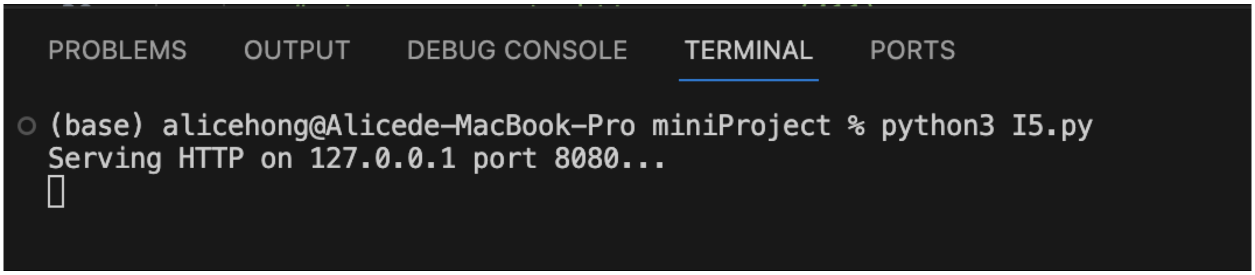
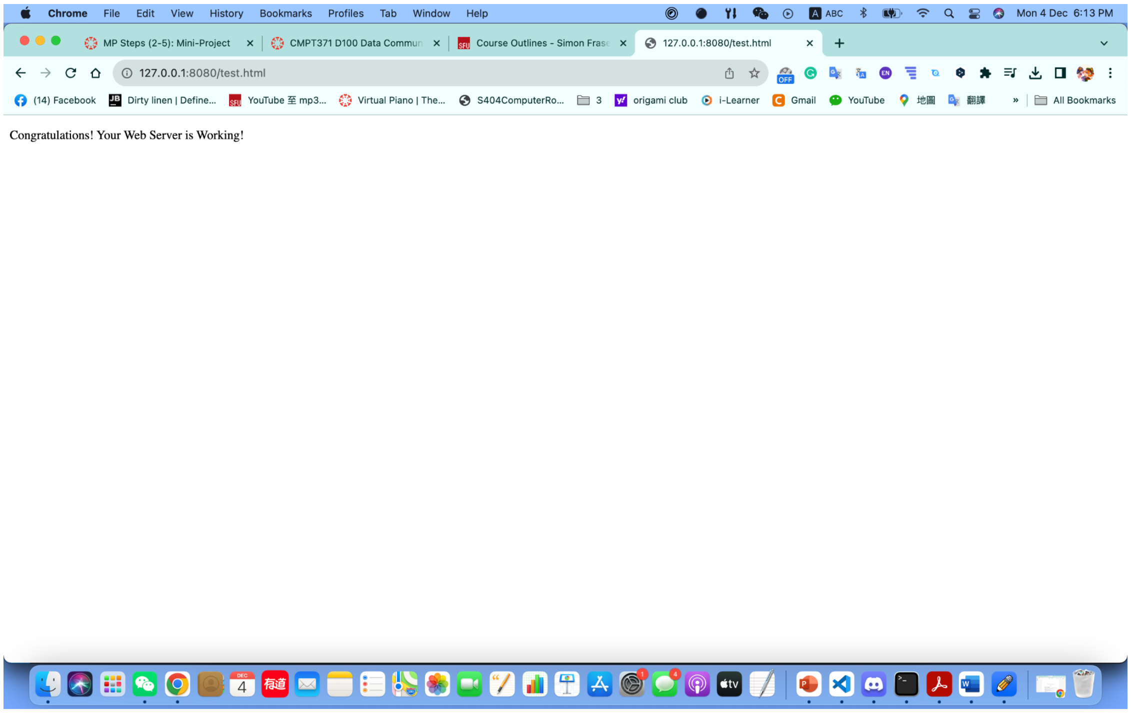
Step two testing procedures

Running the web server:



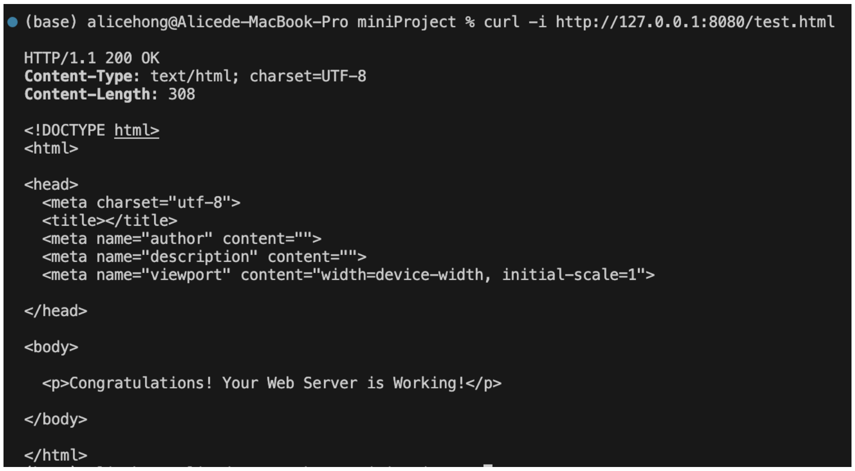
```
○ (base) alicehong@Alicede-MacBook-Pro miniProject % python3 I5.py
  Serving HTTP on 127.0.0.1 port 8080...
```

We used the local host address(127.0.0.1) and 8080 port number for testing:



Congratulations! Your Web Server is Working!

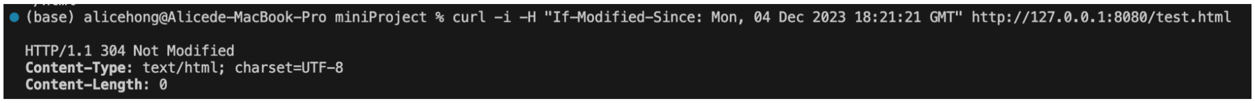
1. Testing status code 200 OK



```
● (base) alicehong@Alicede-MacBook-Pro miniProject % curl -i http://127.0.0.1:8080/test.html

  HTTP/1.1 200 OK
  Content-Type: text/html; charset=UTF-8
  Content-Length: 308

  <!DOCTYPE html>
  <html>

  <head>
    <meta charset="utf-8">
    <title></title>
    <meta name="author" content="">
    <meta name="description" content="">
    <meta name="viewport" content="width=device-width, initial-scale=1">

  </head>

  <body>

    <p>Congratulations! Your Web Server is Working!</p>

  </body>

  </html>
```

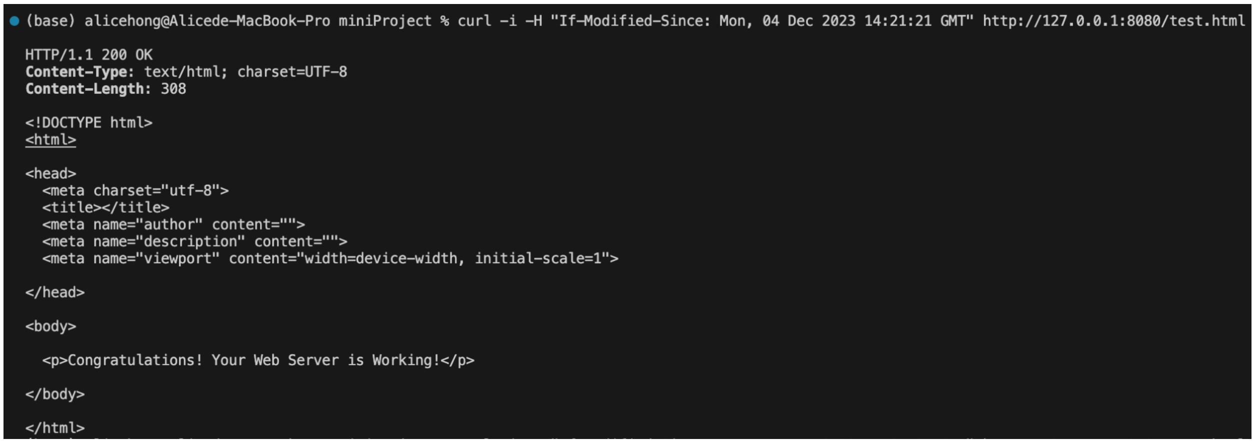
2. Testing status code 304 Not Modified

- Use `curl` to send a request with the If-Modified-Since header
- Input a time that is after the last modified time to test the 304 status code:



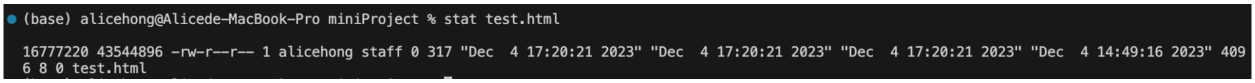
```
(base) alicehong@Alicede-MacBook-Pro miniProject % curl -i -H "If-Modified-Since: Mon, 04 Dec 2023 18:21:21 GMT" http://127.0.0.1:8080/test.html

HTTP/1.1 304 Not Modified
Content-Type: text/html; charset=UTF-8
Content-Length: 0
```

- Input a time that is before the last modified time:



```
(base) alicehong@Alicede-MacBook-Pro miniProject % curl -i -H "If-Modified-Since: Mon, 04 Dec 2023 14:21:21 GMT" http://127.0.0.1:8080/test.html

HTTP/1.1 200 OK
Content-Type: text/html; charset=UTF-8
Content-Length: 308

<!DOCTYPE html>
<html>

<head>
  <meta charset="utf-8">
  <title></title>
  <meta name="author" content="">
  <meta name="description" content="">
  <meta name="viewport" content="width=device-width, initial-scale=1">

</head>

<body>

  <p>Congratulations! Your Web Server is Working!</p>

</body>

</html>
```

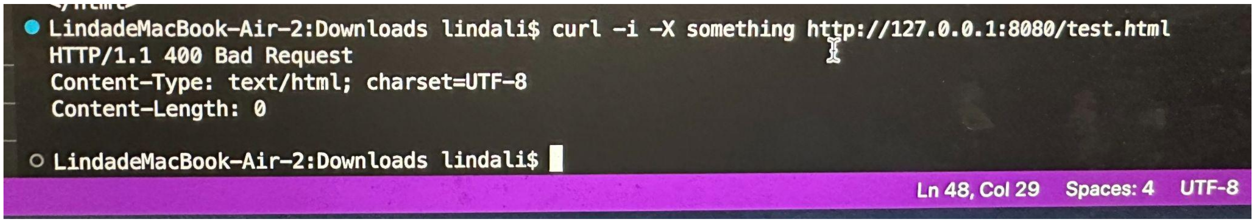
- Using the `stat` command to find the last modified time of the time.html:



```
(base) alicehong@Alicede-MacBook-Pro miniProject % stat test.html

16777220 43544896 -rw-r--r-- 1 alicehong staff 0 317 "Dec  4 17:20:21 2023" "Dec  4 17:20:21 2023" "Dec  4 17:20:21 2023" "Dec  4 14:49:16 2023" 409
6 8 0 test.html
```

3. Testing status code 400 bad request

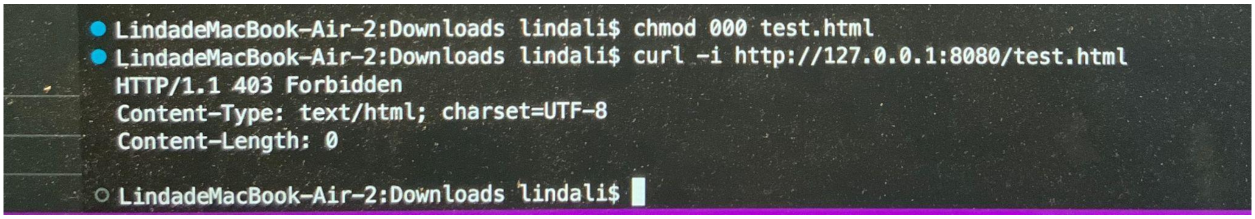
- type in an invalid method called ‘something’



```
LindadeMacBook-Air-2:Downloads lindali$ curl -i -X something http://127.0.0.1:8080/test.html
HTTP/1.1 400 Bad Request
Content-Type: text/html; charset=UTF-8
Content-Length: 0

LindadeMacBook-Air-2:Downloads lindali$
```

