

**COEN 317 : DISTRIBUTED SYSTEMS**  
**PROGRAMMING ASSIGNMENT - 1**

NAME : PUJITHA KALLU

SCU ID : W1653660

---

***Description:***

The goal of this programming assignment is to build a working web server. The assignment aims to involve concepts in distributed programming, client-server interactions, and the challenges involved in creating a web server.

Upon establishing a connection, the server initiates a dedicated thread to process image files, ensuring that both HTML pages and image requests are handled with optimal efficiency. This threading approach allows the server to efficiently manage the potentially resource-intensive task of processing images without affecting the responsiveness of the overall server.

One notable aspect of the program is its implementation of a connection timeout mechanism. Each individual connection is maintained for a specified duration of 10 seconds. This timeout ensures that connections are not kept open indefinitely, contributing to resource management and preventing potential bottlenecks or resource exhaustion.

In terms of protocol adherence, the server follows the **HTTP/1.1** specification when responding to incoming HTTP requests. Also Adding 'protocol 1.0' to the web server program indicates the use of HTTP version 1.0, an early version of the Hypertext Transfer Protocol characterized by stateless connections, limited header information, and individual connections for each request-response cycle. This adherence to standards ensures compatibility and interoperability with a wide range of web clients and browsers.

***Program file consists of:***

1. *Functionality:*

- The program acts as a functional web server in Python.
- It serves files from a specified directory to clients upon receiving HTTP requests.

2. *HTTP Methods:*

- Handles basic HTTP methods, particularly GET requests.

3. *HTTP Status Codes:*

- Supports HTTP status codes such as 200 (OK), 400 (Bad Request), 403 (Forbidden), and 404 (File Not Found), 501(not\_supported).

4. *Response Headers:*

- Supports essential response headers:
- Content-Type
- Content-Length
- Date

5. *File Types:*

- Supports various file types including .html, .txt, .jpg, .gif, .png, .mp4, and .pdf.

6. *Concurrency:*

- Utilizes multi-threading to handle multiple client requests simultaneously.
- Includes a default idle timeout of 500 seconds for connections.

***Submitted Files:***

- Server.py
- Media
- Index.html
- Index3.html
- Error
- Logo.png
- 404\_ERROR.gif
- Gfs.pdf
- Webserver.jpeg
- Webserver.txt
- Socket.jpg
- video.mp4
- Allfiles.html
- Readme.pdf
- Makefile

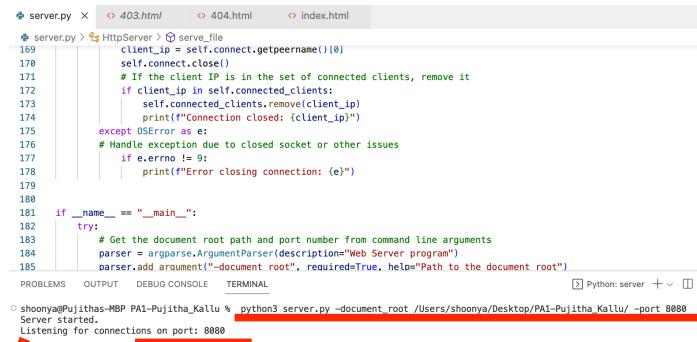
### **Run code:**

- python3 server.py -document\_root "<PATH\_TO\_DIRECTORY\_ROOT>" -port <PORT>

Example: python3 server.py -document\_root /Users/shoonya/Desktop/PA1-Pujitha\_Kallu/ -port 8080

- Open the web browser/client that can connect to the port where the server is configured to run and can initiate HTTP requests.

