

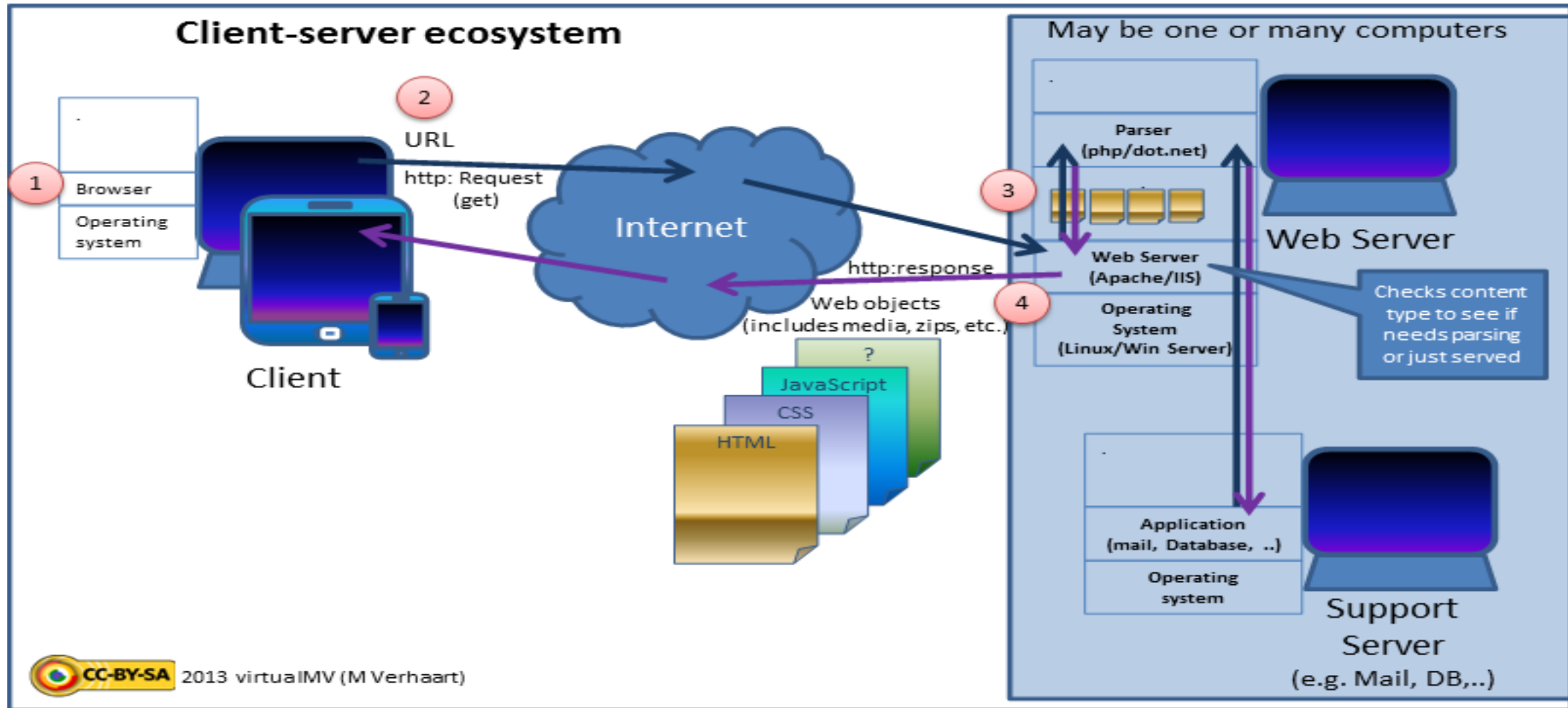
PROJECT TRAINING WORKSHOP

Web API Fundamentals

UNDERSTANDING CLIENT/SERVER



UNDERSTANDING CLIENT/SERVER



UNDERSTANDING OSI Model

Layer 7: Application

Preparing requests
Protocols: HTTP, HTTPS, DNS
Data: Requests

Layer 5: Session

