Computer Networks

Programming Assignment

Team Members

Alaa Mohamed Abdel Hamid	6473
Nouran Hisham	6532
Kareem Sabra	6491

Multi-threaded web server:

In this part we used multi-threading and socket programming concepts. Our server is connected on port 8085 and gives out an error if this port is not free to connect to.

The server makes TCP connections with its clients and responds to requests of type GET or POST only with either 200 OK or 404 not found.

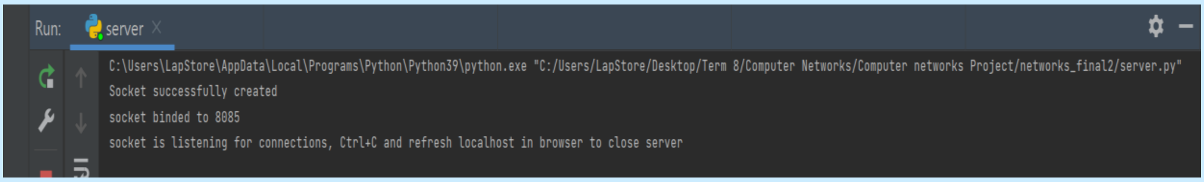
Our server responds to different HTTP versions, HTTP 1.0 and HTTP 1.1. The main differences between the two versions are persistent connections and pipelining which are present in HTTP 1.1 and not present in HTTP 1.0.

In HTTP 1.0, we immediately close the client socket after sending the response to the client. However, in Http 1.1 we keep the socket opened for 10 seconds ready to receive any further requests from the same client without having to connect to our server again.

All our server files in a folder called template which the server searches in for any files requested by the client in GET requests.

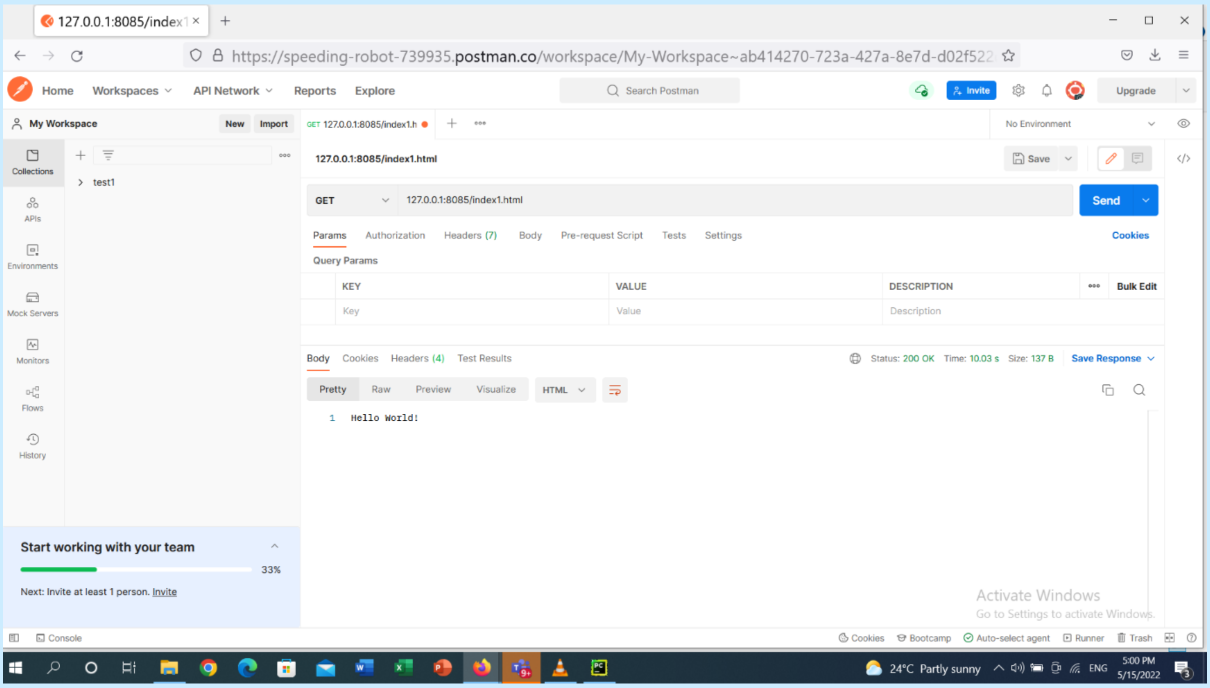
Server sample runs:

This is when we first run our server:



```
C:\Users\LapStore\AppData\Local\Programs\Python\Python39\python.exe "C:/Users/LapStore/Desktop/Term 8/Computer Networks/Computer networks Project/networks_final2/server.py"
Socket successfully created
socket binded to 8085
socket is listening for connections, Ctrl+C and refresh localhost in browser to close server
```

This is testing our server with postman:
GET request (html file):

