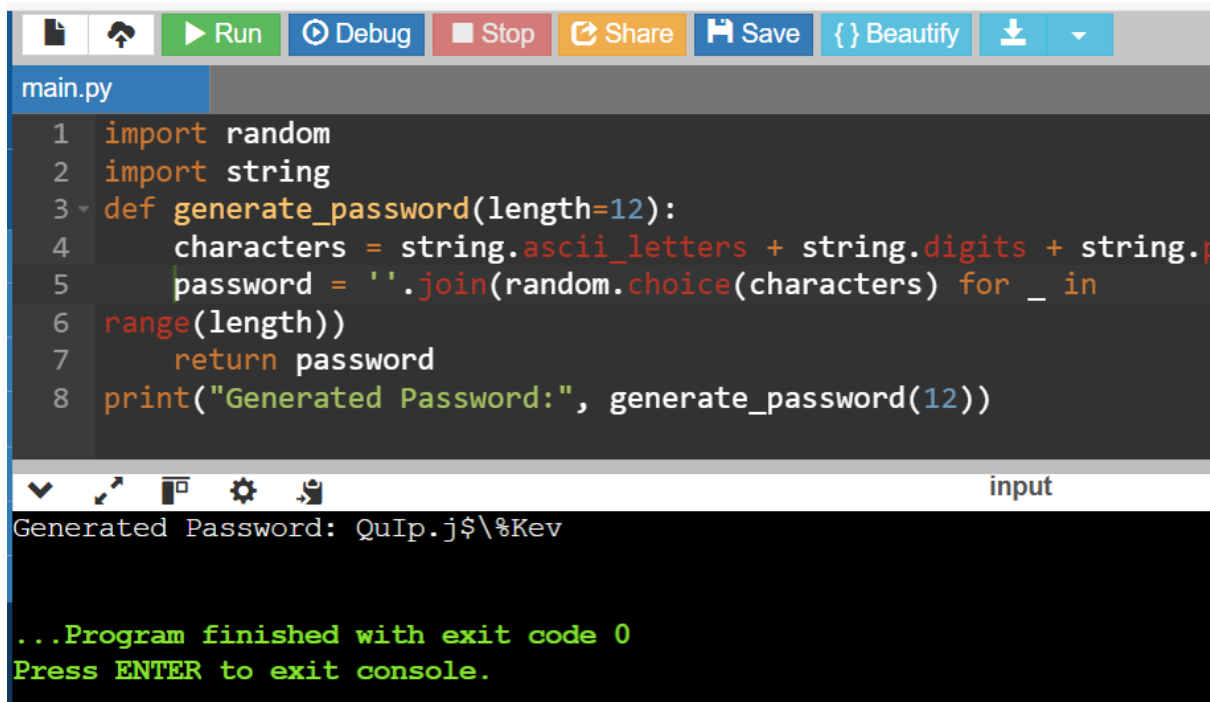


PYTHON BASIC PROJECT

1. PASSWORD GENERATOR



The screenshot shows a Python IDE with a toolbar at the top containing icons for file operations, a 'Run' button, 'Debug', 'Stop', 'Share', 'Save', 'Beautify', and a download icon. The file 'main.py' is open, displaying the following code:

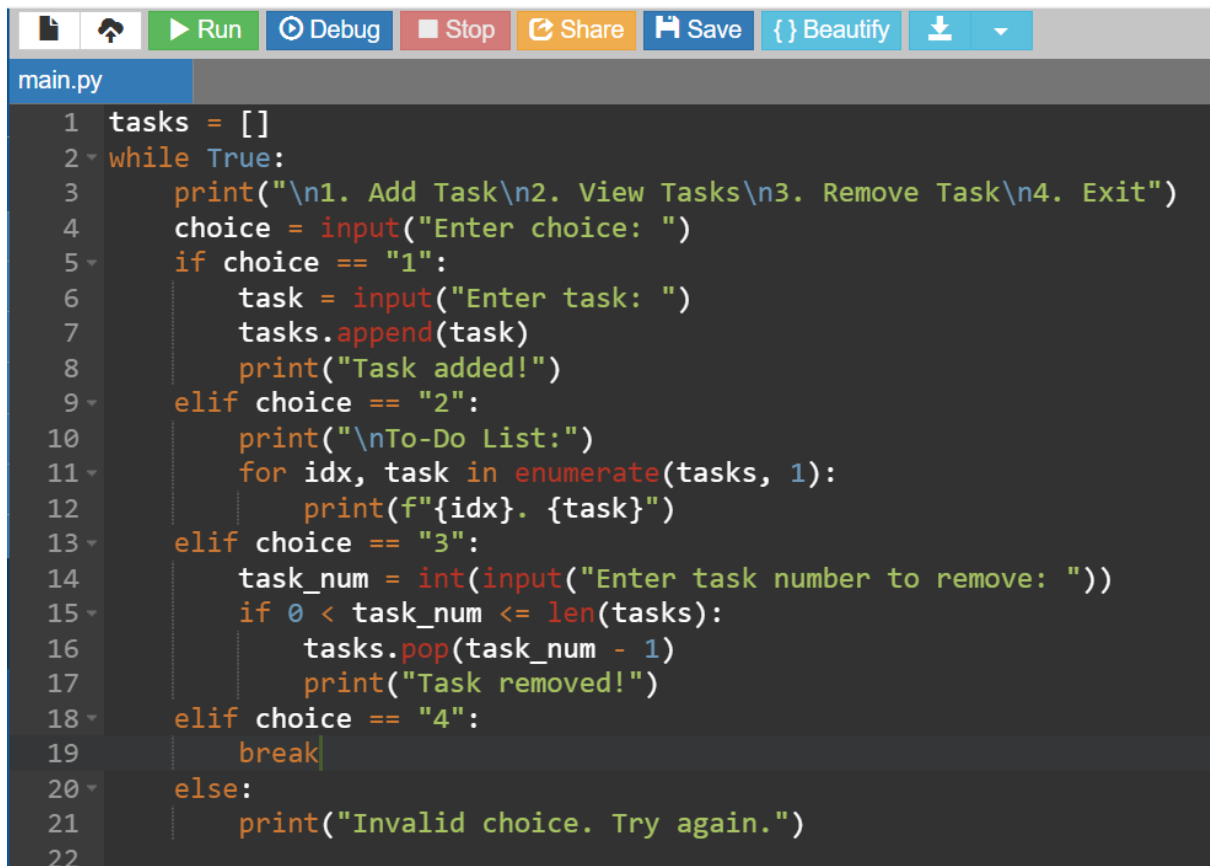
```
1 import random
2 import string
3 def generate_password(length=12):
4     characters = string.ascii_letters + string.digits + string.punctuation
5     password = ''.join(random.choice(characters) for _ in
6 range(length))
7     return password
8 print("Generated Password:", generate_password(12))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

Below the code editor, the console output is visible, showing the generated password and the program's completion message:

```
input
Generated Password: QuIp.j$%\%Kev

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

2. TO-DO LIST



The screenshot shows a Python IDE with a toolbar at the top containing icons for file operations, a 'Run' button, 'Debug', 'Stop', 'Share', 'Save', 'Beautify', and a download icon. The file 'main.py' is open, displaying the following code:

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

OUTPUT

```
input
1. Add Task
2. View Tasks
3. Remove Task
4. Exit
Enter choice: 1
Enter task: work
Task added!

1. Add Task
2. View Tasks
3. Remove Task
4. Exit
Enter choice: 1
Enter task: read
Task added!

1. Add Task
2. View Tasks
3. Remove Task
4. Exit
Enter choice: 1
Enter task: sleep
Task added!

1. Add Task
2. View Tasks
3. Remove Task
4. Exit
Enter choice: 2

To-Do List:
1. work
2. read
3. sleep

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

3. WEATHER APP (API Based)

```
main.py
1 import requests
2 API_KEY = "8f2d6822fb2e4524adf20f8132e6f463"
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"\nCity: {response['name']}")
8     print(f"Temperature: {response['main']['temp']}°C")
9     print(f"Weather: {response['weather'][0]['description']}")
10 else:
11     print("\nCity not found!")

