Problem No: 05

Problem Name: Implement and Establish "HTTP Server Responds with a Webpage in Python"

Objective

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

Theory

The HyperText Transfer Protocol (HTTP) enables communication between a client and a server. A browser sends an HTTP request, and the server responds with:

- Status line (200 OK)
- Response headers
- HTML body content

A minimal server listens on a port, accepts connections, handles GET requests, and returns formatted HTML data.

Key Components

- HTTP request and response structure
- Status codes and headers
- HTML content delivery
- Server socket listening
- Browser acting as client

Application

HTTP servers are used in web applications, APIs, and backend frameworks such as Flask and Django. This experiment demonstrates core web server functionality.

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()
```

Execution Steps

1. Save as `server1002.py`

2. Run: `python server1002.py`
3. Visit: `http://127.0.0.1:1002`

Result

The HTTP server successfully responds to browser requests with a custom webpage, confirming proper client–server communication.

Sample Output

Server Console:

```
Starting HTTP Server at http://127.0.0.1:1002
127.0.0.1 - - "GET / HTTP/1.1" 200 -
127.0.0.1 - - "GET /favicon.ico HTTP/1.1" 200 -
```

Client Browser:

Welcome to Monowar Islam https server

This page is mine HTTP server implemented in Python.

Discussion

This experiment demonstrates the HTTP request–response lifecycle and the implementation of a simple web server using Python's built-in module.

Conclusion

The server handles client requests and delivers HTML content successfully, providing foundational understanding of HTTP-based client–server systems.