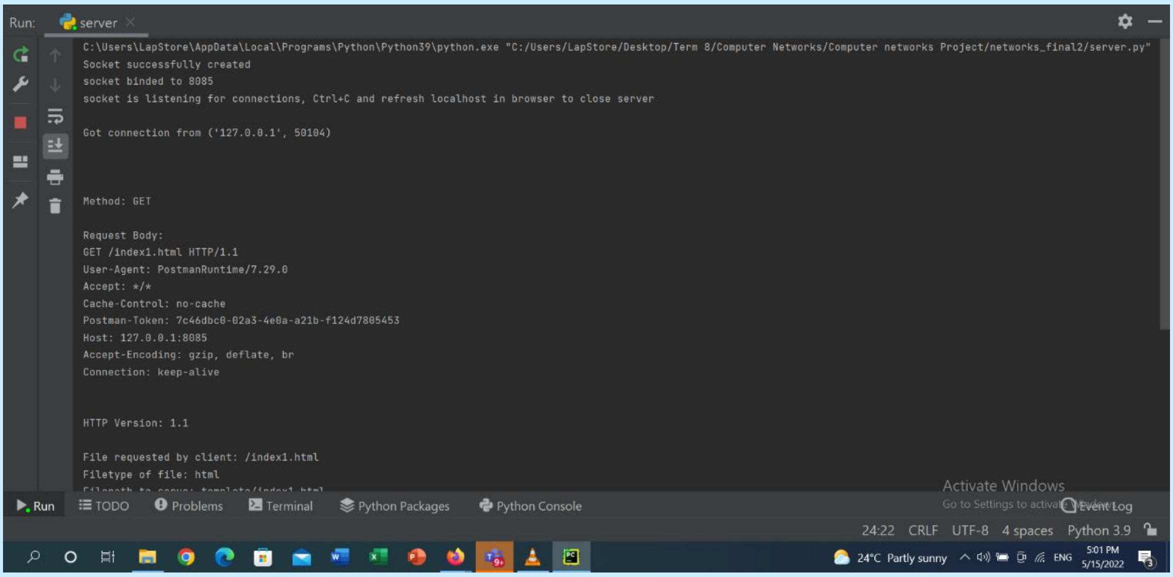


127.0.0.1:8085/index1.html

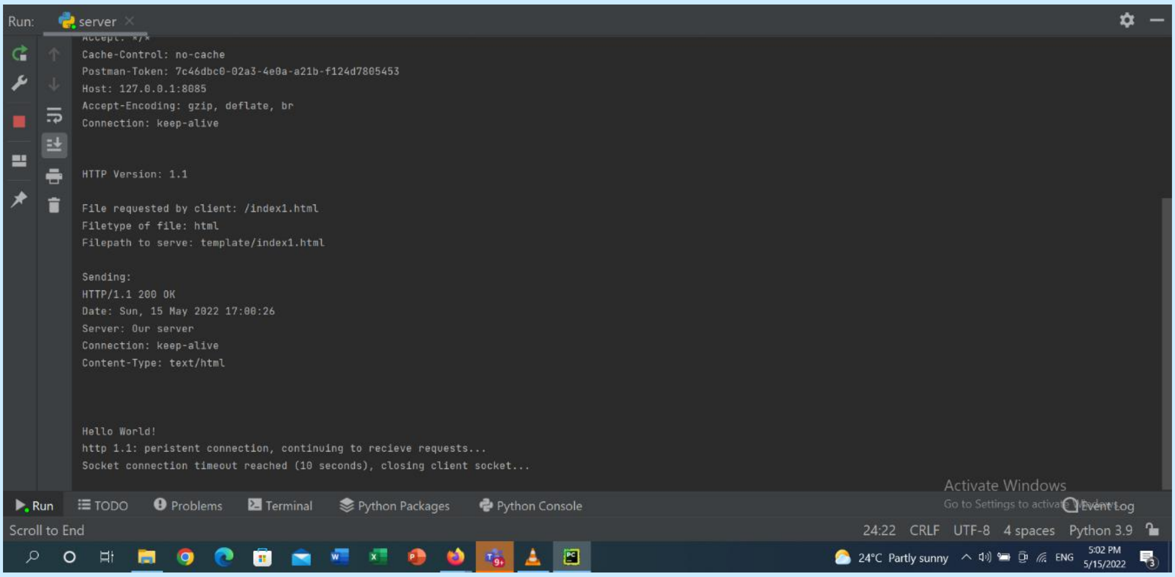
GET 127.0.0.1:8085/index1.html

Status: 200 OK Time: 10.03 s Size: 137 B

```
Hello World!
```

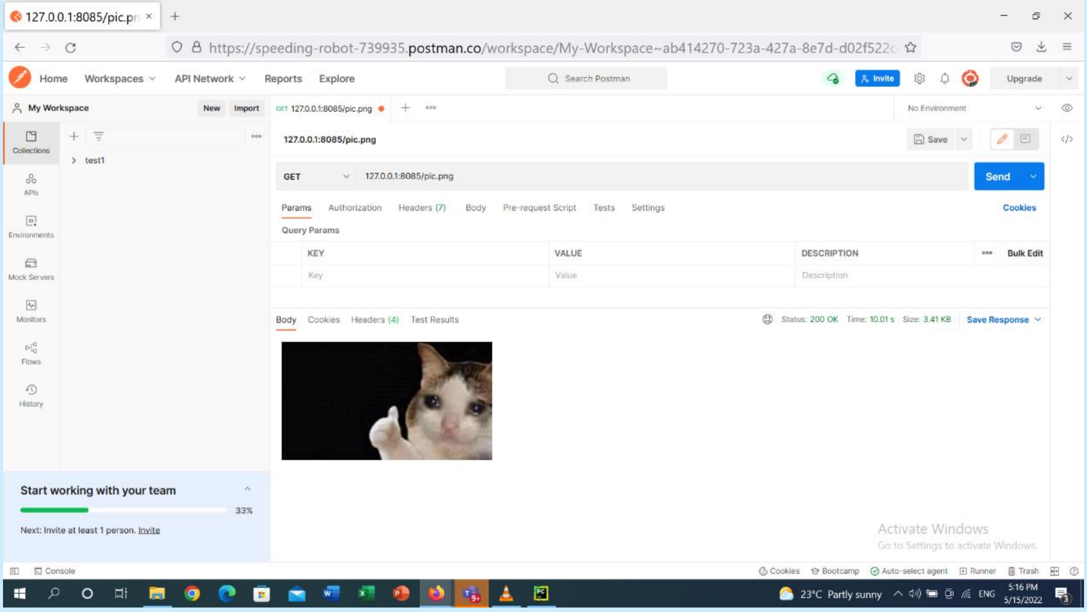


```
C:\Users\LapStore\AppData\Local\Programs\Python\Python39\python.exe "C:/Users/LapStore/Desktop/Term 8/Computer Networks/Computer networks Project/networks_final2/server.py"
Socket successfully created
socket binded to 8085
socket is listening for connections, Ctrl+C and refresh localhost in browser to close server

Got connection from ('127.0.0.1', 50104)



Method: GET

Request Body:
GET /index1.html HTTP/1.1
User-Agent: PostmanRuntime/7.29.0
Accept: */*
Cache-Control: no-cache
Postman-Token: 7c46dbc0-02a3-4e0a-a21b-f124d7805453
Host: 127.0.0.1:8085
Accept-Encoding: gzip, deflate, br
Connection: keep-alive


HTTP Version: 1.1

File requested by client: /index1.html
Filetype of file: html
Filepath to serve: template/index1.html
```



```
Accept: */*
Cache-Control: no-cache
Postman-Token: 7c46dbc0-02a3-4e0a-a21b-f124d7805453
Host: 127.0.0.1:8085
Accept-Encoding: gzip, deflate, br
Connection: keep-alive


HTTP Version: 1.1

File requested by client: /index1.html
Filetype of file: html
Filepath to serve: template/index1.html

Sending:
HTTP/1.1 200 OK
Date: Sun, 15 May 2022 17:00:26
Server: Our server
Connection: keep-alive
Content-Type: text/html



Hello World!
http 1.1: peristent connection, continuing to recieve requests...
Socket connection timeout reached (10 seconds), closing client socket...
```