Example: <http://localhost:8080>



```
server.py 403.html 404.html index.html
server.py > HttpServer > serve_file
169     client_ip = self.connect.getpeername()[1]
170     self.connect.close()
171     # If the client IP is in the set of connected clients, remove it
172     if client_ip in self.connected_clients:
173         self.connected_clients.remove(client_ip)
174         print(f"Connection closed: {client_ip}")
175     except OSError as e:
176         # Handle exception due to closed socket or other issues
177         if e.errno != 9:
178             print(f"Error closing connection: {e}")
179
180 if __name__ == "__main__":
181     try:
182         # Get the document root path and port number from command line arguments
183         parser = argparse.ArgumentParser(description="Web Server program")
184         parser.add_argument("-document_root", required=True, help="Path to the document root")
185
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL Python: server + □
shoonya@Pujithas-MBP PA1_Pujitha_Kallu % python3 server.py -document_root /Users/shoonya/Desktop/PA1_Pujitha_Kallu/ -port 8080
Server started.
Listening for connections on port: 8080
```

### **MULTI THREADING:**

Introducing multithreading to a web server enables concurrent processing of multiple client requests via separate threads. The Global Interpreter Lock (GIL) in Python, however, limits true parallelism, favoring scenarios with I/O-bound tasks where threads can efficiently manage simultaneous connections. While beneficial for responsiveness in handling numerous requests concurrently, GIL's impact is more pronounced in CPU-bound tasks, potentially constraining performance gains in parallel execution.

```
shoonya@Pujithas-MBP PA1_Pujitha_Kallu % python3 server.py -document_root /Users/shoonya/Desktop/PA1_Pujitha_Kallu/ -port 8080
Server started.
Listening for connections on port: 8080

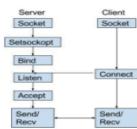
Connection opened: 127.0.0.1
Current thread name is Thread-1 and file opened /Users/shoonya/Desktop/PA1_Pujitha_Kallu/allfiles.html
Current thread name is Thread-2 and file opened /Users/shoonya/Desktop/PA1_Pujitha_Kallu/socket.jpg
HTTP 404 Error: /websever.jpeg cannot be found!
Current thread name is Thread-4 and file opened /Users/shoonya/Desktop/PA1_Pujitha_Kallu/logo.png
Current thread name is Thread-5 and file opened /Users/shoonya/Desktop/PA1_Pujitha_Kallu/404_ERROR.gif
Current thread name is Thread-6 and file opened /Users/shoonya/Desktop/PA1_Pujitha_Kallu/webserver.txt
Current thread name is Thread-7 and file opened /Users/shoonya/Desktop/PA1_Pujitha_Kallu/gfs.pdf
Current thread name is Thread-8 and file opened /Users/shoonya/Desktop/PA1_Pujitha_Kallu/video.mp4
```



## File Types Example

## Image Files

JPG Image:



JPEG Image:



PNG Image:

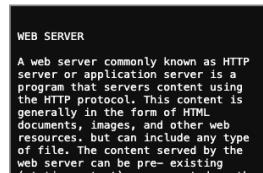


**GIF Image:**



## Text Files

#### Text File:



## Other Files

PDF File:



1. Open browser and enter localhost: [port number] . The server responds by serving the content associated with the default endpoint with 200 status code:

