Task-1

First part Intro:

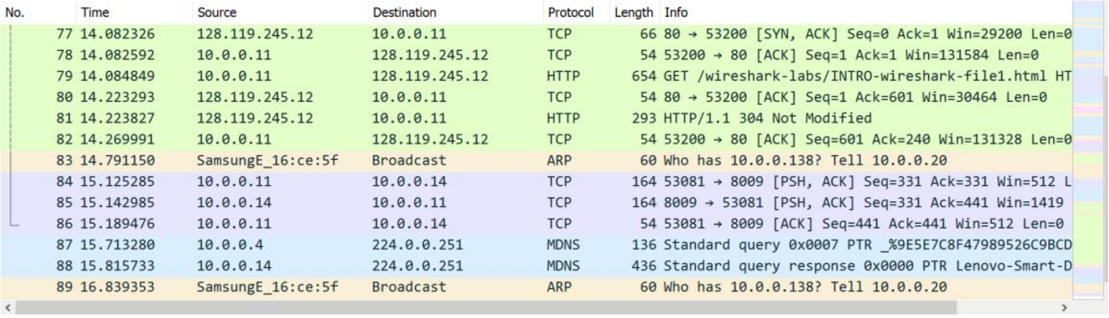
1 . HTTP (because I opened the link).

TCP (one of the most common protocols as we said on class)

ARP(as we can see in the figure the Samsung using it) to read more about this protocol: https://he.wikipedia.org/wiki/Address_Resolution_Protocol

MDNS or multicast DNS (another protocol that we can see in the picture)to read more about this protocol:

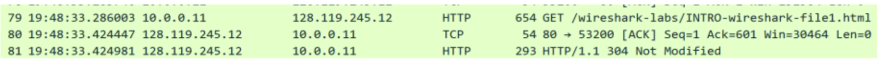
https://en.wikipedia.org/wiki/Multicast_DNS.



No.	Time	Source	Destination	Protocol	Length	Info
77	14.082326	128.119.245.12	10.0.0.11	TCP	66	80 → 53200 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0
78	14.082592	10.0.0.11	128.119.245.12	TCP	54	53200 → 80 [ACK] Seq=1 Ack=1 Win=131584 Len=0
79	14.084849	10.0.0.11	128.119.245.12	HTTP	654	GET /wireshark-labs/INTRO-wireshark-file1.html HT
80	14.223293	128.119.245.12	10.0.0.11	TCP	54	80 → 53200 [ACK] Seq=1 Ack=601 Win=30464 Len=0
81	14.223827	128.119.245.12	10.0.0.11	HTTP	293	HTTP/1.1 304 Not Modified
82	14.269991	10.0.0.11	128.119.245.12	TCP	54	53200 → 80 [ACK] Seq=601 Ack=240 Win=131328 Len=0
83	14.791150	SamsungE_16:ce:5f	Broadcast	ARP	60	Who has 10.0.0.138? Tell 10.0.0.20
84	15.125285	10.0.0.11	10.0.0.14	TCP	164	53081 → 8009 [PSH, ACK] Seq=331 Ack=331 Win=512 L
85	15.142985	10.0.0.14	10.0.0.11	TCP	164	8009 → 53081 [PSH, ACK] Seq=331 Ack=441 Win=1419
86	15.189476	10.0.0.11	10.0.0.14	TCP	54	53081 → 8009 [ACK] Seq=441 Ack=441 Win=512 Len=0
87	15.713280	10.0.0.4	224.0.0.251	MDNS	136	Standard query 0x0007 PTR _%9E5E7C8F47989526C9BCD
88	15.815733	10.0.0.14	224.0.0.251	MDNS	436	Standard query response 0x0000 PTR Lenovo-Smart-D
89	16.839353	SamsungE_16:ce:5f	Broadcast	ARP	60	Who has 10.0.0.138? Tell 10.0.0.20

Figure-1

2.When we are using Time of day as we can see in figure two we can see that the replay was received at the same second (the second http message). But from the firs figure we can subtract and understand that it took 0.138978 which means a little beat more than the second tenth.