GET request (png file):



https://speeding-robot-739935.postman.co/workspace/My-Workspace~ab414270-723a-427a-8e7d-d02f522c

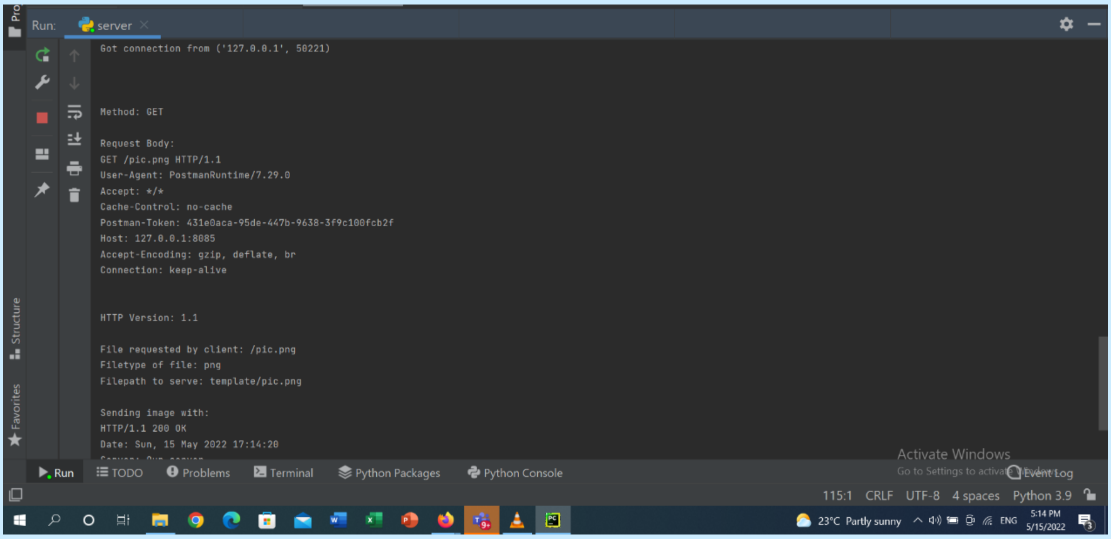
127.0.0.1:8085/pic.png

GET 127.0.0.1:8085/pic.png

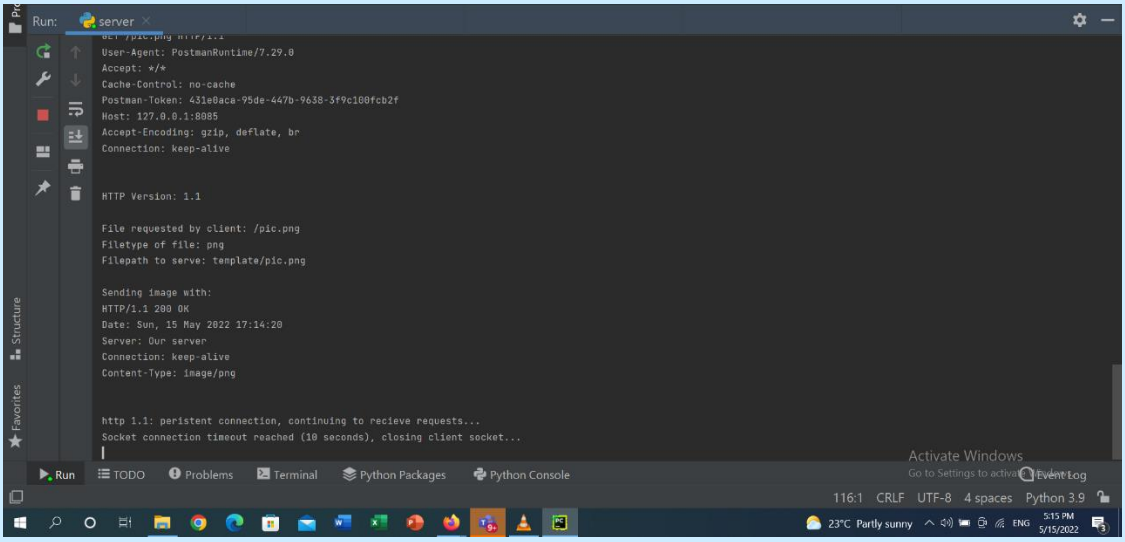
Query Params

KEY	VALUE	DESCRIPTION
Key	Value	Description

Status: 200 OK Time: 10.01 s Size: 3.41 KB

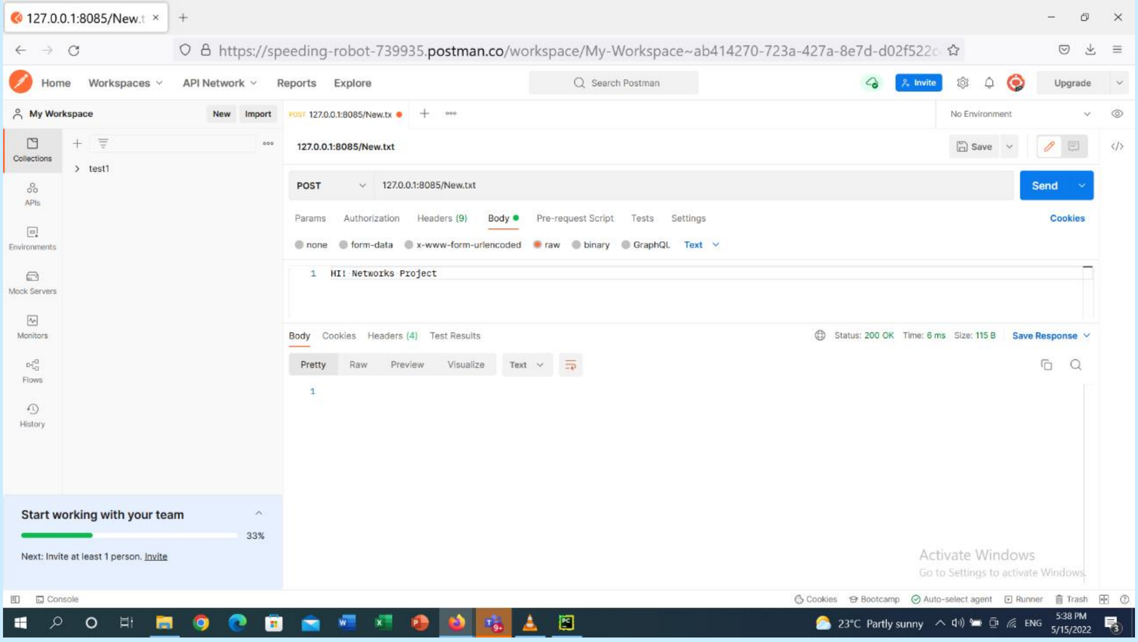


```
Got connection from ('127.0.0.1', 50221)



Method: GET

Request Body:
GET /pic.png HTTP/1.1
User-Agent: PostmanRuntime/7.29.0
Accept: */*
Cache-Control: no-cache
Postman-Token: 431e0aca-95de-447b-9638-3f9c100fcb2f
Host: 127.0.0.1:8085
Accept-Encoding: gzip, deflate, br
Connection: keep-alive


HTTP Version: 1.1

File requested by client: /pic.png
Filetype of file: png
Filepath to serve: template/pic.png

Sending image with:
HTTP/1.1 200 OK
Date: Sun, 15 May 2022 17:14:20
```



```
User-Agent: PostmanRuntime/7.29.0
Accept: */*
Cache-Control: no-cache
Postman-Token: 431e0aca-95de-447b-9638-3f9c100fcb2f
Host: 127.0.0.1:8085
Accept-Encoding: gzip, deflate, br
Connection: keep-alive


HTTP Version: 1.1

File requested by client: /pic.png
Filetype of file: png
Filepath to serve: template/pic.png

Sending image with:
HTTP/1.1 200 OK
Date: Sun, 15 May 2022 17:14:20
Server: Our server
Connection: keep-alive
Content-Type: image/png


http 1.1: peristent connection, continuing to recieve requests...
Socket connection timeout reached (10 seconds), closing client socket...
```

POST request (txt file):



https://speeding-robot-739935.postman.co/workspace/My-Workspace~ab414270-723a-427a-8e7d-d02f522c