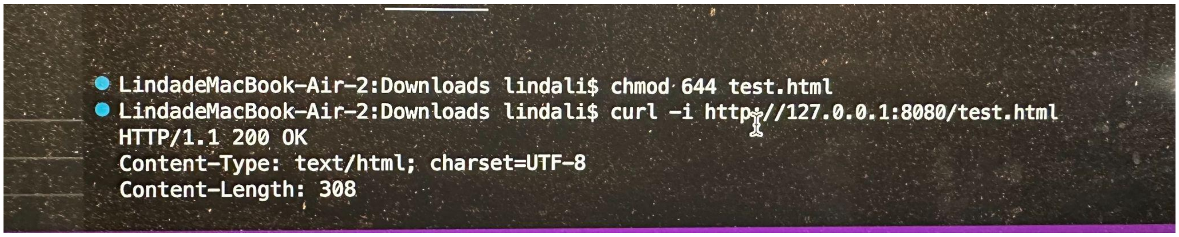
4. Testing status code 403 Forbidden

- restrict file access by using chmod command



```
● LindadeMacBook-Air-2:Downloads lindali$ chmod 000 test.html
● LindadeMacBook-Air-2:Downloads lindali$ curl -i http://127.0.0.1:8080/test.html
  HTTP/1.1 403 Forbidden
  Content-Type: text/html; charset=UTF-8
  Content-Length: 0

○ LindadeMacBook-Air-2:Downloads lindali$
```

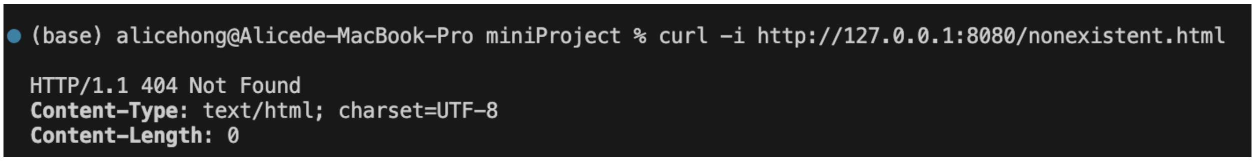
- after allowing access, returns ok



```
● LindadeMacBook-Air-2:Downloads lindali$ chmod 644 test.html
● LindadeMacBook-Air-2:Downloads lindali$ curl -i http://127.0.0.1:8080/test.html
  HTTP/1.1 200 OK
  Content-Type: text/html; charset=UTF-8
  Content-Length: 308
```

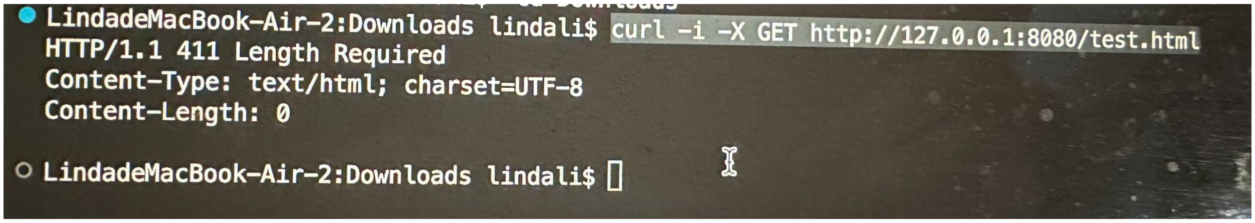
5. Testing status code 404 Not Found

- Use `curl` to request a resource that does not exist (e.g. nonexistent.html)



```
● (base) alicehong@Alicede-MacBook-Pro miniProject % curl -i http://127.0.0.1:8080/nonexistent.html

  HTTP/1.1 404 Not Found
  Content-Type: text/html; charset=UTF-8
  Content-Length: 0
```

6. Testing status code 411 length required



```
● LindadeMacBook-Air-2:Downloads lindali$ curl -i -X GET http://127.0.0.1:8080/test.html
  HTTP/1.1 411 Length Required
  Content-Type: text/html; charset=UTF-8
  Content-Length: 0

○ LindadeMacBook-Air-2:Downloads lindali$
```

Step three - Web proxy server

a)

Differences in Request Handling:

1. Web proxy server:
    - The client communicates with the endpoint server through the proxy.
    - Can cache responses to improve performance and reduce load times.
    - Can enhance security by filtering requests and anonymizing clients.
2. Web Server:
    - Hosts and serves content directly to clients
    - The client knows the endpoint of the server
    - Responds to direct requests for resources like HTML files.