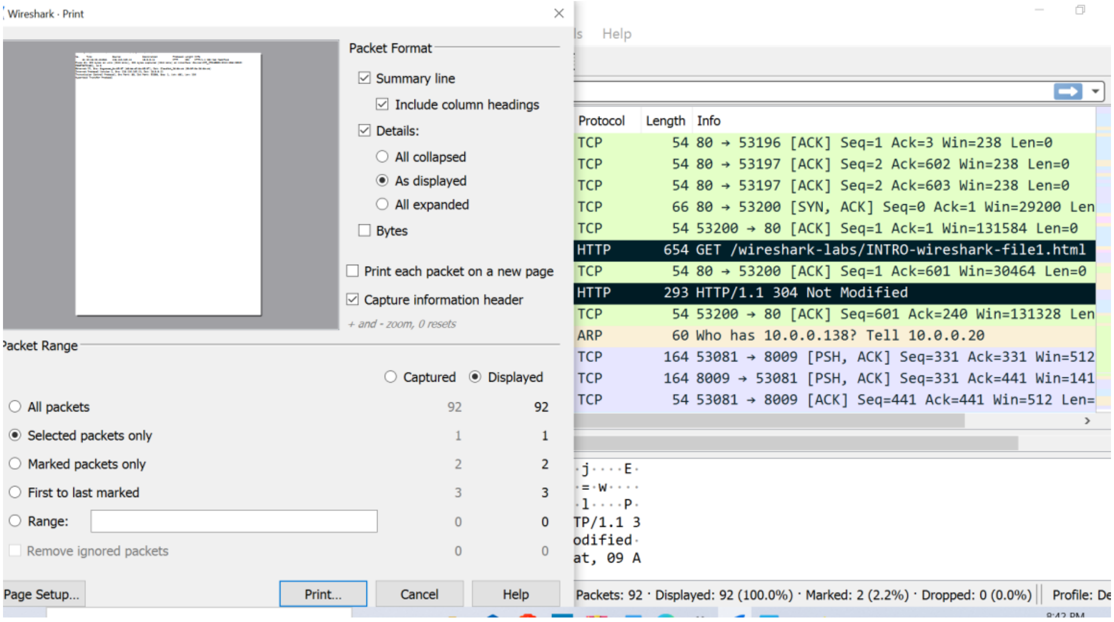


79	19:48:33.286003	10.0.0.11	128.119.245.12	HTTP	654	GET /wireshark-labs/INTRO-wireshark-file1.html
80	19:48:33.424447	128.119.245.12	10.0.0.11	TCP	54	80 → 53200 [ACK] Seq=1 Ack=601 Win=30464 Len=0
81	19:48:33.424981	128.119.245.12	10.0.0.11	HTTP	293	HTTP/1.1 304 Not Modified

Figure-2

3. gaia.cs.unmass.edu internet address is: 128.119.245.12

My internet address is: 10.0.0.11



Wireshark · Print

Packet Format

☑ Summary line

☑ Include column headings

☑ Details:

○ All collapsed

● As displayed

○ All expanded

□ Bytes

□ Print each packet on a new page

☑ Capture information header

+ and - zoom, 0 resets

Packet Range

	○ Captured	● Displayed
○ All packets	92	92
● Selected packets only	1	1
○ Marked packets only	2	2
○ First to last marked	3	3
○ Range:	0	0
□ Remove ignored packets	0	0

```
Protocol  Length Info
TCP           54 80 → 53196 [ACK] Seq=1 Ack=3 Win=238 Len=0
TCP           54 80 → 53197 [ACK] Seq=2 Ack=602 Win=238 Len=0
TCP           54 80 → 53197 [ACK] Seq=2 Ack=603 Win=238 Len=0
TCP           66 80 → 53200 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len
TCP           54 53200 → 80 [ACK] Seq=1 Ack=1 Win=131584 Len=0
HTTP         654 GET /wireshark-labs/INTRO-wireshark-file1.html
TCP           54 80 → 53200 [ACK] Seq=1 Ack=601 Win=30464 Len=0
HTTP         293 HTTP/1.1 304 Not Modified
TCP           54 53200 → 80 [ACK] Seq=601 Ack=240 Win=131328 Len
ARP           60 Who has 10.0.0.138? Tell 10.0.0.20
TCP          164 53081 → 8009 [PSH, ACK] Seq=331 Ack=331 Win=512
TCP          164 8009 → 53081 [PSH, ACK] Seq=331 Ack=441 Win=141
TCP           54 53081 → 8009 [ACK] Seq=441 Ack=441 Win=512 Len=
```

```
·j····E·
·=·W····
·l····P·
TP/1.1 3
odified·
at, 09 A
```

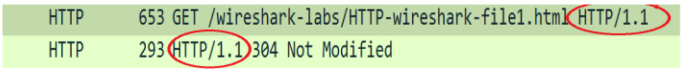
Packets: 92 · Displayed: 92 (100.0%) · Marked: 2 (2.2%) · Dropped: 0 (0.0%) | Profile: De

Figure -3

4. As we can see in figure-3 above just selected the two http massages as requested using mark with the right button I also marked selected packets only as requested and print as displayed the only thing is left is to print it.