POST 127.0.0.1:8085/New.txt

127.0.0.1:8085/New.txt

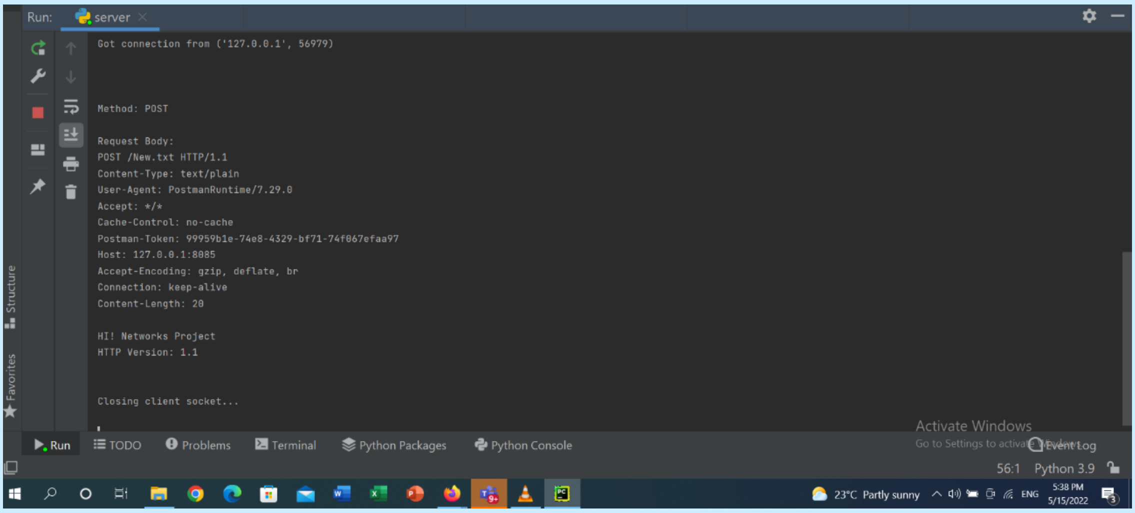
POST 127.0.0.1:8085/New.txt

○ none ○ form-data ○ x-www-form-urlencoded ● raw ○ binary ○ GraphQL Text

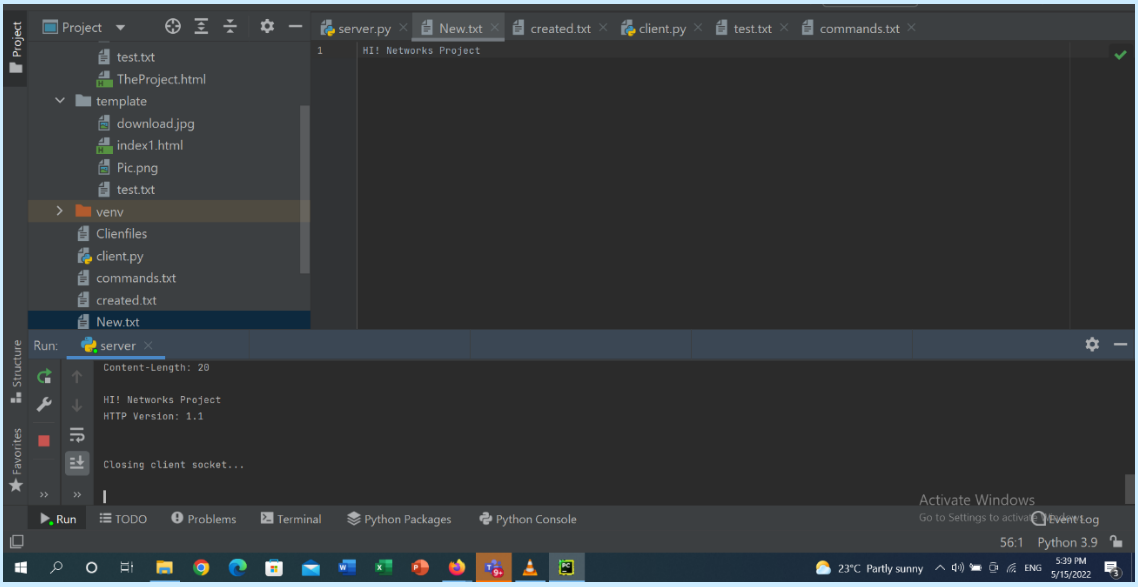
```
HI! Networks Project
```

Body

Status: 200 OK Time: 6 ms Size: 115 B



```
Got connection from ('127.0.0.1', 56979)



Method: POST

Request Body:
POST /New.txt HTTP/1.1
Content-Type: text/plain
User-Agent: PostmanRuntime/7.29.0
Accept: */*
Cache-Control: no-cache
Postman-Token: 99959b1e-74e8-4329-bf71-74f067efaa97
Host: 127.0.0.1:8085
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
Content-Length: 20

HI! Networks Project
HTTP Version: 1.1


Closing client socket...
```



```
HI! Networks Project
```

```
Content-Length: 20

HI! Networks Project
HTTP Version: 1.1


Closing client socket...
```

HTTP web client:

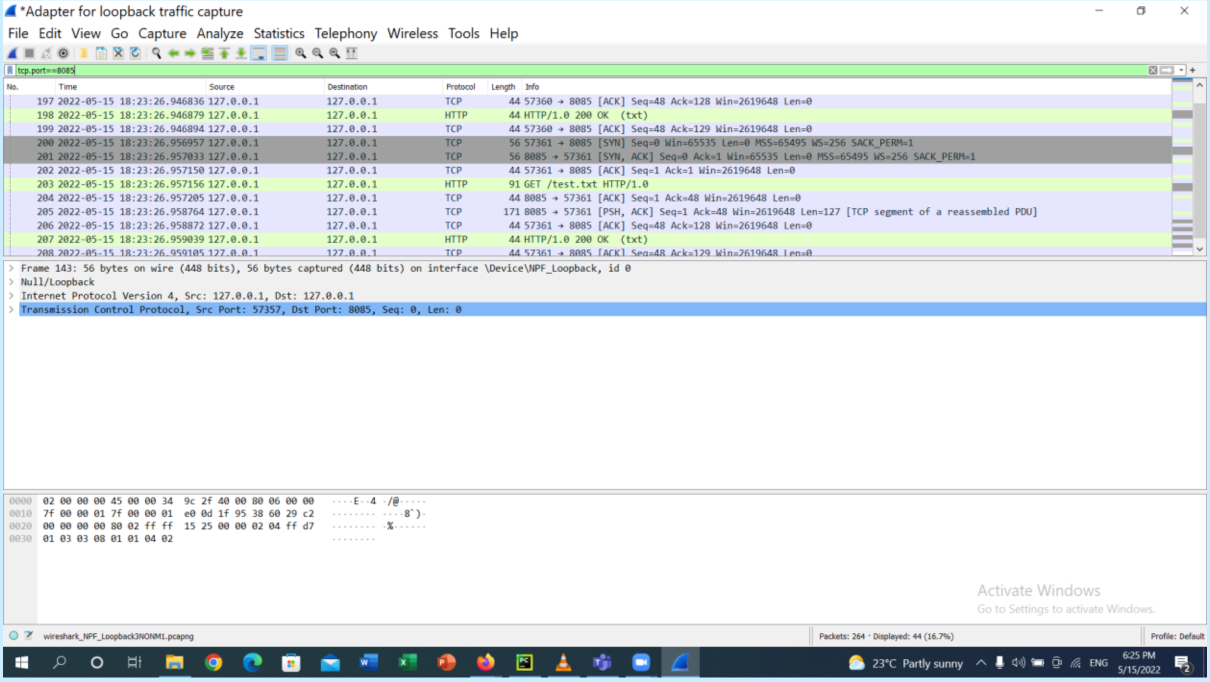
We made our web client using selectors, the client reads a (commands.txt) file in which commands are written in the following format (GET/POST file-name host-name {port-number}). These commands are then parsed to generate the HTTP requests accordingly.

Once the request is created it's sent to our start_connection function to initiate connection with the wanted host.

We tested our client on both our server and info.cer.ch as well as wireshark test.

Client sample runs:

Wireshark testing:


*Adapter for loopback traffic capture
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
tcp.port==8085

