**1st Topic: Over view of the WWW**

**WWW-collection of web resources (website, webpages) by the use of applications access through internet.**

**Sir Tim Berners Lee , 1989**

**2nd Topic: Hypertext Transfer Protocol (HTTP)**

**HTTP- application layer communication protocol used to access resource (hypertext/ hypermedia) on the World Wide Web.**

**TIM BERNERS LEE –Invented it at CERN in 1989, jointly developed by the W3C and IETF.**

**Version HTTP 0.9 (1991) – not persistent connection “GET” method –initially text based.**

**HTTP 1.0 (RFC 1945 May 1996) - “POST & HEAD” method single connection, every request 1 connection.**

**HTTP 1.1 (RFC 2068 Jan 1997) (RFC 2616 Jan 1999) - 8 method was already added GET, DELETE, HEAD, TRACE, POST, CONNECT, PUT, OPTIONS.**

**RFC (7230-7235 June 2014) - Re writing Protocol.**

**HTTP2 (RFC 7540 May 2015) - can already anticipate**

**-HTTP is based on a client-server architecture typically runs on top TCP/IP using transmission control/ internet protocol…**

**-Using TCP PORT 80 by default or TCP port 443 for HTTPS (HTPP OVER SSL/TLS).**

**TCP – Transmission Control Protocol**

**IP – Internet Protocol**

**SSL- Security Socket layer**

**TLS- Transport Layer Security**

**IANA- Internet Assigned Numbers Authority**

**ICANN- Internet Corporation for assigned Names and Numbers**

**HTTP is based on client – server architecture.**

**SERVER:**

* **Origin Server**
* **Proxy Server, Gateways, Tunnels Clients A.K.A User Agents (U.A).**
* **Web Browser, Web Crawlers/ Web Spiders Other end user tools and application.**

**HTTP uses a request – response standard protocol**

**Client – send HTTP request message to the server.**

**Server – process the request and replies with an HTTP response message.**

**HTTP is a stateless communication protocols.**

**Server – do not keep information about clients in between request.**

**HTTP provides support for the other functionalities such as**

**-Cache Control Multint (Net Mail Extension)**

**-Control Media Type (MIME) Specification**

**-Language and Character Specification**

**-Content/Transfer Protocol**

**-Content Negotiation**

**-Client – Server Protocol Negotiations RE2616**

**-Request Pipelining/ Multiplexing**

**-Authentication/ Authorization**

**-ETC.**

**HTTP RESOURCE ADDRESSING**

* **HTTP resource are identified using URL’s (RFC 3986) or more specifically HTTP.**

**URLS - Scheme (https)**

**-Authority**

* + - **USER information or authentication credentials (deprecated).**
    - **HOST domain name (resolve using IP address using DNS of the server where the resource resides.**
    - **PORT NUMBER**

**-Path to resource (resolve relative to the document root on the server).**

* + - **May refer to astatic or dynamic resource.**

**-Query typically provided as key = value pairs with ampersand (&) separators between key value pairs.**

* + - **May be URL- ENCODED.**

**-Fragment I identifier**

**HTTP REQUEST MESSAGE – 4 PARTS (Header, Body….)**

**-Request Line (RLF –terminated line consisting of three space – separated values)**

* + - **Method**
    - **Request target**
    - **Protocol Version**

**MESSAGE HEADER**

**-HTTP RESPONSE MESSAGE**

**STATUS LINE (RLF- terminated line consisting of 3 space separated values)**

**GET (HEAD, DELETE, POST, OPTIONS TRACE)**

* + - **Identical to get expect that the server does not send a message body in the response.**

**PUT – CREATE OR REPLACE**

**SAFE METHODS/ READ ONLY – GET, HEAD, TRACE, OPTIONS**

**IDEMPOTENT METHOD**

* + - **Intended effect on the server of multiple.**
    - **No exception of any charges.**
    - **Additional method : PUT and DELETE**
    - **Net effect does not change.**