localhost:8080

Name	Url	S...	Protocol	Type	Initiator	Size	T...	Waterfall
localhost	http://...	2...	http/1.1	do...	Other	45.2 ...	1...	
scu.js	http://localhost:8080/	2...	h3	sc...	:8080/54	(me... 0...	0...	
instant-notification.js	https://...	2...	h2	sc...	:8080/325	(me... 0...	0...	
p.css?ts=1&kg=gb1rhy&ht=ik...	https://...	2...	h2	sty...	gb1rhy.css	(me... 0...	0...	
all.css	https://...	2...	h2	sty...	:8080/44	(disk... 1...	1...	
AdobeStock_627644...	http://...	2...	http/1.1	jpeg	:8080/533	127 kB 1...	1...	
MissionWeekFeature_...	https://...	2...	http/1.1	png	:8080/545	1.7 MB 1...	1...	
gtm.js?cid=GTM-579K...	https://...	3...	http/1.1	sc...	(index)321	0 B 1...	1...	
760-MA15_Fountain_...	http://...	2...	http/1.1	jpeg	(index)557	157 kB 2...	2...	
simone-savoldi-mJdu...	http://...	2...	http/1.1	jpeg	(index)573	58.2 ...	1...	
ENG.jpg	https://...	2...	h2	jpeg	:8080/629	(me... 1...	1...	
pink-car-760x480.jpg	http://...	2...	http/1.1	jpeg	(index)583	44.7 ... 2...	1...	
BasketballFans-small...	https://...	2...	h2	jpeg	:8080/644	(me... 0...	0...	
bestmag-760x480-76...	http://...	2...	http/1.1	jpeg	(index)593	45.0 ... 2...	1...	
campus-small.jpg	https://...	2...	h2	jpeg	:8080/646	(me... 0...	0...	
SCU-Commencemen...	https://...	2...	h2	jpeg	:8080/649	(me... 1...	1...	
lwcw.js	http://...	3...	http/1.1	sc...	(index)611	0 B 1...	1...	
campaign-banner.png	https://...	2...	h2	image	:8080/651	0 B 1...	1...	

localhost:8080

Name	General	Headers	Preview	Response	Initiator	Timing	...
localhost	Request URL: http://localhost:8080/						
scu.js	Request Method: GET						
instant-notification.js	Status Code: 200 OK						
p.css?ts=1&kg=gb1rhy&ht=ik...	Remote Address: 127.0.0.1:8080						
all.css	Referrer Policy: strict-origin-when-cross-origin						
AdobeStock_62764489...	Request Headers	Raw					
MissionWeekFeature_AlienB...	Request URL:						
gtm.js?cid=GTM-579KDR	Request Method:						
760-MA15_Fountain_5_15_0...	Status Code:	200 OK					
simone-savoldi-mJdu07CzO...	Remote Address:						
ENG.jpg	Referrer Policy:	strict-origin-when-cross-origin					
pink-car-760x480.jpg	Response Headers	Raw					
BasketballFans-small.jpg	Request URL:						
bestmag-760x480-760x480.jpg	Request Method:						
campus-small.jpg	Status Code:	45070					
SCU-Commencement-2019...	Remote Address:						
lwcw.js	Content-Length:	45070					
dancers-small.jpg	Content-Type:	text/html					
cultural-small.jpg	Date:	Wed, 07 Feb 2024 06:29:48 GMT					
campaign-banner.png	Server:	Python HTTP Server					
	Request Headers	Raw					
	Accept:	text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng;q=0.8,application/signed-exchange;q=0.8					
	Accept-Encoding:	gzip, deflate, br					
	Accept-Language:	en-GB,en-US;q=0.9,en;q=0.8					
	Cache-Control:	max-age=0					
	Connection:	keep-alive					

- Open browser and enter localhost: [port number]/index.html. The server responds by serving the content associated with the default endpoint (/index.html) same as above with 200 status code within our document root directory.

