WWW – information system on the Internet that allows documents to be connected to other documents by hyperlinks, enabling the user to search for more information by moving from one document to another.

Internet – refers to the global information system that is logically linked together by a globally address space based on the Internet Protocol.

1969 – internet started

1972 – email

1989 – www started by Sir Tim Berners-Lee

WAIS (Wide Area Information Servers) – specialize data bases

Gopher – is a TCP/IP application layer protocol designed for distributing, searching and retrieving internet in the documents

Usenet – worldwide distributed discussion system available on computers

URL (Uniform Resource Locator) – a way to address a particular document

HTTP (Hypertext Transfer Protocol) – application layer communications protocol used to access resources in the WWW

Versiom History:

* HTTP 0.9 (1991)
* HTTP 1.0 (RFC 1945, May 1996)
* HTTP 1.1 (RFC 2058 Jan 1997, RFC 2626 Jun 1999), RFC 7230-7235 (June 2014)
* HTTP 2 (RFC May 2015)

HTTP FUNDAMENTALS

* HTTP runs on top of TCP/IP, using TCP port 80 by default or TCP port 443 for HTTP’s (HTTP over SSL/TLS)
* HTTP is a stateless communications protocol
* HTTP provides support for other functionalities such as
* Cache control
* Content media type (MIME)
* Language and character set specification
* Content/transfer codings
* Content negotiation
* Client-server protocol negotiation
* Consistent connections
* Request pipelining
* Authentication/authorization

HTTP Resource Addressing

* HTTP resources are identified using URL’s, or more specifically, HTTP URL’s
* Scheme
* Authority
* User information or authentication credentials
* Host – domain name of the server where the resource resides
* Port number
* Patch to resource (resolved relative to the document root on the server)
* Query – typically provided as key/value pairs with ampersand ($) separators between key/value pairs
* Fragment identifier
* Request line
* HTTP Response Message
* Status line
* Empty line(RLF)
* Message Body(0ptianal)

STANDARD METHODS

* Get Method – transfer a current related presentation of the resource identifier by the Request URI, the retrieved resource is returned in the message body of the response as entity
* Head Method –same as get except that the entity is not included in the response (i.e. returns only the status line and headers returned by a GET request w/o the message body. And it used to retrieve the METADATA about the entity implied by the request
* Post Method – perform resource specific processing of the entities enclosed
* Put Method – store the enclosed entity in the message body under the specific request URI (i.e. the resource identified by the request). Submit a data and expect a resource to be created under that URL. Unsafe but idempotent
* Delete Method – remove the resource associated with the request URI
* Options Method - request information
* Trace Method – request a loop-back of the request message (i.e. request the server to echo back to the client the received request message)
* Connect Method – request the establishment of a tunnel to the destination origin

GENERAL HEADER FIELDS

* cache control
* connection
* date
* pragma
* trailer
* transfer
* upgrade
* via
* warning

REQUEST HEADER FIELDS

* accept
* accept – encoding
* accept – language
* accept – charset
* authorization
* proxy authenticate
* expect
* host
* if – match
* if - none – match
* if - modified - since
* if – unmodified – since
* if – range
* max – forwards
* range
* referer
* user – agents
* location

ENTITY HEADER FIELDS

* allow
* content – encoding
* content – language
* content – length
* content – location
* content – MD5
* content – range
* content – type
* expires
* last – modified

HTTP STATUS CODE

* Informal
* 100 continue
* 101 switching protocol
* Success
* 200 OK
* 201 created
* 202 accept
* 203 non – authoritative information
* 204 no content
* 205 reset content
* 206 partial content
* Redirect
* 300 multiple choices
* 301 moved permanently
* 302 found
* 303 see other
* 304 not modified
* 305 used proxy
* 306 unused
* 307 temporary redirect
* Client Error
* 400 bad request
* 401 unauthorized
* 402 payment required
* 403 forbidden
* 404 not found
* 405 method not allowed
* 406 not acceptable
* 407 proxy authentication
* 408 request time – out
* 409 conflict
* 410 gone
* 411 length required
* 412 precondition failed
* 413 request entity too large
* 414 request URI too large
* 415 unsupported media type
* 416 request range not satisfiable
* 417 expectation failed
* 426 upgrade required
* Server Error
* 500 internal server error
* 501 not implementation
* 502 bad gateway
* 503 service unavailable
* 504 gateway time – out
* 505 HTTP version not supported