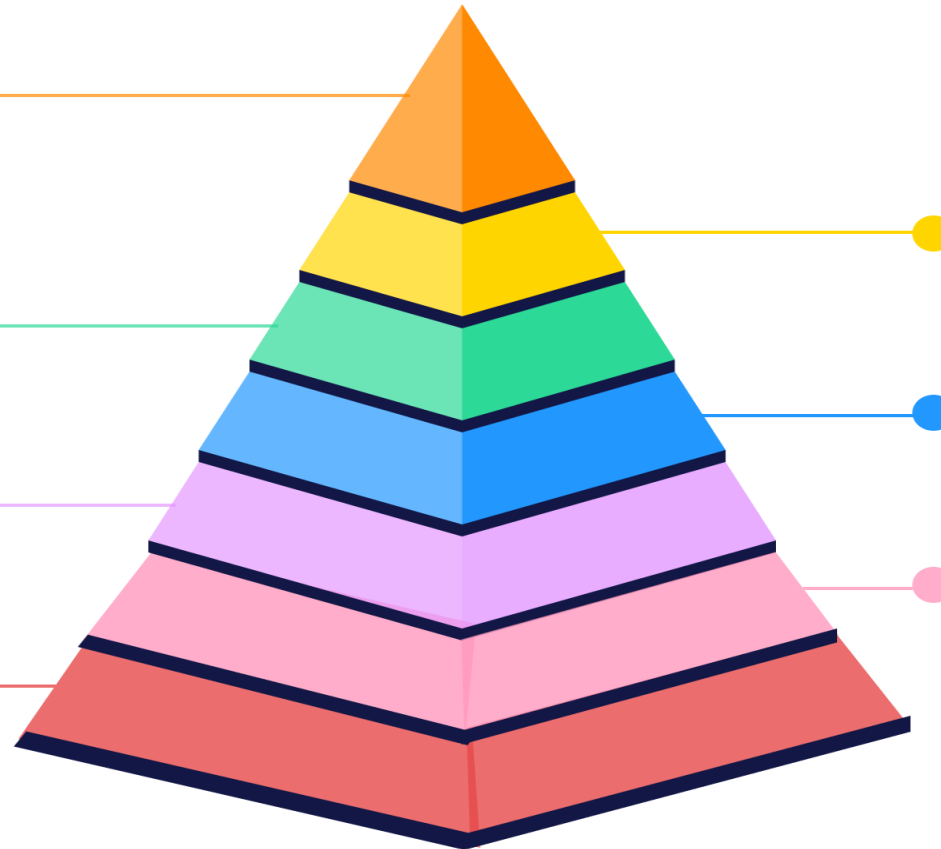
Creating communication channels
(sessions) between devices
Protocols: RPC
Data: Requests

Layer 3: Network

Breaks segments into network packets
and finds best route for sending packets.
Protocols: IP, NAT, ARP
Data: Packets