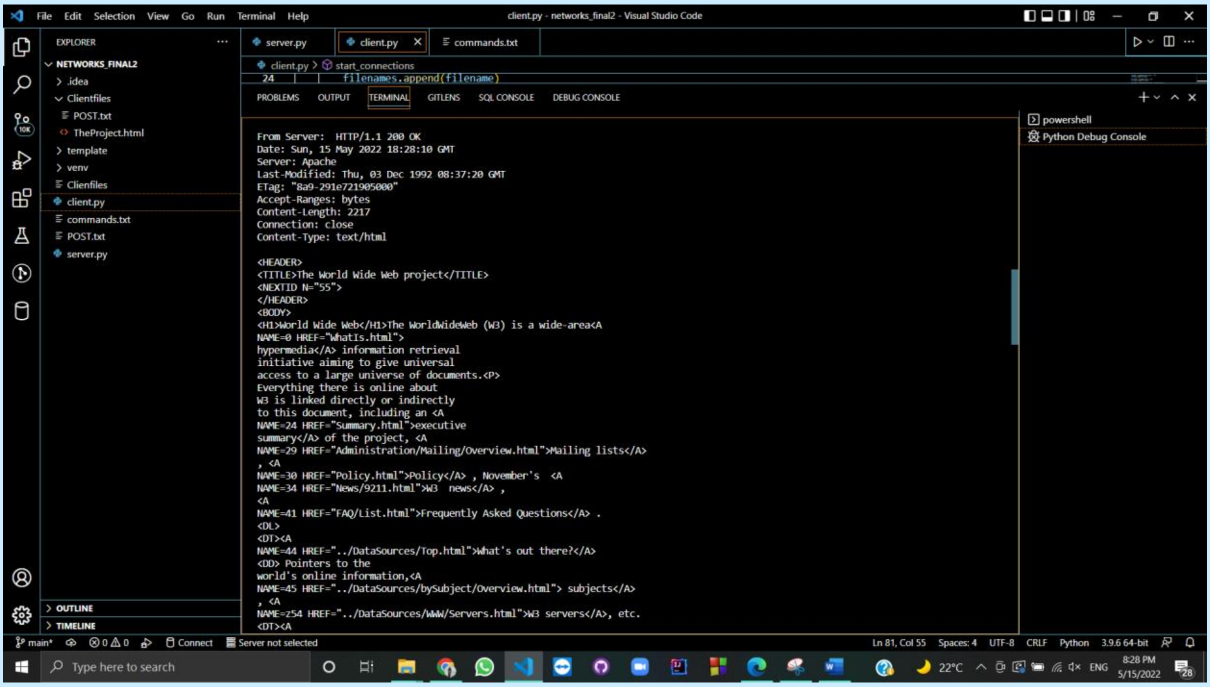
```
No.  Time                       Source    Destination Protocol Length Info
 197 2022-05-15 18:23:26.946836 127.0.0.1 127.0.0.1   TCP       44 57360 → 8085 [ACK] Seq=48 Ack=128 Win=2619648 Len=0
 198 2022-05-15 18:23:26.946879 127.0.0.1 127.0.0.1   HTTP      44 HTTP/1.0 200 OK  (txt)
 199 2022-05-15 18:23:26.946894 127.0.0.1 127.0.0.1   TCP       44 57360 → 8085 [ACK] Seq=48 Ack=129 Win=2619648 Len=0
 200 2022-05-15 18:23:26.956957 127.0.0.1 127.0.0.1   TCP       56 57361 → 8085 [SYN] Seq=0 Win=65535 Len=0 MSS=65495 WS=256 SACK_PERM=1
 201 2022-05-15 18:23:26.957033 127.0.0.1 127.0.0.1   TCP       56 8085 → 57361 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=65495 WS=256 SACK_PERM=1
 202 2022-05-15 18:23:26.957150 127.0.0.1 127.0.0.1   TCP       44 57361 → 8085 [ACK] Seq=1 Ack=1 Win=2619648 Len=0
 203 2022-05-15 18:23:26.957156 127.0.0.1 127.0.0.1   HTTP      91 GET /test.txt HTTP/1.0
 204 2022-05-15 18:23:26.957205 127.0.0.1 127.0.0.1   TCP       44 8085 → 57361 [ACK] Seq=1 Ack=48 Win=2619648 Len=0