input
Enter city name: hyderabad

City: Hyderabad
Temperature: 28.92°C
Weather: few clouds
```

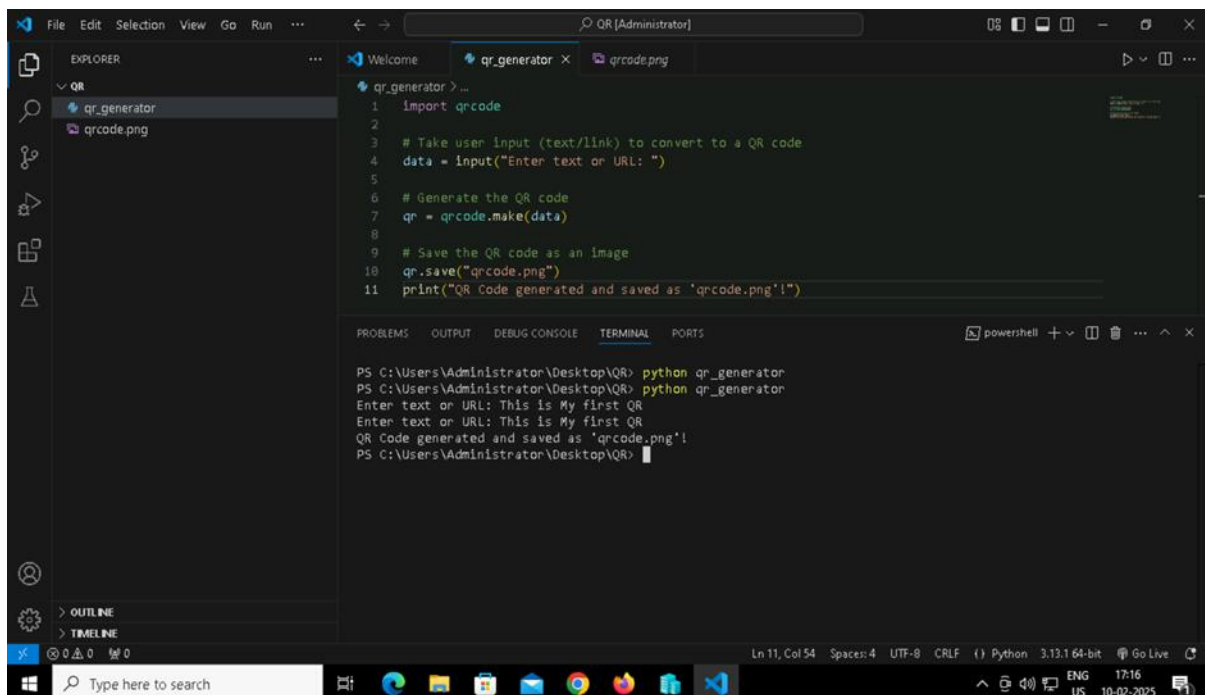
4. NUMBER GUESSING GAME

```
main.py
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
```

input

Guess the number (1-100): 22
Too low! Try again.
Guess the number (1-100): 6
Too low! Try again.
Guess the number (1-100): 15
Too low! Try again.
Guess the number (1-100): 25
Too low! Try again.
Guess the number (1-100): 35
Congratulations! You guessed it right.

5. QR CODE GENERATOR



```
File Edit Selection View Go Run ... QR [Administrator]
EXPLORER
  QR
    qr_generator
    qrcode.png
  qr_generator > ...
1 import qrcode
2
3 # Take user input (text/link) to convert to a QR code
4 data = input("Enter text or URL: ")
5
6 # Generate the QR code
7 qr = qrcode.make(data)
8
9 # Save the QR code as an image
10 qr.save("qrcode.png")
11 print("QR Code generated and saved as 'qrcode.png'!")
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\Users\Administrator\Desktop\QR> python qr_generator
PS C:\Users\Administrator\Desktop\QR> python qr_generator
Enter text or URL: This is My first QR
Enter text or URL: This is My first QR
QR Code generated and saved as 'qrcode.png'!
PS C:\Users\Administrator\Desktop\QR>

Ln 11, Col 54 Spaces: 4 UTF-8 CRLF Python 3.13.1 64-bit Go Live

OUTPUT

