## HTML Introduction:

**HTML** stands for HyperText Markup Language. It is used to design web pages using a markup language. HTML is the combination of Hypertext and Markup language. Hypertext defines the link between the web pages. A markup language is used to define the text document within tag which defines the structure of web pages. This language is used to annotate (make notes for the computer) text so that a machine can understand it and manipulate text accordingly. Most markup languages (e.g. HTML) are human-readable. The language uses tags to define what manipulation has to be done on the text.

HTML is a markup language used by the browser to manipulate text, images, and other content, in order to display it in the required format.  **HTML page structure:** The basic structure of an HTML page is laid out below. It contains the essential building-block elements (i.e. doctype declaration, HTML, head, title, and body elements) upon which all web pages are created.

**Every tag should be closed if tagging will be not closed then the application it consider to the next closing tag:**

**<html>:- Opening tag of html page**

**<head>**

**<title> BIO DATA</title>**

**</head>: closing tag of the HTML heading tag**

**<body>**

**<h2> headline tag </h2>**

**<p> my paragraph usually used for the text blocks </p>**

**<ul> # used for the un ordered items**

**<li>Unordered item 1</li> # li is used for the list**

**<li>Unordered item 2</li>**

**</ul>**

**<ol> :- ordered items**

**<li>Ordered item 1</li>**

**<li>Ordered item 2</li>**

**</ol>**

**<img src=”/path/to/image.png” /> :** inserting image code :- Notice the /> at the end (instead of </img>). This is because image elements have a source attribute (src) which fetches the image to be displayed. There’s no content that needs to go inside. There are other elements, similar to img, that don’t require a closing tag.

For more information, check out **<a href="https://www.google.com">Google</a> :-** inserting the link in the HTML language

**</body>**

**</html>:- Closing tag of html page**

* Along with these we can also insert box, can build form using HTML language.
* HTML Status Codes: - Codes are the more important for a web application penetration tester to understand the response of the request which was requested by the browser.
* **2XX:- Success**: - This class of status codes indicates the action requested by the client was received, understood, and accepted.
* 200 OK
* 201 Created
* 202 Accepted etc
* **3xx redirection:-** This class of status code indicates the client must take additional action to complete the request. Many of these status codes are used in URL redirection.A user agent may carry out the additional action with no user interaction only if the method used in the second request is GET or HEAD. A user agent may automatically redirect a request. A user agent should detect and intervene to prevent cyclical redirects.
* [301 Moved Permanently](https://en.wikipedia.org/wiki/HTTP_301)
* [302 Found (Previously "Moved temporarily")](https://en.wikipedia.org/wiki/HTTP_302)
* [303 See Other](https://en.wikipedia.org/wiki/HTTP_303) etc
* **4xx client errors**
* 400 Bad Request
* 401 Unauthorized (RFC 7235)
* 402 Payment Required
* [403 Forbidden](https://en.wikipedia.org/wiki/HTTP_403)
* [404 Not Found](https://en.wikipedia.org/wiki/HTTP_404)
* **5xx server errors:-** The [server](https://en.wikipedia.org/wiki/Server_(computing)) failed to fulfill a request.

Response status codes beginning with the digit "5" indicate cases in which the server is aware that it has encountered an error or is otherwise incapable of performing the request. Except when responding to a HEAD request, the server *should* include an entity containing an explanation of the error situation, and indicate whether it is a temporary or permanent condition. Likewise, user agents *should* display any included entity to the user. These response codes are applicable to any request method.

* 500 Internal Server Error
* 501 Not Implemented
* 502 Bad Gateway
* 503 Service Unavailable
* 504 Gateway Timeout