 205 2022-05-15 18:23:26.958764 127.0.0.1 127.0.0.1   TCP      171 8085 → 57361 [PSH, ACK] Seq=1 Ack=48 Win=2619648 Len=127 [TCP segment of a reassembled PDU]
 206 2022-05-15 18:23:26.958872 127.0.0.1 127.0.0.1   TCP       44 57361 → 8085 [ACK] Seq=48 Ack=128 Win=2619648 Len=0
 207 2022-05-15 18:23:26.959039 127.0.0.1 127.0.0.1   HTTP      44 HTTP/1.0 200 OK  (txt)
 208 2022-05-15 18:23:26.959105 127.0.0.1 127.0.0.1   TCP       44 57361 → 8085 [ACK] Seq=48 Ack=129 Win=2619648 Len=0
```

```
> Frame 143: 56 bytes on wire (448 bits), 56 bytes captured (448 bits) on interface \Device\NPF_Loopback, id 0
> Null/Loopback
> Internet Protocol Version 4, Src: 127.0.0.1, Dst: 127.0.0.1
> Transmission Control Protocol, Src Port: 57357, Dst Port: 8085, Seq: 0, Len: 0
```

```
0000  02 00 00 00 45 00 00 34  9c 2f 40 00 80 06 00 00   ····E··4 ·/@·····
0010  7f 00 00 01 7f 00 00 01  e0 0d 1f 95 38 60 29 c2   ········ ····8`)·
0020  00 00 00 00 80 02 ff ff  15 25 00 00 02 04 ff d7   ········ ·%······
0030  01 03 03 08 01 01 04 02                            ········
```

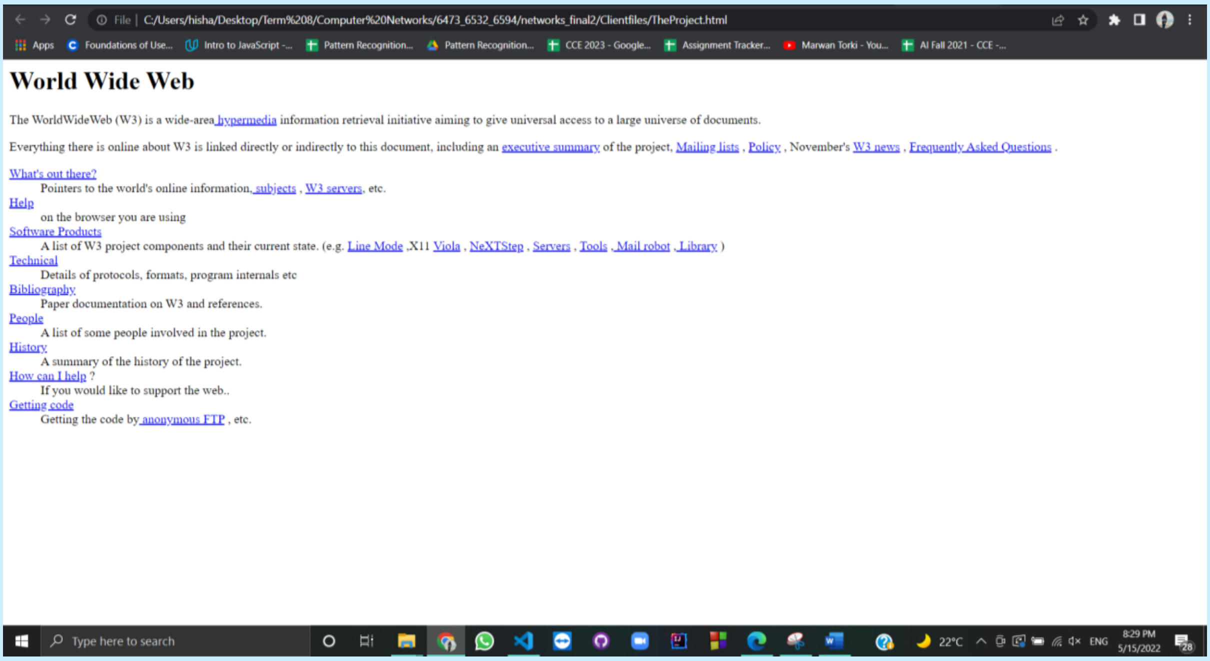
wireshark_NPF_Loopback3NONM1.pcapng | Packets: 264 · Displayed: 44 (16.7%) | Profile: Default

