Unit-IV
Web Basics

Topics

1. Introduction
2. HTTP Protocol
3. HTML
4. URL Basics
5. Web Server
6. Web Framework
7. Introduction to WSGI

1. Introduction

- Internet: It is a network of networks in which users at any one computer can exchange information from any other computer .

- Web: It is a subset of the Internet consisting of the pages that can be accessed by a Web browser.

- Web Page: A Web page also known as Electronic Page is a part of the World Wide Web.
- It is a basic unit of every web site containing the information.

- A web page can contain an article, a paragraph, photographs, and it is generally a combination of text and graphics.

- Website: A website is a collection of publicly accessible, interlinked Web pages that share a single domain name.
- Websites can be created and maintained by an individual, group, business or organization to serve a variety of purposes.

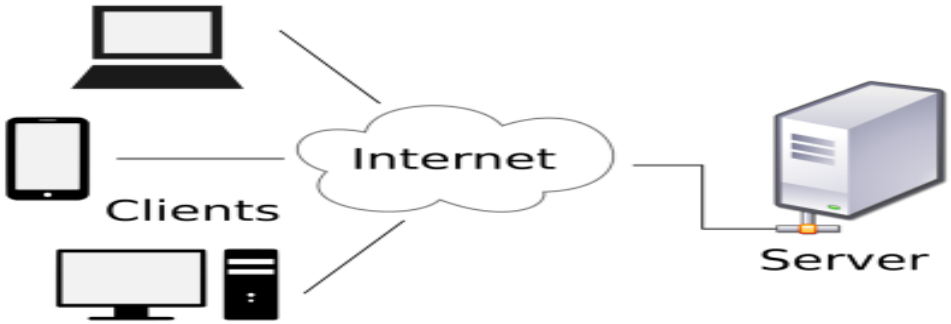
- World wide web: World Wide Web, which is also known as a Web, is a collection of websites or web pages stored in web servers and connected to local computers through the internet.
- These websites contain text pages, digital images, audios, videos, etc.
- Web browser: It is a software application used to access information on the World Wide Web.
- User can request for any web page by just entering a URL into address bar.
- Web browser can show text, audio, video, animation and more.
- It is the responsibility of a web browser to interpret text and commands contained in the web page.

Client Server Architecture:

- A centralized network architecture that classifies computer into two sections,

➢ Client

➢ Server



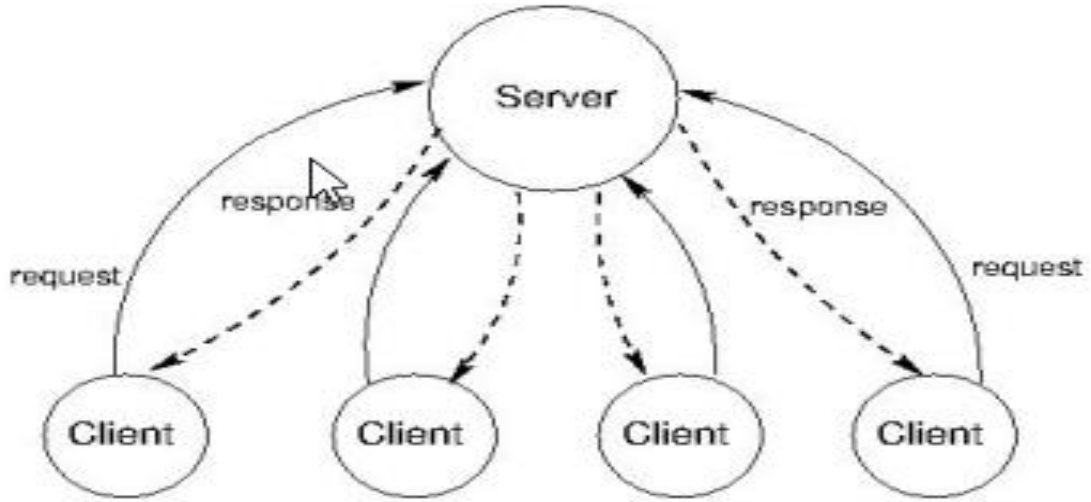
A client is the requester, which can be a program that we use to make requests through the network with parameters included.

