (1-100): 52
Congratulations! You guessed it right.

...Program finished with exit code 0
Press ENTER to exit console.
```