Testing our client with info.cern.ch:


File Edit Selection View Go Run Terminal Help
client.py - networks_final2 - Visual Studio Code
EXPLORER
NETWORKS_FINAL2
.idea
Clientfiles
POST.txt
TheProject.html
template
venv
Clienfiles
client.py
commands.txt
POST.txt
server.py
OUTLINE
TIMELINE
server.py
client.py
commands.txt
client.py > start_connections
PROBLEMS OUTPUT TERMINAL GITLENS SQL CONSOLE DEBUG CONSOLE
powershell
Python Debug Console
main* ⊗ 0 ⚠ 0 Connect Server not selected
Ln 81, Col 55 Spaces: 4 UTF-8 CRLF Python 3.9.6 64-bit

```
24        filenames.append(filename)
```

```
From Server:  HTTP/1.1 200 OK
Date: Sun, 15 May 2022 18:28:10 GMT
Server: Apache
Last-Modified: Thu, 03 Dec 1992 08:37:20 GMT
ETag: "8a9-291e721905000"
Accept-Ranges: bytes
Content-Length: 2217
Connection: close
Content-Type: text/html

<HEADER>
<TITLE>The World Wide Web project</TITLE>
<NEXTID N="55">
</HEADER>
<BODY>
<H1>World Wide Web</H1>The WorldWideWeb (W3) is a wide-area<A
NAME=0 HREF="WhatIs.html">
hypermedia</A> information retrieval
initiative aiming to give universal
access to a large universe of documents.<P>
Everything there is online about
W3 is linked directly or indirectly
to this document, including an <A
NAME=24 HREF="Summary.html">executive
summary</A> of the project, <A
NAME=29 HREF="Administration/Mailing/Overview.html">Mailing lists</A>
, <A
NAME=30 HREF="Policy.html">Policy</A> , November's  <A
NAME=34 HREF="News/9211.html">W3  news</A> ,
<A
NAME=41 HREF="FAQ/List.html">Frequently Asked Questions</A> .
<DL>
<DT><A
NAME=44 HREF="../DataSources/Top.html">What's out there?</A>
<DD> Pointers to the
world's online information,<A
NAME=45 HREF="../DataSources/bySubject/Overview.html"> subjects</A>
, <A
NAME=z54 HREF="../DataSources/WWW/Servers.html">W3 servers</A>, etc.
<DT><A
```



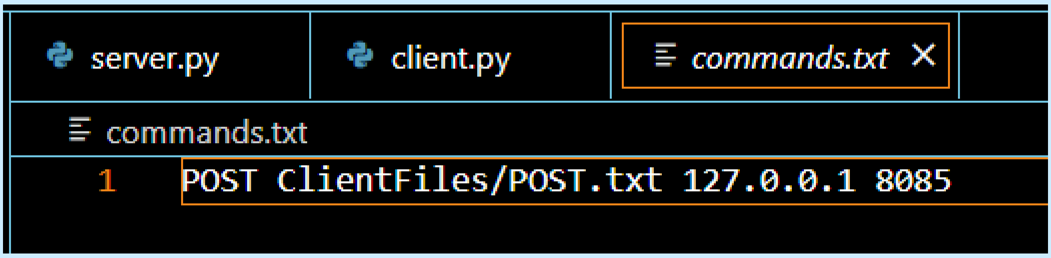
World Wide Web

The WorldWideWeb (W3) is a wide-area hypermedia information retrieval initiative aiming to give universal access to a large universe of documents.

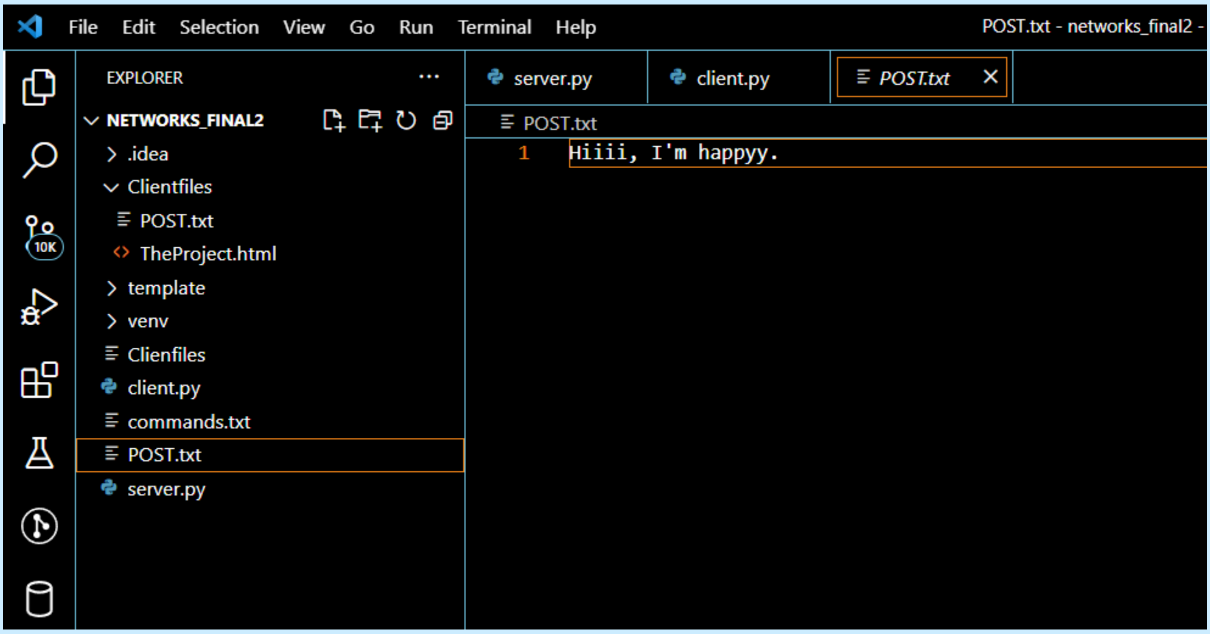
Everything there is online about W3 is linked directly or indirectly to this document, including an executive summary of the project, Mailing lists , Policy , November's W3 news , Frequently Asked Questions .