localhost:8080/index.html

Name	Url	S...	Protocol	Type	Initiator	Size	T...
Index.html	http://...	2...	http/1.1	do...	Other	45.2 ...	1...
scu.js	https://...	2...	h3	sc...	index.html:53	0...	...
instant-notification.js	https://...	2...	h2	sc...	index.html:...	0...	...
ENG.jpg	https://...	2...	h2	jpeg	index.html:...	0...	...
BasketballFans-small...	https://...	2...	h2	jpeg	index.html:...	0...	...
campus-small.jpg	https://...	2...	h2	jpeg	index.html:...	0...	...
SCU-Commencemen...	https://...	2...	h2	jpeg	index.html:...	0...	...
dancers-small.jpg	https://...	2...	h2	jpeg	index.html:...	0...	...
cultural-small.jpg	https://...	2...	h2	jpeg	index.html:...	0...	...
gzb1rhy.css	https://...	2...	h2	sty...	index.html:42	(disk... 5...	...
all.css	https://...	2...	h2	sty...	index.html:43	(disk... 2...	...
scu.css	https://...	2...	h2	sty...	index.html:47	(disk... 5...	...
AdobeStock_627644...	https://...	2...	http/1.1	jpeg	index.html:...	127 KB 1...	...
MissionWeekFeature...	https://...	2...	http/1.1	png	index.html:...	1.7 MB 2...	...
p.css?ts=1&e=gzb1rhy...	https://...	2...	h2	sty...	gzb1rhy.css:1	(me... 1...	...
760-MA15_Fountain...	https://...	2...	http/1.1	jpeg	index.html:...	157 KB 6...	...
simone-savoldi-nJDu...	https://...	2...	http/1.1	jpeg	index.html:...	58.2 ... 1...	...
pink-car-760x480.jpg	https://...	2...	http/1.1	jpeg	index.html:...	44.7 ... 1...	...
twrcw.js	http://...	3...	http/1.1	sc...	index.html:...	0 B 2...	...

localhost:8080/index.html

Name	Headers	Preview	Response	Initiator	Timing	Co...
index.html	<div>General</div> <div>Request URL: http://localhost:8080/index.html</div> <div>Request Method: GET</div> <div>Status Code: 200 OK</div> <div>Remote Address: 127.0.0.1:8080</div> <div>Referer Policy: strict-origin-when-cross-origin</div> <div>Response Headers</div> <div>Content-Length: 45070</div> <div>Content-Type: text/html</div> <div>Date: Wed, 07 Feb 2024 06:32:30 GMT</div> <div>Server: Python HTTP Server</div> <div>Request Headers</div> <div>Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7</div>					

3. The web server after removing read access to the file, the server will respond with a 403 Forbidden status code, indicating insufficient permissions to access the file.

localhost:8080/logo.png

Name	Headers	Preview	Response	Initiator	Timing	Co...
logo.png	<div>General</div> <div>Request URL: http://localhost:8080/logo.png</div> <div>Request Method: GET</div> <div>Status Code: 403 Forbidden</div> <div>Remote Address: 127.0.0.1:8080</div> <div>Referer Policy: strict-origin-when-cross-origin</div> <div>Response Headers</div> <div>Content-Length: 715</div> <div>Content-Type: text/html</div> <div>Date: Wed, 07 Feb 2024 06:45:28 GMT</div> <div>Server: Python HTTP Server</div> <div>Request Headers</div> <div>Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7</div> <div>Accept-Encoding: gzip, deflate, br</div> <div>Accept-Language: en-GB,en-US;q=0.9,en;q=0.8</div>					

→ Before removing read access to logo.png file

```
shoonya@Pujithas-MBP PA1_Pujitha_Kallu % ls -altr
total 23192
-rwxrwxrwx@ 1 shoonya staff 62686 Oct 25 16:50 logo.png
drwxrwx---@ o shoonya staff 192 Oct 25 10:51 media
-rw-r--r--@ 1 shoonya staff 5162143 Oct 25 19:40 video.mp4
drwxr-xr-x@ 7 shoonya staff 224 Feb 3 00:42 error
-rw-r--r--@ 1 shoonya staff 2167644 Feb 3 15:39 404_ERROR.gif
-rw-r--r--@ 1 shoonya staff 275941 Feb 3 16:04 gfs.pdf
-rw-r--r--@ 1 shoonya staff 40927 Feb 3 22:55 index3.html
-rw-r--r--@ 1 shoonya staff 389541 Feb 3 23:48 webserver.jpeg
-rw-r--r--@ 1 shoonya staff 380 Feb 3 23:52 webserver.txt
-rw-r--r--@ 1 shoonya staff 22718 Feb 3 23:56 socket.jpg
-rw-r--r--@ 1 shoonya staff 927 Feb 6 08:42 allfiles.html
-rw-r--r--@ 1 shoonya staff 1551 Feb 6 14:36 makefile.rtf
-rw-r--r--@ 1 shoonya staff 40927 Feb 6 15:16 index.html
-rw-r--r--@ 1 shoonya staff 8196 Feb 6 15:35 .DS_Store
-rw-r--r--@ 1 shoonya staff 3646735 Feb 6 16:19 Readme.pdf
-rw-r--r--@ 1 shoonya staff 9491 Feb 6 17:04 server.py
drwxr-xr-x@ 19 shoonya staff 608 Feb 6 17:52 .
-rw-r--r--@ 1 shoonya staff 7386 Feb 6 19:05 webserver.py
drwx-----+ 27 shoonya staff 864 Feb 6 22:50 ..
shoonya@Pujithas-MBP PA1_Pujitha_Kallu %
```