Layer 1: Physical

Converts data into physical signals
Data: Optics (Fibre), Electricity



Layer 6: Presentation

Data format
eg. encryption and compression
Protocols: TLS, SSL
Data: Requests

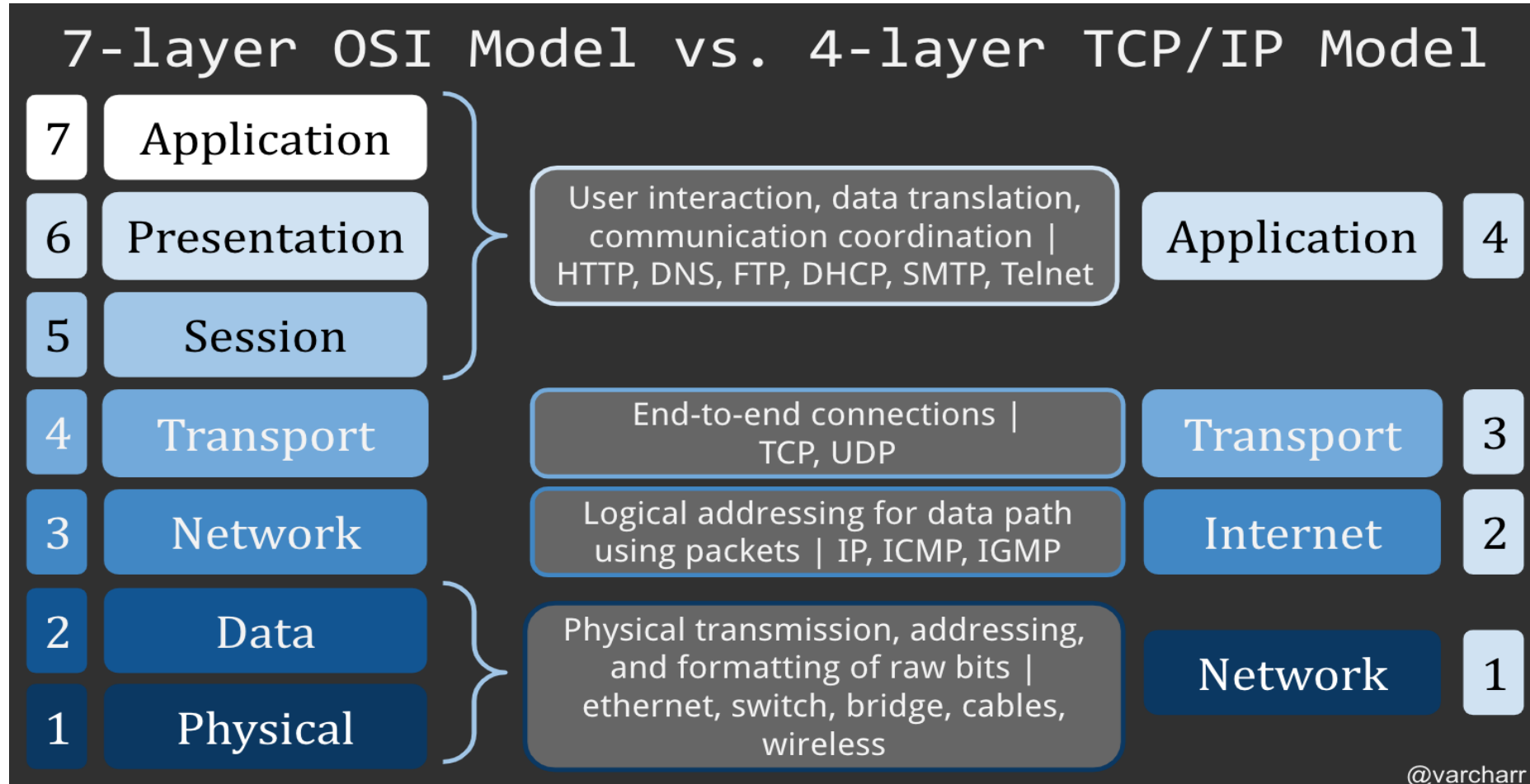
Layer 4: Transport

Breaks requests down into pieces that
can be sent over the network and
handles the sending of those
Protocols: TCP, UDP
Data: Segments