A server is the response provider, which is a program that listens for the client's requests and responds to them.

- The server component provides a function or service to one or many clients, which initiate requests for such services.

- Server itself might be a client. For example, the server could request something from a database server, which in this case, would make the server a client of the database server.

  Examples of computer applications that use the client–server model are Email, network printing, and the World Wide Web.



- Components:

1. The Server to listen for requests
2. The Clients to connect to the server
3. The Connection medium

- Connectors: Protocols, Remote procedure calls (RPC)

1. Protocols: The special set of rules that end points in a telecommunication connection use when they communicate.

   Examples: TCP/IP, HTTP, FTP and etc.

2. RPC: It is protocol that one program can use to request a service from a program located in another computer on a network without having to understand the network's details.

Data is sent/received though connectors

➢ Conditions:

- clients cannot communicate directly with each other. If needed, the server acts as a message relay for the clients to communicate.
- Only clients can initiates communication.
- All workloads are done at the server-side.

2.Hyper Text Transfer Protocol

- HTTP is a protocol for encoding and transporting information, like text, images, sound, video and multimedia, between a web browser and a web server over the Internet.

- Basically, HTTP is a TCP/IP based communication protocol, that is used to deliver data (HTML files, image files, query results, etc.) on the World Wide Web.

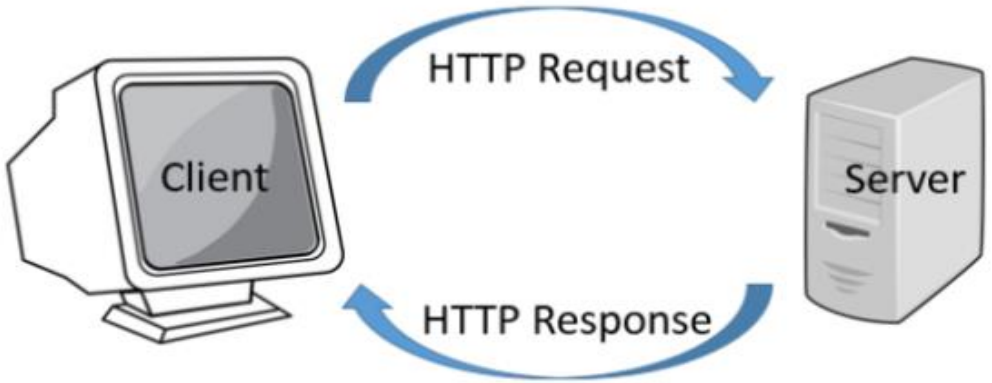
- The default port for browser access is 80, which provides a standardized way for computers to communicate with each other.

Basic Features:

There are three basic features that make HTTP a simple but powerful protocol:

1. HTTP is connectionless: The HTTP client, i.e., a browser initiates an HTTP request and after a request is made, the client waits for the response.

- The server processes the request and sends a response back after which client disconnect the connection.
- Further requests are made on new connection like client and server are new to each other.



2. HTTP is media independent: It means, any type of data can be sent by HTTP as long as both the client and the server know how to handle the data content.

- It is required for the client as well as the server to specify the content type using appropriate Multipurpose Internet Mail Extension MIME-type.