→ After removing read permissions to logo.png file

```
shoonya@Pujithas-MBP PA1_Pujitha_Kallu % ls -altr
total 23192
--wx-wx-wx 1 shoonya staff 62686 Oct 25 16:50 logo.png
drwxrwx---@ o shoonya staff 192 Oct 25 10:51 media
-rw-r--r--@ 1 shoonya staff 5162143 Oct 25 19:40 video.mp4
drwxr-xr-x@ 7 shoonya staff 224 Feb 3 00:42 error
-rw-r--r--@ 1 shoonya staff 2167644 Feb 3 15:39 404_ERROR.gif
-rw-r--r--@ 1 shoonya staff 275941 Feb 3 16:04 gfs.pdf
-rw-r--r--@ 1 shoonya staff 40927 Feb 3 22:55 index3.html
-rw-r--r--@ 1 shoonya staff 389541 Feb 3 23:48 webserver.jpeg
-rw-r--r--@ 1 shoonya staff 380 Feb 3 23:52 webserver.txt
-rw-r--r--@ 1 shoonya staff 22718 Feb 3 23:56 socket.jpg
-rw-r--r--@ 1 shoonya staff 927 Feb 6 08:42 allfiles.html
-rw-r--r--@ 1 shoonya staff 1551 Feb 6 14:36 makefile.rtf
-rw-r--r--@ 1 shoonya staff 40927 Feb 6 15:16 index.html
-rw-r--r--@ 1 shoonya staff 8196 Feb 6 15:35 .DS_Store
-rw-r--r--@ 1 shoonya staff 3646735 Feb 6 16:19 Readme.pdf
-rw-r--r--@ 1 shoonya staff 9491 Feb 6 17:04 server.py
drwxr-xr-x@ 19 shoonya staff 608 Feb 6 17:52 .
-rw-r--r--@ 1 shoonya staff 7386 Feb 6 19:05 webserver.py
drwx-----+ 34 shoonya staff 1088 Feb 6 22:48 ..
shoonya@Pujithas-MBP PA1_Pujitha_Kallu %
```

4. The web server results in a 404 Not Found response as the required file abc.pdf/index4.html [any file which is not present in the root directory] is not present in the specified document root directory.

The screenshot shows a browser window with the URL `localhost:8080/index4.html`. A modal dialog displays "HTTP Status Code 404 - File Not Found" and the message "The server has not found anything matching the Request-URL." Below the browser is the Network tab of the developer tools, showing a single request for "index4.html". The status code is highlighted in red as 404 File Not Found.