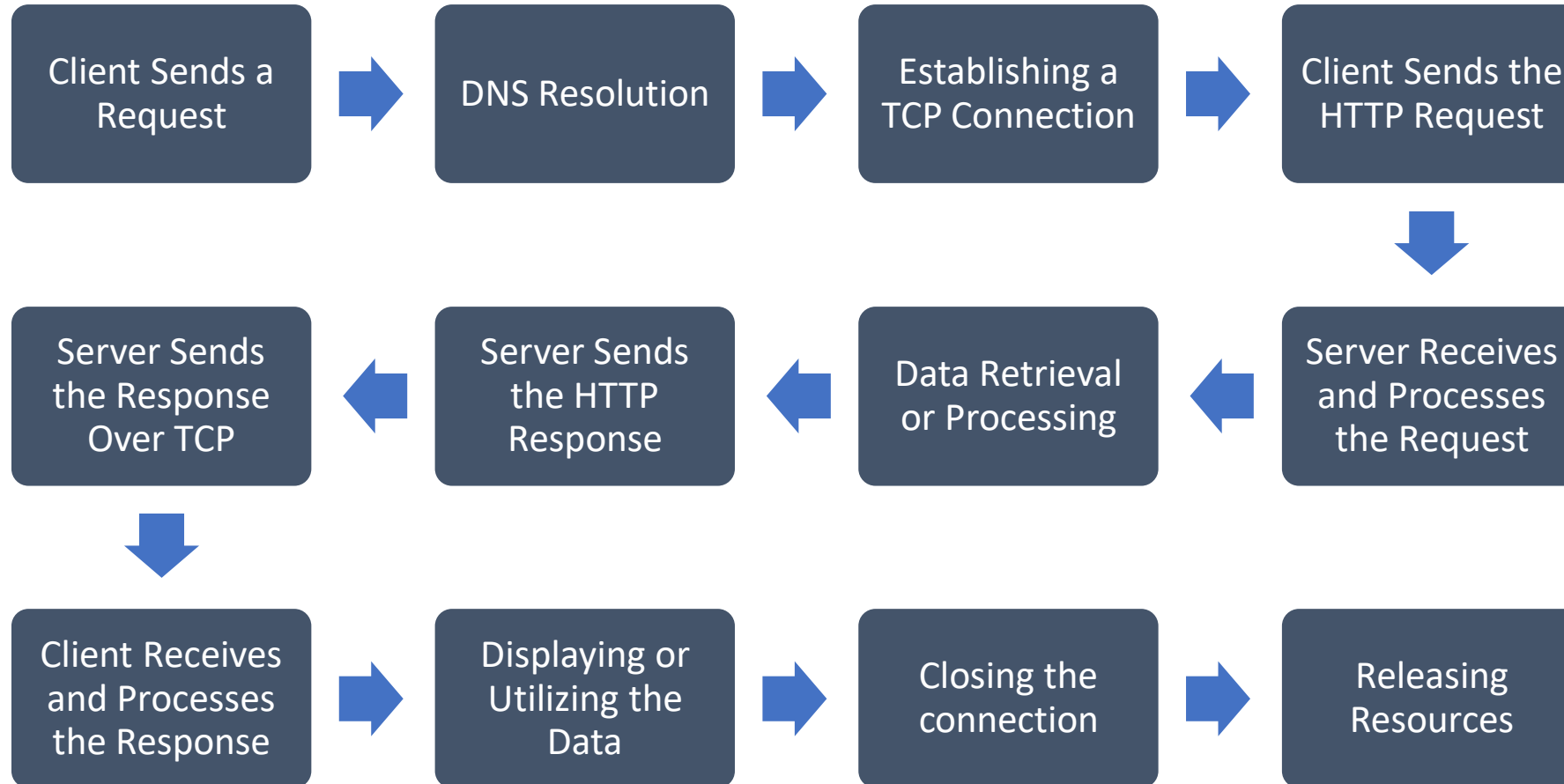
Layer 2: Data Link

Establishes connections between
physical machines using MAC addresses
Breaks segments packets into frames
and sends from source to destination
Protocols: Wifi, Ethernet
Data: Frames

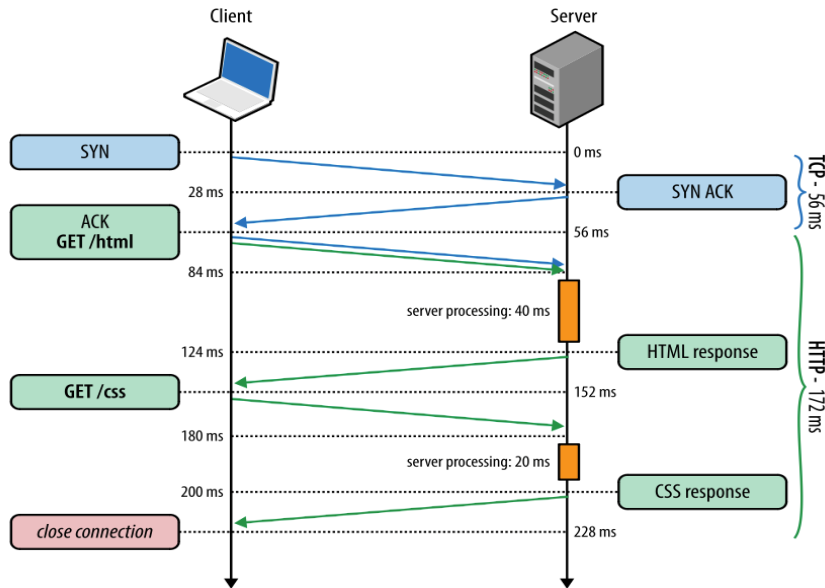
UNDERSTANDING TCP/IP Model



UNDERSTANDING REQUEST/RESPONSE



UNDERSTANDING REQUEST/RESPONSE



(1) User issues URL from a browser
http://host:port/path/file



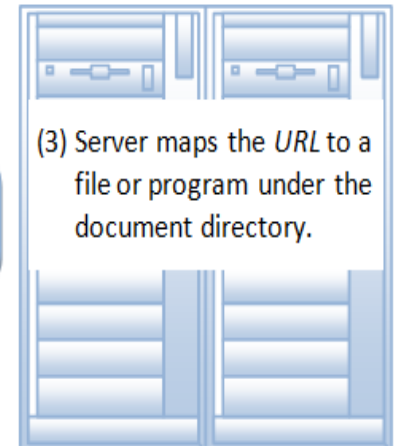
Client (Browser)