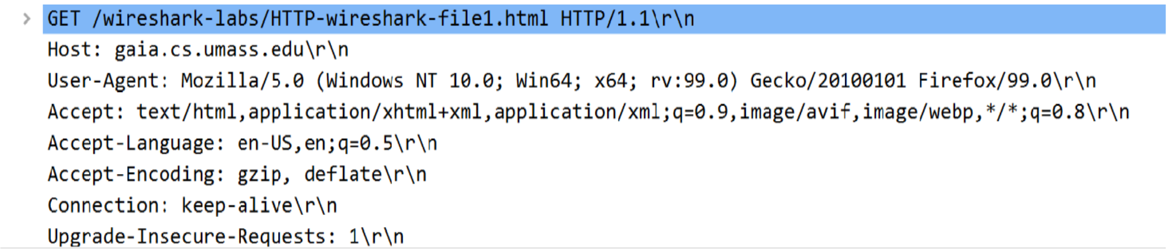
Second part HTTP:

1.Both are running 1.1 version as we can see in the following picture marked in the red circles.



```
HTTP      653 GET /wireshark-labs/HTTP-wireshark-file1.html HTTP/1.1
HTTP      293 HTTP/1.1 304 Not Modified
```

2. Accept language appears in the get massage and the language is: en-US , as we can see in the following picture.



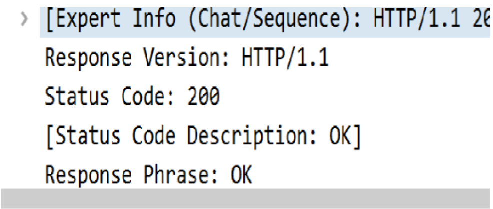
```
> GET /wireshark-labs/HTTP-wireshark-file1.html HTTP/1.1\r\n
  Host: gaia.cs.umass.edu\r\n
  User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:99.0) Gecko/20100101 Firefox/99.0\r\n
  Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8\r\n
  Accept-Language: en-US,en;q=0.5\r\n
  Accept-Encoding: gzip, deflate\r\n
  Connection: keep-alive\r\n
  Upgrade-Insecure-Requests: 1\r\n
```

3. gaia.cs.unmass.edu internet address is: 128.119.245.12

My internet address is: 10.0.0.11

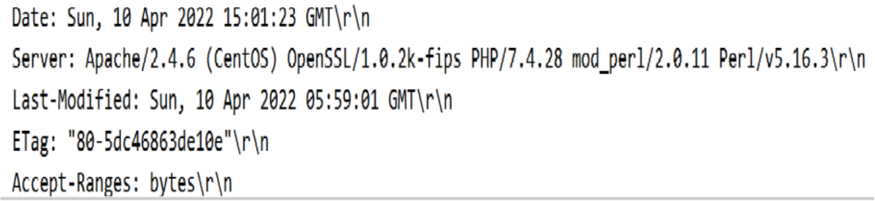
As we can see in figure 2 on the first part.

4.The status code located in the OK massage from the HTTP and the code number is 200 as we can see in the following picture.



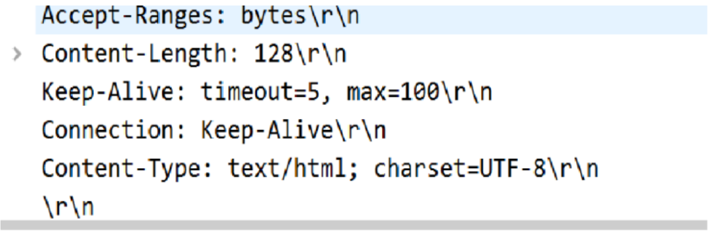
```
› [Expert Info (Chat/Sequence): HTTP/1.1 20
  Response Version: HTTP/1.1
  Status Code: 200
  [Status Code Description: OK]
  Response Phrase: OK
```

5.It says that it is last modified at 10,4,2022 5:59:01 as
we can see in the following picture it is appears in the OK massage.



```
Date: Sun, 10 Apr 2022 15:01:23 GMT\r\n
Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.2k-fips PHP/7.4.28 mod_perl/2.0.11 Perl/v5.16.3\r\n
Last-Modified: Sun, 10 Apr 2022 05:59:01 GMT\r\n
ETag: "80-5dc46863de10e"\r\n
Accept-Ranges: bytes\r\n
```

6. The number of bytes that are returned located in the ok massage(the content length) as we can see the answer is 128 bytes.