**CACHEABLE METHODS**

* + - **Allowed to be stored for future use.**
    - **Additional Method : POST**

**HTTP STATUS CODE**

* **INFORMATIONAL (1xx) providing info regarding request.**

**-100 Continue**

**-1001 Switching Protocols**

* **SUCCESS (2xx)**

**-200 Ok**

**-201 Created**

**-202 Accepted**

**-203 Non Authoritative Info**

**-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 Unused (not being used)**

**-307 Temporary Redirect**

* **CLIENT ERROR (4xx)**

**-400-426 Errors on the Client Side**

**-400 Bad Request**

**-401 Unauthorized**

**-402 Payment Required**

**-403 Forbidden**

**-404 Not Found**

**-405 Method not Allowed**

* **SERVER ERROR**

**-500 International Server Error CONFIGURATION**

**-501 Not Implemented**

**-502 Bad Gateway**

**-503 Service Unavailable**

**-504 Gateway Timeout**

**-505 HTTP version not supported**

**MESSAGE HEADERS (General, Response and/ or Entity Headers)**

**GENERAL HEADER FIELDS**

* **Cache – Control**
* **Pragma**
* **Transfer Encoding – Compression**
* **Connections – Typical value keep alive**
* **Date**

**REQUEST HEADER FIELDS**

* **Accept (Media Type; languages)**
* **Authorization – provided credentials**
* **Host- when has a request**
* **Accept Encoding**
* **Accept**
* **Accept Language**

**RESPONSE HEADER FIELDS**

* **Vary**
* **ETAG**
* **WWW. Authenticate**
* **Accept Rangers**

**RESPONSE HEADER FIELDS**

* **Allow**
* **Expires**

**Web Day- distributed authorizing versioning.**

**445- ERROR**

**Request Method – Profind,Proppatch, MKCOL, Copy, Move, Lock, Unlock**

**Message Header – Day, Depth Destination, It-lock-token, Overwrite, Timeout**

**Status Codes- (207) Multi-status, (422) Unprocesse able Entity, (423) Locked, (424) Failed Dependency, (507) Insufficient Storage.**

**3rd Topic: Hypertext Mark up Language**

HTML

* Reformulation of HTML in XML.
* Intended to facilitate the introduction of new HTML element and/or attributes.
* Provide interoperability between HTML and XML based system.

XHTML 1.0 (W3C Recommendation, Jan 2000)

XHTML 1.1 Module-Based XHTML (W3C Recommendation, May 2001)

IAN HICKSON -Undertaken by the WHATWG (led by Apple, Mozilla, Opera) in 2004, which the W3C signifying interest to participate in 2006.

* The first draft of HTML5 was published in 2008.

HTML5 incorporated specification from HTML, XHTML and the DOM, and included various extensions of existing APIs and introduced new APIs as well.

REVISION THE HTML

* HTML5 (W3C Recommendation, October 2014)
* HTML 5.1 (W3C Recommendation, November 2016)
* HTML 5.2 (W3C Recommendation, December 2017)
* HTML Living Standard (WHATWG)
* Differences between HTML4 and HTML 5.

HTML DOCUMENT STRUCTURE

Root Element (html)

* Declarative header section containing document metadata (title element, base element, link element, meta element, style element), delimited by the head element.
* Document body containing the documents actual content.

<!DOCTYPE html>

* Case sensitive
* Standard rendering.
* Ensuring the markup is valid.

TAGS

* An element has beginning and ending tag, that consists of a content.

CONTEXT MODEL

* Definition of what will be the content on the element.

ATTRIBUTES AND ATTRIBUTES VALUES

* Global Attributes
* Custom-Non-Visible Data
* WAI-ARIA Attributes

Notes: All attribute name on HTML element and HTML documents get SCII-lowercased automatically, so the ASCII restriction on ASCII uppercase letters doesn’t affect such documents.

MHTML – Mathematical Markup Language

SVG – Scalable Vector Graphics