(2) Browser sends a request message

```
GET URL HTTP/1.1
Host: host:port
.....
```

(4) Server returns a response message

```
HTTP/1.1 200 OK
.....
```



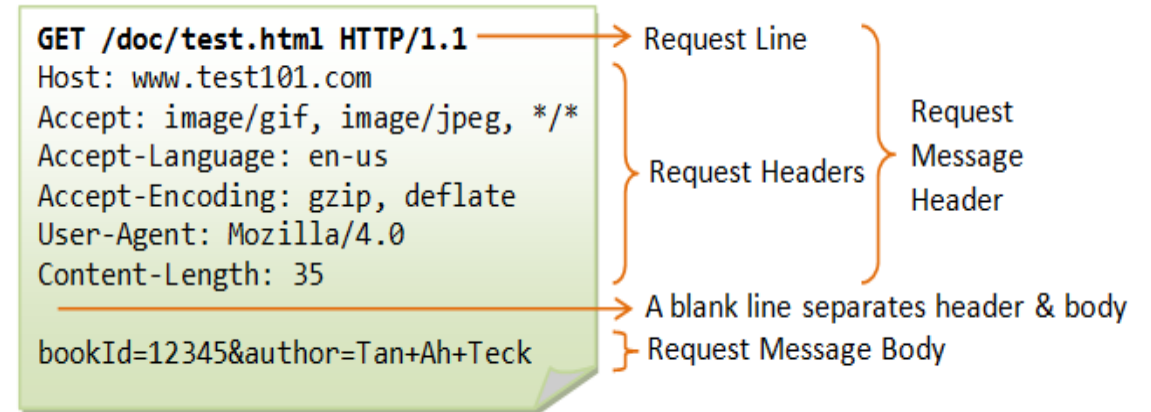
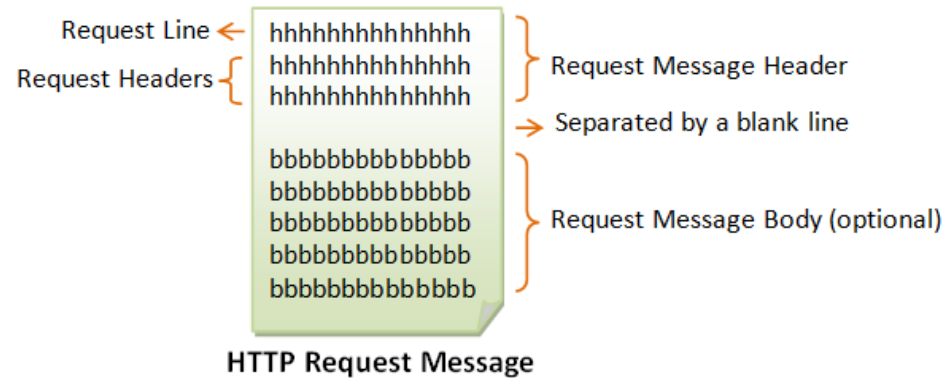
(3) Server maps the *URL* to a file or program under the document directory.

Server (@ host:port)