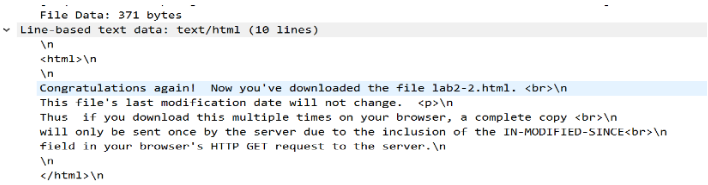


```
  Accept-Ranges: bytes\r\n
› Content-Length: 128\r\n
  Keep-Alive: timeout=5, max=100\r\n
  Connection: Keep-Alive\r\n
  Content-Type: text/html; charset=UTF-8\r\n
  \r\n
```

7. no I don’t see any in the HTTP massage below.

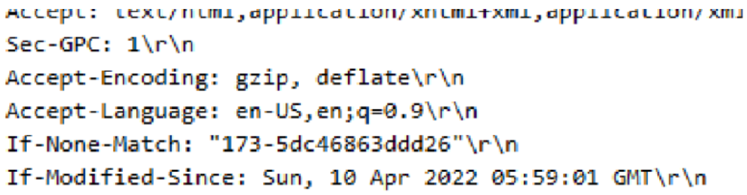
8.NO, there is not “IF-MODIFIED-SINCE” while I checked the first GET.

9.Yes, we can see it explicitly in the following picture, I found it while entering the line based text which tells you how many rows do you have there.



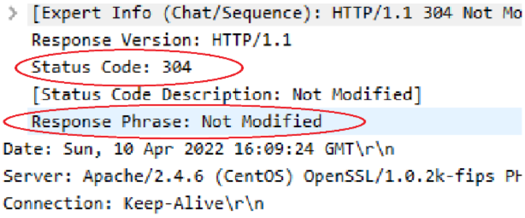
```
    File Data: 371 bytes
⌄ Line-based text data: text/html (10 lines)
    \n
    <html>\n
    \n
    Congratulations again!  Now you've downloaded the file lab2-2.html. <br>\n
    This file's last modification date will not change.  <p>\n
    Thus  if you download this multiple times on your browser, a complete copy <br>\n
    will only be sent once by the server due to the inclusion of the IN-MODIFIED-SINCE<br>\n
    field in your browser's HTTP GET request to the server.\n
    \n
    </html>\n
```

10.Yes, in the second GET there is the "IF-MODIFIED-SINCE" as we can see in the following picture, it is easy to understand because this is the second time before capturing the same URL so the wire shark gives this option.



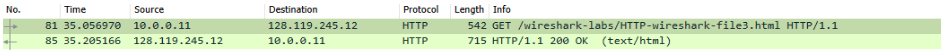
```
Accept: text/html,application/xhtml+xml,application/xml
Sec-GPC: 1\r\n
Accept-Encoding: gzip, deflate\r\n
Accept-Language: en-US,en;q=0.9\r\n
If-None-Match: "173-5dc46863ddd26"\r\n
If-Modified-Since: Sun, 10 Apr 2022 05:59:01 GMT\r\n
```

11.The status cod that returned in the second
one is 304 which says that the file did not modified as we can see in the following picture. The phrase is not modified , there for the massage didn't return the file content because the file didn't modified, the content was returned in the first massage at the second message the file didn't modified and that is what the massage returned.



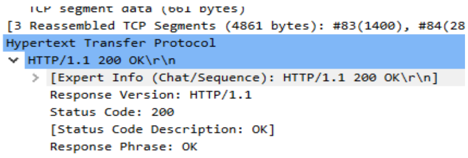
```
  > [Expert Info (Chat/Sequence): HTTP/1.1 304 Not Mo
    Response Version: HTTP/1.1
    Status Code: 304
    [Status Code Description: Not Modified]
    Response Phrase: Not Modified
Date: Sun, 10 Apr 2022 16:09:24 GMT\r\n
Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.2k-fips PH
Connection: Keep-Alive\r\n
```

