INTERNET

* a global system of interconnected computer networks that use the standard Internet Protocol Suite (TCP/IP) to serve billions of users worldwide.
* it is a network of networks that consists of millions of private, public, academic, business, and government networks, of local to global scope, that are linked by a broad array of electronic, wireless and optical networking technologies.

Vinton Cerf

* Father of the internet

World Wide Web

* global information medium which users can read and write via computers connected to the internet.
* the term is often mistakenly used as a synonym for the internet itself, but the web is a service that operates over the internet, as e-mail does.
* September 1994
  + Tim Berners-Lee founded the World Wide Web Consortium (W3C) as the Massachusetts Institute of Technology with the support of the Defense Advanced Research Projects Agency (DARPA) and European Commission.

Tim Berners-Lee

* Father of the Web

HTTP (Hypermedia)

* Application layer used primarily to retrieve hypertext (or hypermedia) documents and resources on the World Wide Web.
* Jointly developed by the W3C and IETF

Protocol

* Set of rules need to be followed.

HTTP Versions

* HTTP 0.9
  + 1991
  + 3 Methods – Get, Head, Post
* HTTP 1.0
  + RFC 1945, May 1996
  + All 8 methods
* HTTP 1.1
  + RFC 2068, Jan 1997
  + RFC 2616, June 1999

HTTP Fundamentals

* HTTP runs on top of TCP/IP, using TCP port 80 by default (TCP port 443 for HTTPS)
* HTTP resources are identified using URIs (RFC 2396, RFC 3896) or more
* URLs (http/https)
  + Scheme
  + Authority
    - User/authentication information/credentials
    - Host
      1. Domain name of the server where the resource resides
    - Port number
  + Path to the resource
    - resolved relative to the document root on the server
      1. May refer to a static or dynamic resource
  + Query
    - Provides as key value pairs, with ampersand (&) separators between key /value pairs
    - May be URL encoded
  + Fragment Identifier
    - example: http://usr:pwd@server.org:81/info/profile.php?id=2113#add
* HTTP is based on client server architecture
  + Clients aka User Agents (UA)
    - Web browsers
    - Web crawlers
    - Email clients
    - Other end user and tools and applications.
  + Servers
    - Origin servers
    - proxy servers
    - gateways
    - tunnels
* HTTP uses a request-response standard protocol
  + Client sends an HTTP request to the server
  + Server processes the request and replies with an HTTP response message
* HTTP is a stateless communication protocol
  + Servers do not keep information about clients in between requests
  + Web applications effect session tracking using mechanism such as cookies on URL-encoded session information to keep track of related client requests
* HTTP provides support for other functionalities such as
  + Cache Control
  + Content media type (MIME) specifications
  + Language and character set specifications
  + Content/transfer coding
  + Client-server protocol negotiations
  + Persistent connections
  + Request pipelining
  + etc.

Cookie

* very small text file

Cache

* Local storage/copy of resource that is fetched from a server

HTTP Request Message

* Request Line (crlf terminated line consisting of the 3-spaced)
* Message headers (general, requests and/or entity headers)
* Empty line (crlf)
* Message body (optional)

\*\*Fiddler

* web debugging proxy

HTTP Response Messages

* Status line
  + http protocol version
  + status code
  + reason phrase
* Message headers
* Empty line (crlf)
* Message body (optional)

HTTP Request Methods

1. Get
   * most commonly used HTTP method
   * used to request from the server the retrieval of the resource id by the request uri; the retrieved response is returned in the message body of the response as an entity

* Head
  + identical to GET, except the message body is not included in the response
  + also used to retrieved response is returned in the message body of the response as an entity
* Post
  + used to request the server accept data entities enclosing in the message body for processing by the response identified by the request URI
  + typically used in submitting html form data
* Put
  + request the server to store the enclosed entity in the message under the specified request URI
* Delete
  + request the server to delete the resource identified by the request URI
* Options
  + used to request from the server information about the communication options available for the response identified by the required URI
* Trace
  + request the server to “echo” back to the client the received request
  + typically used for testing/diagnostics of the request chain
* Connect
  + reserved for use of tunneling proxy servers.