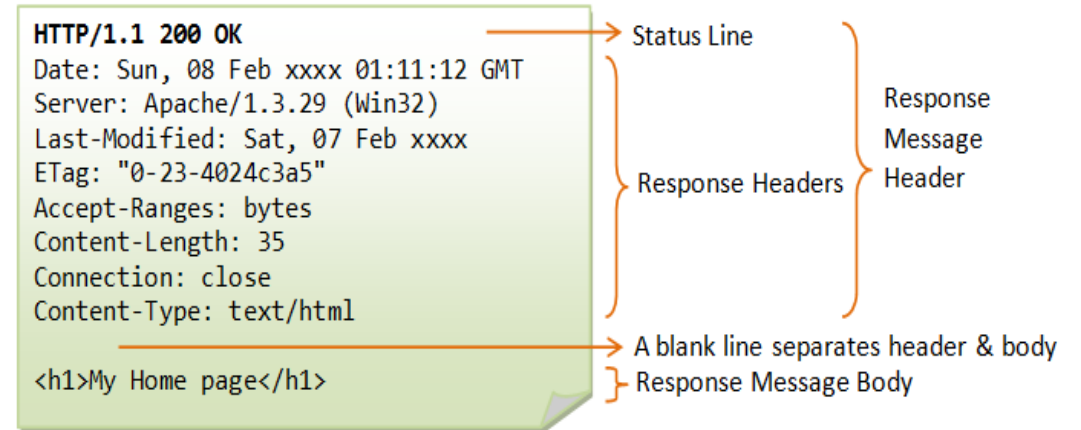
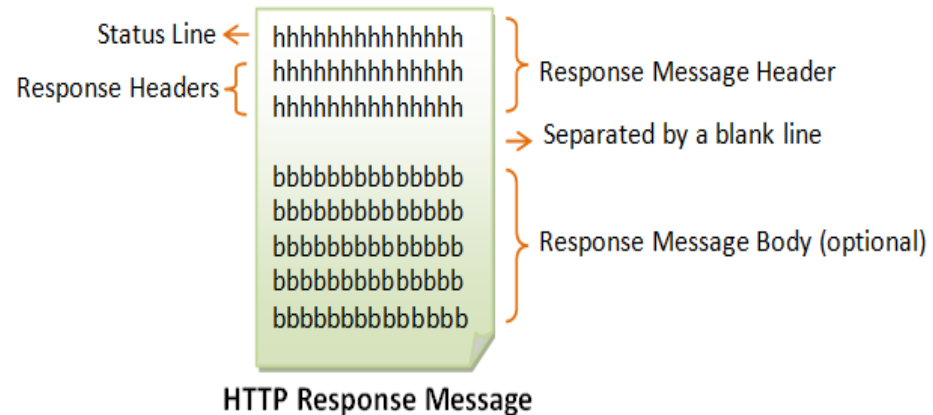
(5) Browser formats the response and displays

HTTP (Over TCP/IP)

UNDERSTANDING REQUEST



UNDERSTANDING RESPONSE



HTTP METHODS

GET	HEAD	POST	PUT
DELETE	TRACE	OPTIONS	CONNECT
PATCH			

/books

GET	/books	Lists all the books in the database
DELETE	/books/{bookId}	Deletes a book based on their id
POST	/books	Creates a Book
PUT	/books/{bookId}	Method to update a book
GET	/books/{bookId}	Retrieves a book based on their id

HTTP HEADERS

GENERAL	REQUEST	RESPONSE	ENTITY
---------	---------	----------	--------

```
general-header = Cache-Control      ;
                  | Connection      ;
                  | Date             ;
                  | Pragma           ;
                  | Trailer          ;
                  | Transfer-Encoding ;
                  | Upgrade          ;
                  | Via              ;
                  | Warning          ;
```

```
request-header = Accept             ;
                  | Accept-Charset   ;
                  | Accept-Encoding   ;
                  | Accept-Language   ;
                  | Authorization     ;
                  | Expect            ;
                  | From              ;
                  | Host              ;
                  | If-Match          ;
                  | If-Modified-Since ;
                  | If-None-Match     ;
                  | If-Range          ;
                  | If-Unmodified-Since ;
                  | Max-Forwards      ;
                  | Proxy-Authorization ;
                  | Range             ;
                  | Referer           ;
                  | TE                ;
                  | User-Agent        ;
```

```
response-header = Accept-Ranges     ;
                  | Age              ;
                  | ETag             ;
                  | Location          ;
                  | Proxy-Authenticate ;
                  | Retry-After       ;
                  | Server            ;
                  | Vary              ;
                  | WWW-Authenticate ;
```

```
entity-header = Allow                ;
                  | Content-Encoding   ;
                  | Content-Language   ;
                  | Content-Length     ;
                  | Content-Location   ;
                  | Content-MD5        ;
                  | Content-Range      ;
                  | Content-Type       ;
                  | Expires            ;
                  | Last-Modified      ;
                  | extension-header
```

```
extension-header = message-header
```

HTTP STATUS CODES

<https://developer.mozilla.org/en-US/docs/Web/HTTP/Status>

1XX - Informational

- It means the request has been received and the process is continuing.
- 100-199

2XX - Successful

- It denotes the client's request was successfully received, understood, and accepted.
- 200-299

3XX - Redirection

- It means further action need to taken to complete the request.
- 300-399

4XX - Client Error

- Invalid request due to incorrect syntax or this request cannot be fulfilled.
- 400-499

5XX - Server Error

- Server has failed to fulfill a valid request
- 500-599