12.Just 1, as we can see in the following picture , the number for the bill of rights massage is 81.



```
No.      Time       Source          Destination     Protocol  Length Info
     81 35.056970  10.0.0.11       128.119.245.12  HTTP        542 GET /wireshark-labs/HTTP-wireshark-file3.html HTTP/1.1
     85 35.205166  128.119.245.12  10.0.0.11       HTTP        715 HTTP/1.1 200 OK  (text/html)
```

13.Number 83 as we can see the code and the phrase are
associated with this packet number and inside the 1400 bytes.



```
    TCP segment data (661 bytes)
[3 Reassembled TCP Segments (4861 bytes): #83(1400), #84(28
Hypertext Transfer Protocol
 v HTTP/1.1 200 OK\r\n
   > [Expert Info (Chat/Sequence): HTTP/1.1 200 OK\r\n]
     Response Version: HTTP/1.1
     Status Code: 200
     [Status Code Description: OK]
     Response Phrase: OK
```

14.As we can see from the picture above the code is 200 and the phrase is ok.

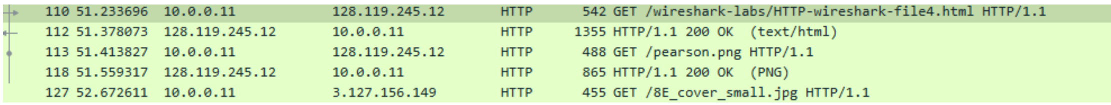
15. 3 packets as we can see also from the picture in question 13 the bytes that was requires are 4861 and not 4500 as we said on class if you divide it to massage every massage is taking a little bit more to the headers etc. you can't divide it for exactly 3 parts, and in the picture, we can see that the packets are 83,84,85.

16. My browser sent 3 http GET message requests. The first two to a same Ip the three are: The initial page, Pearson.png, and cover.jpg.

The initial page to 128.119.245.12

Person.png to 128.119.245.12

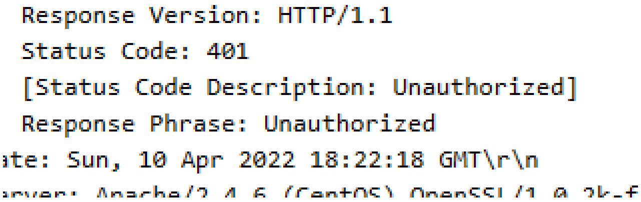
Cover.jpg to 3.127.156.149



```
110 51.233696  10.0.0.11       128.119.245.12   HTTP    542 GET /wireshark-labs/HTTP-wireshark-file4.html HTTP/1.1
112 51.378073  128.119.245.12  10.0.0.11        HTTP   1355 HTTP/1.1 200 OK  (text/html)
113 51.413827  10.0.0.11       128.119.245.12   HTTP    488 GET /pearson.png HTTP/1.1
118 51.559317  128.119.245.12  10.0.0.11        HTTP    865 HTTP/1.1 200 OK  (PNG)
127 52.672611  10.0.0.11       3.127.156.149    HTTP    455 GET /8E_cover_small.jpg HTTP/1.1
```

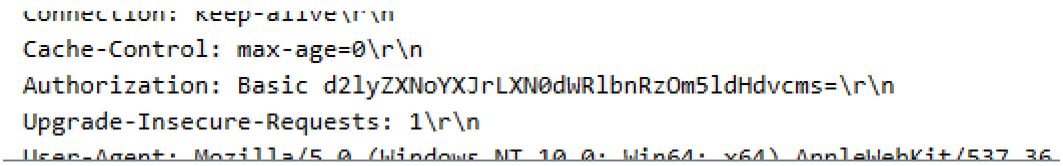
17. I think that it was serially. As you can see in the picture above the first picture was requested and sent back before the second was even requested. If it was in parallel, they were been requested together and sent back together approximately which is not the case, there for I believe that it was serially.

18.The code is 401 and the phrase is authorized as we can see in the following picture (it says that I put the right code =)).



```
 Response Version: HTTP/1.1
 Status Code: 401
 [Status Code Description: Unauthorized]
 Response Phrase: Unauthorized
ate: Sun, 10 Apr 2022 18:22:18 GMT\r\n
erver: Apache/2.4.6 (CentOS) OpenSSL/1.0.2k-f
```

19.The new field is the Authorization basic it is encoded in a format known as Base64 format (the explanation is under the question).



```
 Connection: keep-alive\r\n
 Cache-Control: max-age=0\r\n
 Authorization: Basic d2lyZXNoYXJrLXN0dWRlbnRzOm5ldHdvcms=\r\n
 Upgrade-Insecure-Requests: 1\r\n
 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
```