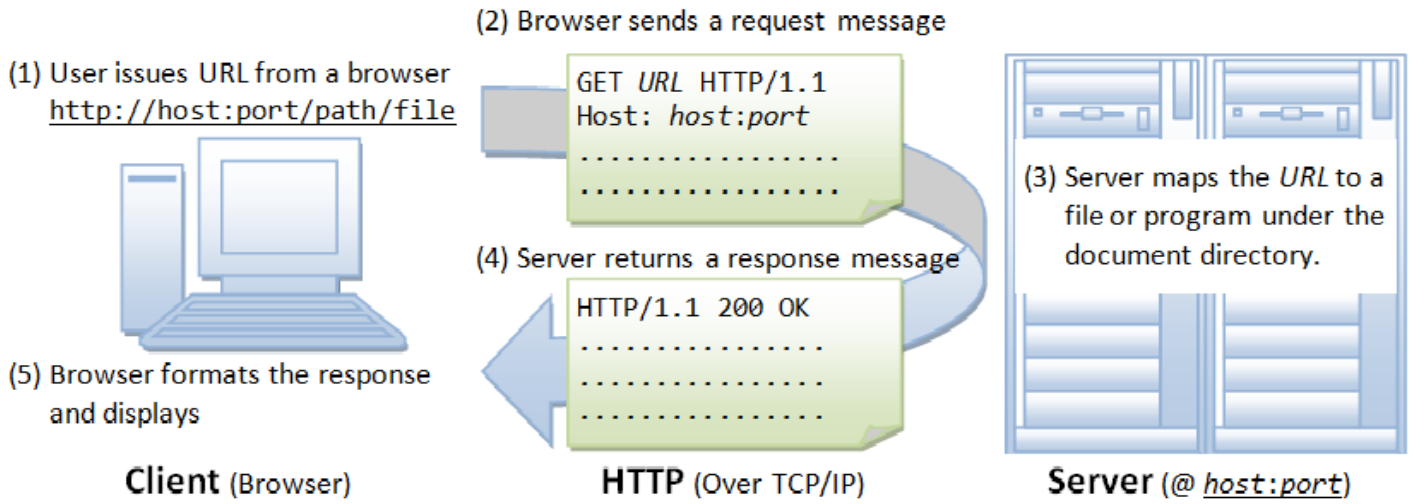
3. HTTP is stateless: The server and client are aware of each other only during a current request. Afterwards, both of them forget about each other.

- Due to this nature of the protocol, neither the client nor the server can retain information between different requests across the web pages.

- Whenever , a URL is issued from browser to get a web resource using HTTP.

  e.g. http://www.nowhere123.com/index.html

- The browser turns the URL into a *request message* and sends it to the HTTP server.
- The HTTP server interprets the *request message*, and returns you an appropriate *response message*, which is either the resource you requested or an error message.
- This process is illustrated below:



2.1 Uniform Resource Locator (URL)

- A URL (Uniform Resource Locator) is used to uniquely identify a resource over the web.
- URL has the following syntax:

protocol://hostname:port/path-and-file-name

There are 4 parts in a URL:

- ***Protocol*:** The application-level protocol used by the client and server, e.g., HTTP, FTP, and telnet.
- ***Hostname*:** The DNS domain name (e.g., www.nowhere123.com) or IP address (e.g., 192.128.1.2) of the server.
- ***Port*:** The TCP port number that the server is listening for incoming requests from the clients.
- ***Path-and-file-name*:** The name and location of the requested resource, under the server document base directory.

For example,

http://www.amazon.com/docs/index.html

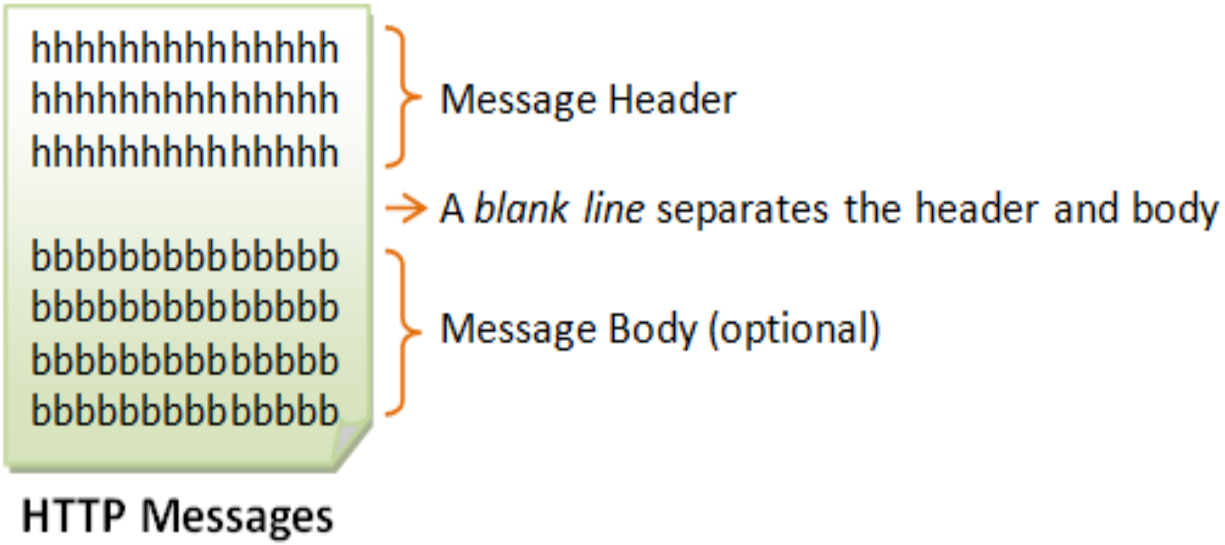
- The communication protocol is HTTP
- The hostname is www.amazon.com
- The port number was not specified in the URL, and takes on the default number, which is TCP port 80 for HTTP.
- The path and file name for the resource to be located is "/docs/index.html".