What's out there?
    Pointers to the world's online information, subjects , W3 servers, etc.
Help
    on the browser you are using
Software Products
    A list of W3 project components and their current state. (e.g. Line Mode ,X11 Viola , NeXTStep , Servers , Tools , Mail robot , Library)
Technical
    Details of protocols, formats, program internals etc
Bibliography
    Paper documentation on W3 and references.
People
    A list of some people involved in the project.
History
    A summary of the history of the project.
How can I help ?
    If you would like to support the web..
Getting code
    Getting the code by anonymous FTP , etc.

We couldn't test the POST requests of our client on any known server because simply, we wouldn't be able to see that the file was actually created in the server, so we tested this part on our own server.



```
POST ClientFiles/POST.txt 127.0.0.1 8085
```

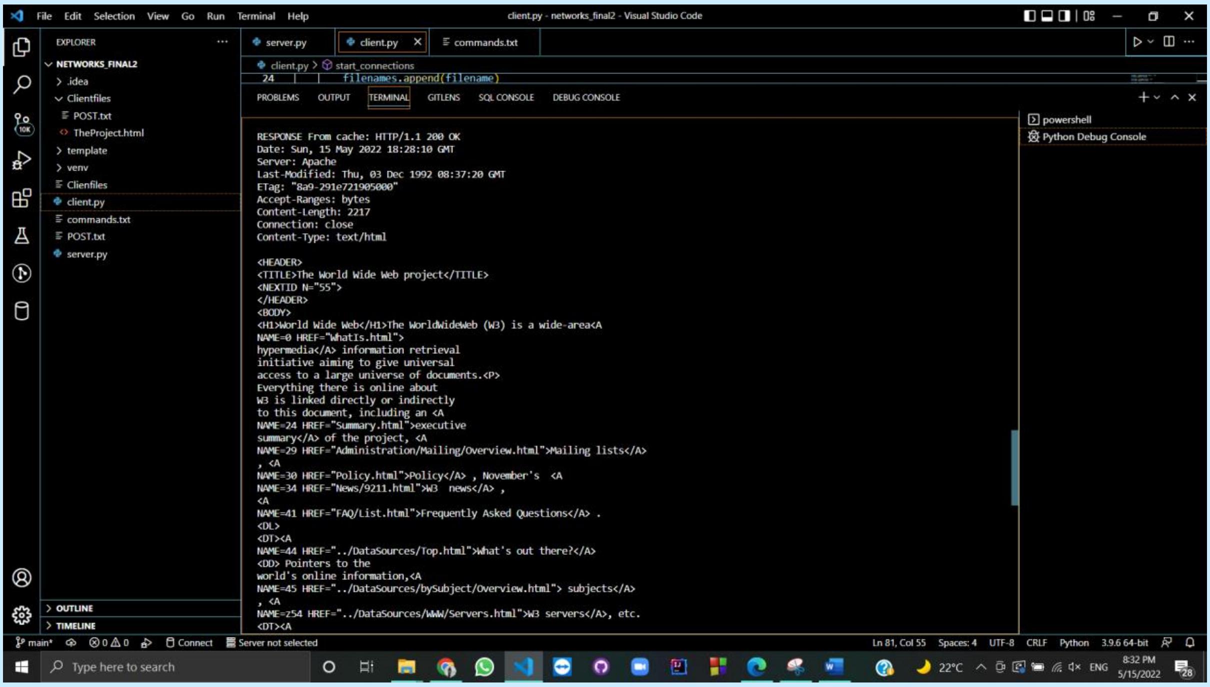


```
Hiiii, I'm happyy.
```

Bonus:

Additionally, we implemented a simple cache in our client to prevent uncalled for connections when the same request has been made before.

We just saw how we connected our client to info.cern.ch, we'll try to send the same exact request and we'll see that our client gets the response this time from the cache without connecting to the server again.



```
            filenames.append(filename)
```

```
RESPONSE From cache: HTTP/1.1 200 OK
Date: Sun, 15 May 2022 18:28:10 GMT
Server: Apache
Last-Modified: Thu, 03 Dec 1992 08:37:20 GMT
ETag: "8a9-291e721905000"
Accept-Ranges: bytes
Content-Length: 2217
Connection: close
Content-Type: text/html

<HEADER>
<TITLE>The World Wide Web project</TITLE>
<NEXTID N="55">
</HEADER>
<BODY>
<H1>World Wide Web</H1>The WorldWideWeb (W3) is a wide-area<A
NAME=0 HREF="WhatIs.html">
hypermedia</A> information retrieval
initiative aiming to give universal
access to a large universe of documents.<P>
Everything there is online about
W3 is linked directly or indirectly
to this document, including an <A
NAME=24 HREF="Summary.html">executive
summary</A> of the project, <A
NAME=29 HREF="Administration/Mailing/Overview.html">Mailing lists</A>
, <A
NAME=30 HREF="Policy.html">Policy</A> , November's  <A
NAME=34 HREF="News/9211.html">W3  news</A> ,
<A
NAME=41 HREF="FAQ/List.html">Frequently Asked Questions</A> .
<DL>
<DT><A
NAME=44 HREF="../DataSources/Top.html">What's out there?</A>
<DD> Pointers to the
world's online information,<A
NAME=45 HREF="../DataSources/bySubject/Overview.html"> subjects</A>
, <A
NAME=z54 HREF="../DataSources/WWW/Servers.html">W3 servers</A>, etc.
<DT><A
```