5. Making a bad POST request to index.html results in a 400 Bad Request response from the web server.

The screenshot shows a Postman request configuration for a POST method to `http://localhost:8080/index.html`. The Headers tab shows "Content-Type: application/x-www-form-urlencoded". The Body tab is empty. The Response tab shows a 400 Bad Request response with a content length of 914 B. The response body contains the following HTML:

```

1 <!DOCTYPE html>
2 <html>
3
4 <head>
5   <meta charset="UTF-8">
6   <title>HTTP Status Code 400 - Bad Request</title>

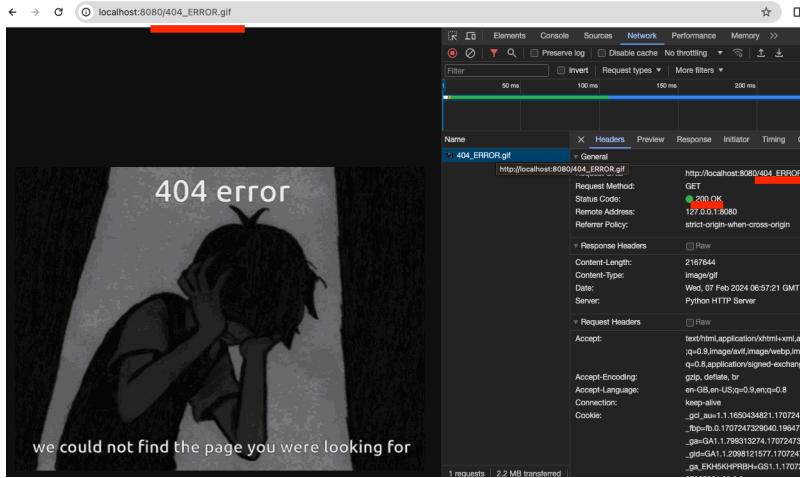
```

### ***Client opening various contents of different file:***

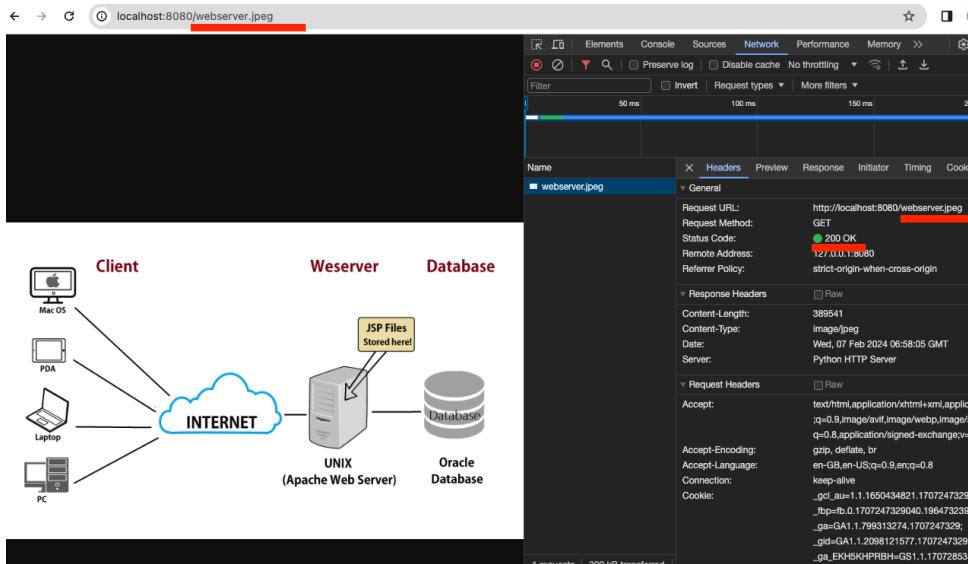
1. The server responds by serving the content of the existing text file webserver.txt from the specified document root directory.

The screenshot shows a browser window with the URL `localhost:8080/webserver.txt`. The developer tools Network tab shows a request for "webserver.txt" with a status code of 200 OK. The status code is highlighted in red as 200 OK.

2. The server responds by serving the content of the existing gif file `404_ERROR.gif` from the specified document root directory.



3. The server responds by serving the content of the existing jpeg file `webserver.jpeg` from the specified document root directory.