\*\*Idempotent Methods

* + The methods GET, HEAD, PUT, and DELETE share this property

HTTP message headers

* General Header Fields
  + Cache-control
  + Connection
  + Date
  + Pragma
  + Trailer
  + Transfer-encoding
  + Upgrade
  + Via
  + Warning
* Request Header Fields
  + 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-forward
  + Proxy-authorization
  + Range
  + Referrer
  + User-agent
* Response Header Fields
  + Accept-range
  + Age
  + E-tag
  + Location
  + Proxy-authenticate
  + Retry-after
  + Server
  + Vary
  + WWW-Authenticate
* Entity Header Fields
  + Allow
  + Content-encoding
  + Content-language
  + Content-length
  + Content-location
  + Content-MD5
  + Content-range
  + Content-type
  + Expires
  + Last Modified

HTTP Status Codes

* Informational (1xx)
  + 100 Continue
  + 101 Switching Protocols
* Success (2xx)
  + 200 Ok
  + 201 Created
  + 202 Accepted
  + 203 Non-authoritative Information
  + 204 No Content
  + 205 Reset Content
  + 206 Partial Content
* Redirection (3xx)
  + 300 Multiple choices
  + 301 Moved permanently
  + 302 Found
  + 303 See Other
  + 304 Not modified
  + 305 Use proxy
  + 306 Switch proxy
  + 307 Temporary Redirect
* Client Error (4xx)
  + 400 Bad Request
  + 401 Unauthorized
  + 402 Payment Required
  + 403 Forbidden
  + 404 Not Found
  + 405 Method not Allowed
  + 406 Not acceptable
  + 407 Proxy Authentication Required
  + 408 Request timeout
  + 409 Conflict
  + 410 Gone
  + 411 Length Required
  + 412 Precondition Failed
  + 413 Request Entity Too Large
  + 414 Request-URI Too Long
  + 415 Unsupported Media Type
  + 416 Request Range Not Satisfiable
  + 417 Expectation Failed
* Server Error (5xx)
  + 500 Internal Server Error
  + 501 Not Implemented
  + 502 Bad Gateway
  + 503 Service Unavailable
  + 504 Gateway Timeout
  + 505 HTTP Version not Supported

Hypertext Markup Language

* Language used to markup documents (i.e web pages) in the WWW
  + Structure (and content)
  + Presentation (mostly deprecated in favor of style sheets)
* Versions
  + HTML 2.0, 3.2,4.0
  + HTML 4.01
    - Strict – deprecated
    - Transitional – still make use the deprecated
    - Frameset – using frames
  + HTML 5 (Working draft)
* Head
  + title, base, link, meta, style, script
* Body
  + Grouping elements
    - div, span
  + Headings
    - h1 – h6
  + Paragraphs, line breaks, horizontal ruler
    - p , br , hr
  + Lists
    - ul, ol, li, dl, dt, dd, dir, menu
  + Tables
    - table, th, td, thead, tfoot, tbody, colgroup, col
  + Structured text
    - Phrase elements
      * em, strong, dfn. code, samp, kbd, var, cite, abbr, acronym
    - Quotations
      * blockquote, q
    - Subscripts and superscripts
      * sub, sup
    - Preformatted texts
      * pre
  + Font styles and alignments
    - tt, i, b, big, small, strike, s, u, font, basefront, center
  + Document changes
    - ins, del
  + Links and anchors
    - a
  + Objects, images, applets
    - object, img, param, applet
  + Scripts
    - script, noscript
  + Miscellaneous
    - address, bdo
  + Frames, noframes, iframe
  + Frameset (for frameset DTD)

Extensible Hypertext Markup Language (XHTML)

* reformulation of HTML in XML (Extensible markup Language)
* Versions
  + XHTML 1.0
  + XHTML 1.1
  + XHTML 1.2
  + XHTML 2.0
  + XHTML 5 (working draft)
* Basic differences between HMTL and XHTML
  + All elements have beginning and ending tags
  + All elements are nested properly
  + Elements and attribute names are case sensitive (lowercase)
  + Attribute values are quoted
  + Attribute values cannot be minimized
* XHTML Doc. Components
  + Document Type Declaration (DOCTYPES)
    - HTML 4.01 DOCTYPES
    - XHTML 1.0 DOCTYPES
  + Elements
    - Tags
    - Content (elements)
    - Attributes
      * id, class, title, alt, lang, intrinsic event attributes (e.g onClick)
    - Character Entity References
      * &lt; &gt; &amp; &nbsp; &shy;
* XHTML Elements
  + Block-level and inline (aka text-level) elements
    - Block-level elements represent “larger” document structures and may contain inline and other block level elements
    - Inline elements contain only data and other inline elements
  + Author Styles
    - External styles sheets (recommended)
    - Embedded styles
    - Inline styles
  + User Style
  + User Agent Style (example default style sheet for HTML 4)

Quirks

* Older ways of rendering documents

Standards

* Following the specification by W3M for HTML, CSS