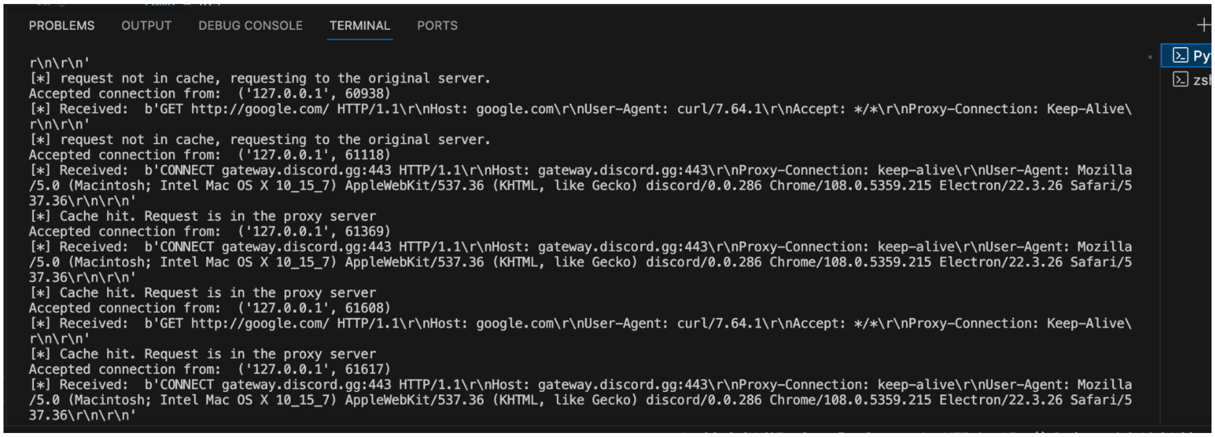
Requirements for the proxy server:

- Accepts the client request
- Check if the client request object in in cache
- If it’s in cache, response the client
- If it’s not in cache, proxy server request the original server
- Then the proxy server responds to the client, and saves the response in cache for next time.

b) Yes, changes are needed at the client side, we need to configure the proxy in our web browser.

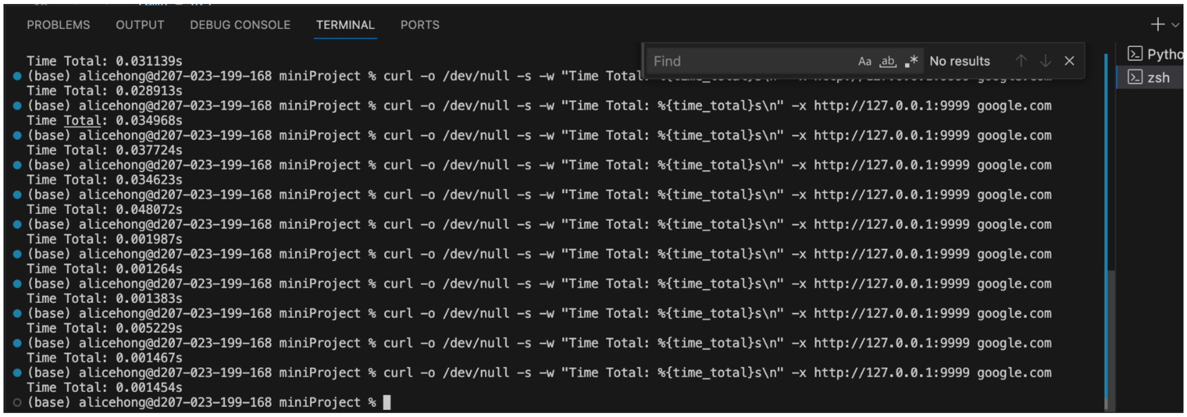
Alternative ways to test the proxy server: test the caching efficiency