4. The server responds by serving the content of the existing jpg file socket.jpg from the specified root:

The screenshot shows a browser's developer tools Network tab with a request for "socket.jpg". The request details are as follows:

- Request URL: http://localhost:8080/socket.jpg
- Request Method: GET
- Status Code: 200 OK
- Remote Address: 127.0.0.1:8080
- Referrer Policy: strict-origin-when-cross-origin

The response headers include:

- Content-Length: 22718
- Content-Type: image/jpeg
- Date: Wed, 07 Feb 2024 06:59:27 GMT
- Server: Python HTTP Server

The request headers include:

- Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,\*/\*;q=0.8,application/signed-exchange;v=b2;
- Accept-Encoding: gzip, deflate, br
- Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
- Connection: keep-alive
- Cookie: \_gcl\_au=1.1.1650434821.1707247323; \_fbp=fb.0.1707247329040.196473239; \_ga=GA1.1.2098121577.1707247323; \_gid=GA1.1.2098121577.1707247323; ga\_EKH5KPRBH=GS1.1.1707285

Below the browser screenshot is a flowchart illustrating the socket communication process:

```
graph TD; Server[Server: Socket] --> Setsockopt[Setsockopt]; Setsockopt --> Bind[Bind]; Bind --> Listen[Listen]; Listen --> Accept[Accept]; Client[Client: Socket] -- Connect --> Listen; Accept --> SendRecv[Send/Recv]; Client --> SendRecv;
```

5. The server responds by serving the content of the existing png file logo.png from the specified document root directory.

The screenshot shows a browser's developer tools Network tab with a request for "logo.png". The request details are as follows:

- Request URL: http://localhost:8080/logo.png
- Request Method: GET
- Status Code: 200 OK
- Remote Address: 127.0.0.1:8080
- Referrer Policy: strict-origin-when-cross-origin

The response headers include:

- Content-Length: 62686
- Content-Type: image/png
- Date: Wed, 07 Feb 2024 07:02:52 GMT
- Server: Python HTTP Server

The request headers include:

- Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,\*/\*;q=0.8,application/signed-exchange;v=b2;
- Accept-Encoding: gzip, deflate, br
- Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
- Cache-Control: max-age=0
- Connection: keep-alive
- Cookie: \_gcl\_au=1.1.1650434821.1707247323; \_fbp=fb.0.1707247329040.196473239; \_ga=GA1.1.2098121577.1707247323; \_gid=GA1.1.2098121577.1707247323; ga\_EKH5KPRBH=GS1.1.1707285

The response body displays the Santa Clara University logo:

6.The server responds by serving the content of the existing video file video.mp4 from the specified document root directory.

The screenshot shows a browser window with a video player displaying 'video.mp4' at 0:10 / 0:10. Below the video player is a timeline. To the right is the developer tools Network tab. A request for 'video.mp4' is listed with the following details:

- Request URL: http://localhost:8080/video.mp4
- Request Method: GET
- Status Code: 200 OK
- Remote Address: 127.0.0.1:8080
- Referrer Policy: strict-origin-when-cross-origin
- Response Headers:
  - Content-Length: 5162143
  - Content-Type: video/mp4
  - Date: Wed, 07 Feb 2024 07:03:38 GMT
  - Server: Python HTTP Server
- Request Headers:
  - Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,\*/\*;q=0.8,application/signed-exchange;v=b3;q=0.7
  - Accept-Encoding: gzip, deflate, br
  - Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
  - Connection: keep-alive
  - Cookie: \_gcl\_au=1.1.1650434821.1707247325; \_\_fb\_dtsg=0.1707247329040.196473235; \_ga=GA1.1.799313274.1707247329; \_gid=GA1.1.2098121577.1707247329; \_ga\_EKH5KHPRBH=GS1.1.17072853

10 requests | 5.2 MB transferred

7.The server responds by serving the content of the existing pdf file gfs.pdf from the specified document root directory.

