Problem No: 04
Problem Name: Implement and Establish "HTTP Server Responds with a Webpage in Python"

Objective:

To understand how HTTP communication works and to implement a basic HTTP server capable of sending an HTML response to a web browser.

Theory:

The HyperText Transfer Protocol (HTTP) is the foundation of data communication on the web. A client (usually a browser) sends an HTTP request, and the server responds with headers and webpage content. A minimal HTTP server listens on a port, accepts connections, parses requests, and returns HTML data formatted as an HTTP response.

Key Components:

- HTTP request and response structure
- Status line and response headers
- HTML content delivery
- Server socket listening and accepting connections
- Browser acting as the client

Application:

HTTP servers are used in all web-based applications, serving webpages, APIs, images, and dynamic data. This lab demonstrates the fundamentals behind web servers like Apache, Nginx, Flask, and Django.

Implementation in Python:

Server Code (server1002.py)

```
from http.server import HTTPServer, BaseHTTPRequestHandler
HOST="127.0.0.1"
PORT=1002
class MyHandler(BaseHTTPRequestHandler):
  def do_GET(self):
    self.send_response(200)
    self.send_header("Content-type","text/html")
    self.end_headers()
                html="""<!DOCTYPE html><html><head><title>Monowar HTTP Server</title><style>body{font-family:Arial;background:#f0f0f0;text-align:center;}h1{color:#333;}</style></head><body><h1>Welcome to Monowar Islam https server</h1><p>This page is mine HTTP server implemented in Python.</p></body></html>"""
    self.wfile.write(html.encode("utf-8"))
if __name__=="__main__":
```

```
print(f"Starting HTTP Server at http://{HOST}:{PORT}")
server=HTTPServer((HOST,PORT),MyHandler)
server.serve_forever()
```

Result:
A functioning HTTP server that responds to browser requests with a custom webpage, confirming successful communication between client and server.

Sample Output:

Server Console:
Starting HTTP Server at http://127.0.0.1:1002
127.0.0.1 - - [03/Dec/2025 00:47:16] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [03/Dec/2025 00:47:17] "GET /favicon.ico HTTP/1.1" 200 -

Client Browser:

Discussion :

This lab helps understand the foundation of web communication and the role of HTTP protocol in client–server systems. It also shows how raw sockets can be used to build the simplest form of a web server.

Conclusion :
The HTTP server successfully handles client requests and displays a webpage. This experiment provides foundational knowledge for developing advanced web applications and backend frameworks.