Exercise 3:

Question 1: What is the status code and phrase returned from the server to the client browser?

- 200, OK

Question 2: When was the HTML file that the browser is retrieving last modified at the server? Does the response also contain a DATE header? How are these two fields different?

- Last-Modified: Tue, 23 Sep 2003 05:29:00 GMT
- Yes, it does
- Date header field indicates when the message was sent, Last-Modified is the last modification time of the resource on the source server

Question 3: Is the connection established between the browser and the server persistent or non-persistent? How can you infer this?

- is persistent because Connection: Keep-Alive

Question 4: How many bytes of content are being returned to the browser?

- 73 bytes

Question 5: What is the data contained inside the HTTP response packet?

- is the text data of html -- “Congratulations. You've downloaded the file lab2-1.html!\n”

Exercise 4:

Question 1: Inspect the contents of the first HTTP GET request from the browser to the server. Do you see an “IF-MODIFIED-SINCE” line in the HTTP GET?

- Can not

Question 2: Does the response indicate the last time that the requested file was modified?

- No, nothing

Question 3: Now inspect the contents of the second HTTP GET request from the browser to the server. Do you see an “IF-MODIFIED-SINCE:” and “IF-NONE-MATCH” lines in the HTTP GET? If so, what information is contained in these header lines?

- If-Modified-Since: indicates that the server is asked if the resource has been modified since a certain point in time. This point in time is usually the time returned by Last-Modified on the server side.

- If-None-Match: It asks the server if a tag string value does not match, and the tag value is usually the latest string identifier returned by the server's Etag.

Question 4: What is the HTTP status code and phrase returned from the server in response to this second HTTP GET? Did the server explicitly return the contents of the file? Explain.

- 304 Not Modified
- The request resource is the same as the local cache, unmodified. No page content is returned but only an HTTP Header, thus greatly reducing bandwidth consumption

Question 5: What is the value of the ETag field in the 2nd response message and how it is used? Has this value changed since the 1 st response message was received?

- ETag: "1bfef-173-8f4ae900"
- ETag can quickly determine if a resource has changed
- No changes have been made

Exercise 5:

Server:

```
Zeal@Zeals-Laptop MINGW64 ~/Desktop/COMP/COMP3331/Labs/lab2 (master)
$ java PingServer 3333
Received from 127.0.0.1: PING 3331 1664536871.1565301
   Reply sent.
Received from 127.0.0.1: PING 3332 1664536871.2681413
   Reply not sent.
Received from 127.0.0.1: PING 3333 1664536871.8826587
   Reply sent.
Received from 127.0.0.1: PING 3334 1664536872.021657
   Reply sent.
Received from 127.0.0.1: PING 3335 1664536872.1166565
   Reply not sent.
Received from 127.0.0.1: PING 3336 1664536872.7328403
   Reply sent.
Received from 127.0.0.1: PING 3337 1664536872.7725587
   Reply sent.
Received from 127.0.0.1: PING 3338 1664536872.7985697
   Reply sent.
Received from 127.0.0.1: PING 3339 1664536872.942196
   Reply not sent.
Received from 127.0.0.1: PING 3340 1664536873.5441911
   Reply sent.
Received from 127.0.0.1: PING 3341 1664536873.5959303
   Reply not sent.
Received from 127.0.0.1: PING 3342 1664536874.2121906
   Reply not sent.
Received from 127.0.0.1: PING 3343 1664536874.8185382
   Reply sent.
Received from 127.0.0.1: PING 3344 1664536874.9916744
   Reply sent.
Received from 127.0.0.1: PING 3345 1664536875.0795724
   Reply not sent.
```

Client:

```
Zeal@Zeals-Laptop MINGW64 ~/Desktop/COMP/COMP3331/Labs/lab2 (master)
$ py3 -u "c:\Users\Zeal\Desktop\COMP\COMP3331\Labs\lab2\PingClient.py"
ping to 127.0.0.1, seq = 1, rtt = 111.611 ms
ping to 127.0.0.1, seq = 2, time out
ping to 127.0.0.1, seq = 3, rtt = 138.998 ms
ping to 127.0.0.1, seq = 4, rtt = 95.0 ms
ping to 127.0.0.1, seq = 5, time out
ping to 127.0.0.1, seq = 6, rtt = 39.719 ms
ping to 127.0.0.1, seq = 7, rtt = 26.011 ms
ping to 127.0.0.1, seq = 8, rtt = 143.626 ms
ping to 127.0.0.1, seq = 9, time out
ping to 127.0.0.1, seq = 10, rtt = 51.739 ms
ping to 127.0.0.1, seq = 11, time out
ping to 127.0.0.1, seq = 12, time out
ping to 127.0.0.1, seq = 13, rtt = 173.136 ms
ping to 127.0.0.1, seq = 14, rtt = 87.898 ms
ping to 127.0.0.1, seq = 15, time out
min = 26.011 ms
max = 173.136 ms
avg = 96.42 ms
```