The screenshot shows a browser window with a PDF viewer displaying 'gfs.pdf'. The PDF page title is 'The Google File System' and it's authored by Sanjay Ghemawat, Howard Gobioff, and Shun-Tak Leung from Google. Below the PDF viewer is a timeline. To the right is the developer tools Network tab. A request for 'gfs.pdf' is listed with the following details:

- Request URL: http://localhost:8080/gfs.pdf
- Request Method: GET
- Status Code: 200 OK
- Remote Address: 127.0.0.1:8080
- Referrer Policy: strict-origin-when-cross-origin
- Response Headers:
  - Content-Length: 275941
  - Content-Type: application/pdf
  - Date: Wed, 07 Feb 2024 07:04:33 GMT
  - Server: Python HTTP Server
- Request Headers:
  - Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,\*/\*;q=0.8,application/signed-exchange;v=b3;q=0.7
  - Accept-Encoding: gzip, deflate, br
  - Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
  - Connection: keep-alive
  - Cookie: \_gcl\_au=1.1.1650434821.170724732

## Additional feature:

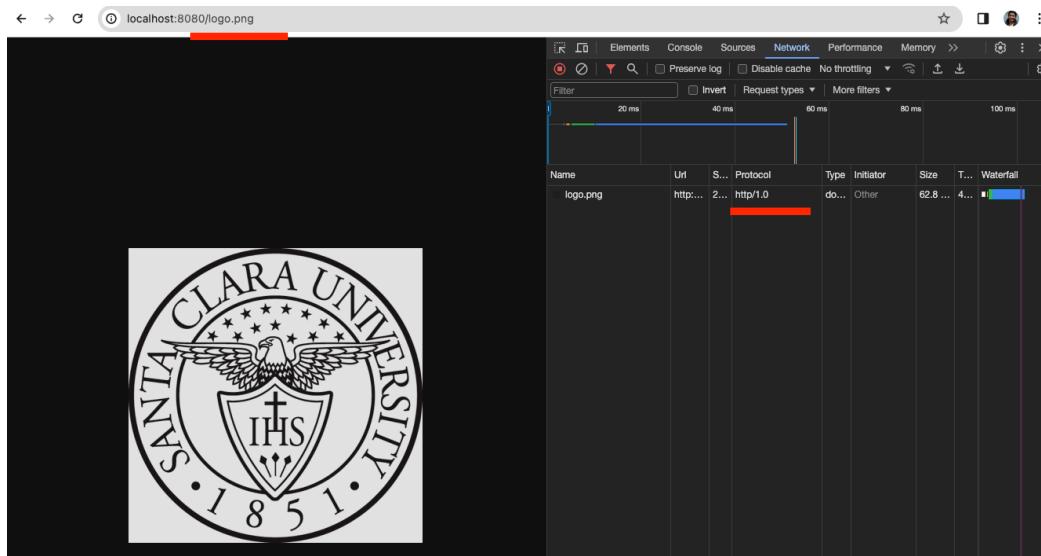
By default, the server.py program takes 1.1 HTTP protocol (persistent connection).

The program supports 1.0 protocol using the optional command line argument listed below:

```
python3 server.py -document_root /Users/shoonya/Desktop/PA1_Pujitha_Kallu/ -port 8080 --protocol 1.0
```

You can observe that the connection is getting closed below:

```
shoonya@Pujithas-MBP PA1_Pujitha_Kallu % python3 server.py --document_root /Users/shoonya/Desktop/PA1_Pujitha_Kallu/ -port 8080 --protocol 1.0
Server started.
Listening for connections on port: 8080
Connection opened: 127.0.0.1
Current thread name is Thread-1 and file opened /Users/shoonya/Desktop/PA1_Pujitha_Kallu/logo.png
Connection closed for HTTP/1.0
Current thread name is Thread-2 and file opened /Users/shoonya/Desktop/PA1_Pujitha_Kallu/logo.png
Connection closed for HTTP/1.0
```



### References:

1. <https://docs.python.org/3/library/mimetypes.html>
2. <https://docs.python.org/3/library/socket.html>
3. <https://docs.python.org/3/library/argparse.html>
4. <https://docs.python.org/3/tutorial/errors.html>
5. Reference from class notes/slides.