- First send the request to the proxy server, if it’s in cache, it will print cache hit, otherwise, proxy server will go access the original server for the request and print not in cache.



```
r\n\r\n'
[*] request not in cache, requesting to the original server.
Accepted connection from:  ('127.0.0.1', 60938)
[*] Received:  b'GET http://google.com/ HTTP/1.1\r\nHost: google.com\r\nUser-Agent: curl/7.64.1\r\nAccept: */*\r\nProxy-Connection: Keep-Alive\
r\n\r\n'
[*] request not in cache, requesting to the original server.
Accepted connection from:  ('127.0.0.1', 61118)
[*] Received:  b'CONNECT gateway.discord.gg:443 HTTP/1.1\r\nHost: gateway.discord.gg:443\r\nProxy-Connection: keep-alive\r\nUser-Agent: Mozilla
/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) discord/0.0.286 Chrome/108.0.5359.215 Electron/22.3.26 Safari/5
37.36\r\n\r\n'
[*] Cache hit. Request is in the proxy server
Accepted connection from:  ('127.0.0.1', 61369)
[*] Received:  b'CONNECT gateway.discord.gg:443 HTTP/1.1\r\nHost: gateway.discord.gg:443\r\nProxy-Connection: keep-alive\r\nUser-Agent: Mozilla
/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) discord/0.0.286 Chrome/108.0.5359.215 Electron/22.3.26 Safari/5
37.36\r\n\r\n'
[*] Cache hit. Request is in the proxy server
Accepted connection from:  ('127.0.0.1', 61608)
[*] Received:  b'GET http://google.com/ HTTP/1.1\r\nHost: google.com\r\nUser-Agent: curl/7.64.1\r\nAccept: */*\r\nProxy-Connection: Keep-Alive\
r\n\r\n'
[*] Cache hit. Request is in the proxy server
Accepted connection from:  ('127.0.0.1', 61617)
[*] Received:  b'CONNECT gateway.discord.gg:443 HTTP/1.1\r\nHost: gateway.discord.gg:443\r\nProxy-Connection: keep-alive\r\nUser-Agent: Mozilla
/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) discord/0.0.286 Chrome/108.0.5359.215 Electron/22.3.26 Safari/5
37.36\r\n\r\n'
```

- After cache hit, the response time for the same client request is reduced significantly (from 0.048072s to 0.001987s). The subsequent requests were also significantly faster since those requests were sent to the same site and the proxy server served these requests directly from the cache.



```
  Time Total: 0.031139s
● (base) alicehong@d207-023-199-168 miniProject % curl -o /dev/null -s -w "Time Total
  Time Total: 0.028913s
● (base) alicehong@d207-023-199-168 miniProject % curl -o /dev/null -s -w "Time Total: %{time_total}s\n" -x http://127.0.0.1:9999 google.com
  Time Total: 0.034968s
● (base) alicehong@d207-023-199-168 miniProject % curl -o /dev/null -s -w "Time Total: %{time_total}s\n" -x http://127.0.0.1:9999 google.com
  Time Total: 0.037724s
● (base) alicehong@d207-023-199-168 miniProject % curl -o /dev/null -s -w "Time Total: %{time_total}s\n" -x http://127.0.0.1:9999 google.com
  Time Total: 0.034623s
● (base) alicehong@d207-023-199-168 miniProject % curl -o /dev/null -s -w "Time Total: %{time_total}s\n" -x http://127.0.0.1:9999 google.com
  Time Total: 0.048072s
● (base) alicehong@d207-023-199-168 miniProject % curl -o /dev/null -s -w "Time Total: %{time_total}s\n" -x http://127.0.0.1:9999 google.com
  Time Total: 0.001987s
● (base) alicehong@d207-023-199-168 miniProject % curl -o /dev/null -s -w "Time Total: %{time_total}s\n" -x http://127.0.0.1:9999 google.com
  Time Total: 0.001264s
● (base) alicehong@d207-023-199-168 miniProject % curl -o /dev/null -s -w "Time Total: %{time_total}s\n" -x http://127.0.0.1:9999 google.com
  Time Total: 0.001383s
● (base) alicehong@d207-023-199-168 miniProject % curl -o /dev/null -s -w "Time Total: %{time_total}s\n" -x http://127.0.0.1:9999 google.com
  Time Total: 0.005229s
● (base) alicehong@d207-023-199-168 miniProject % curl -o /dev/null -s -w "Time Total: %{time_total}s\n" -x http://127.0.0.1:9999 google.com
  Time Total: 0.001467s
● (base) alicehong@d207-023-199-168 miniProject % curl -o /dev/null -s -w "Time Total: %{time_total}s\n" -x http://127.0.0.1:9999 google.com
  Time Total: 0.001454s
○ (base) alicehong@d207-023-199-168 miniProject %
```