Other examples of URL are:

- ftp://www.ftp.org/docs/first.txt
- mailto:user@trail123.com
- news:soc.diversity.India
- telnet://www.applications123.com/

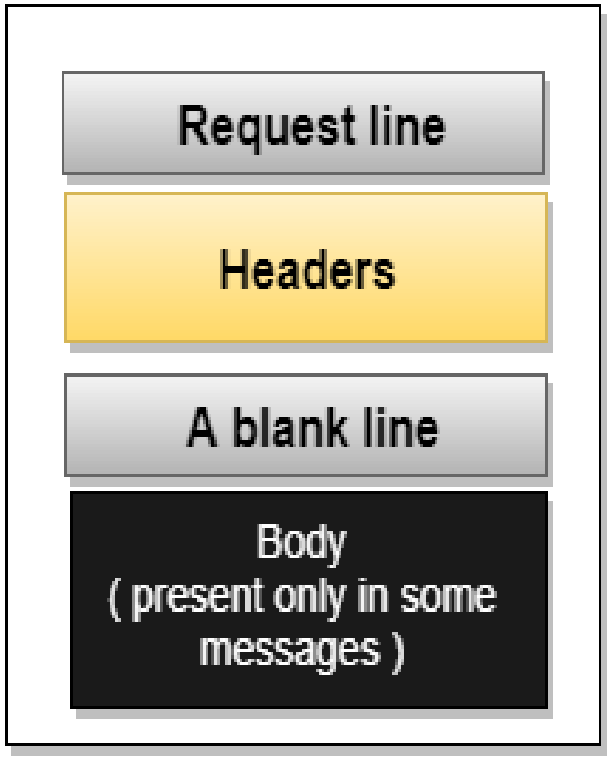
2.2 HTTP Request and Response Messages:

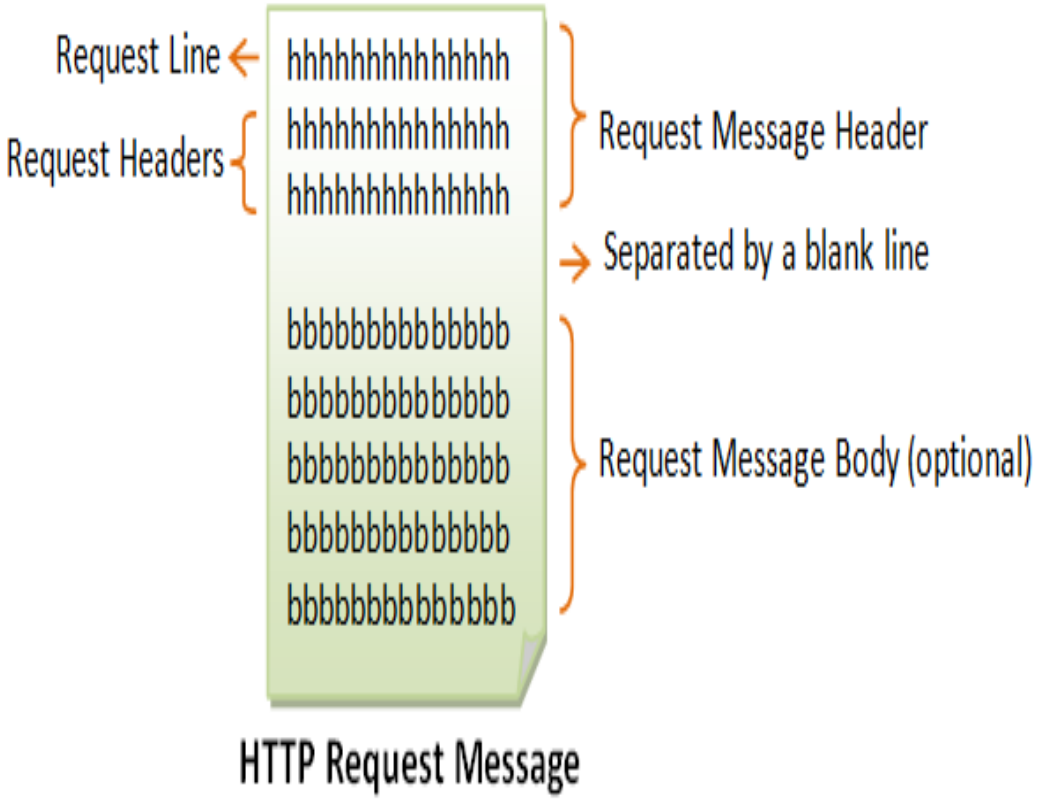
- HTTP client and server communicate by sending text messages.
- The client sends a *request message* to the server.
- The server, in turn, returns a *response message*.
- An HTTP message consists of a *message header* and an optional *message body*, separated by a *blank line*, as illustrated below:



1. HTTP Request Message

- The format of an HTTP request message is as follow:





HTTP Request Message

Request Line :

- The first line of the header is called the *request line*, followed by optional *request headers*.
- The request line has the following syntax:

request-method-name request-URI HTTP-version

- *request-method-name*: HTTP protocol defines a set of request methods, e.g., GET, POST, HEAD, and OPTIONS. The client can use one of these methods to send a request to the server.
- *request-URI*: specifies the resource requested.
- *HTTP-version*: Two versions are currently in use: HTTP/1.0 and HTTP/1.1.
- Examples of request line are:
  - GET /test.html HTTP/1.1
  - HEAD /query.html HTTP/1.0
  - POST /index.html HTTP/1.1

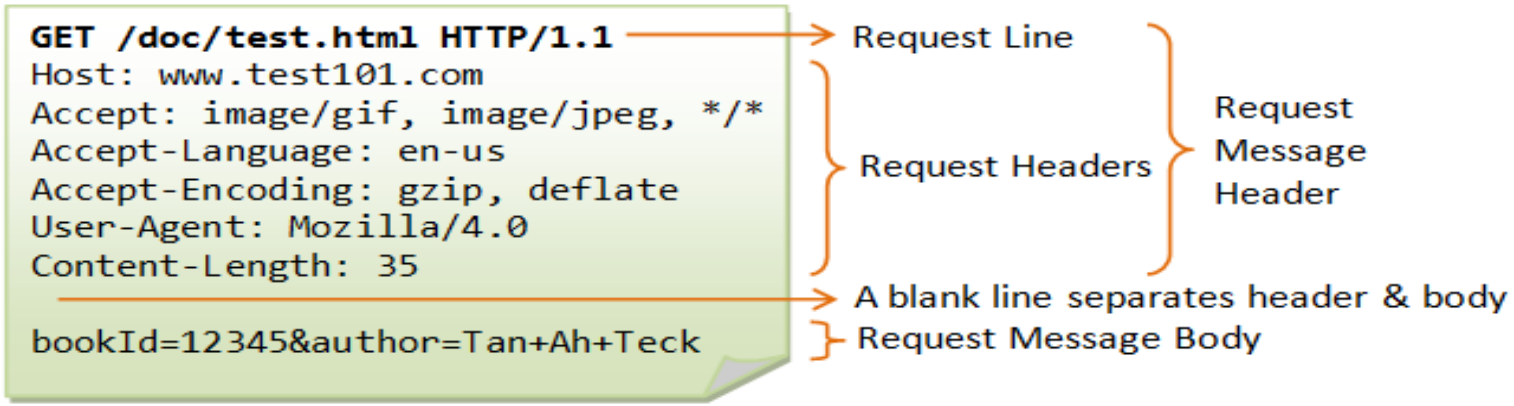
Request Headers:

- The request headers are in the form of name:value pairs.
- Examples of request headers are:

request-header-name:request-header-value1,request-header-value2, ...

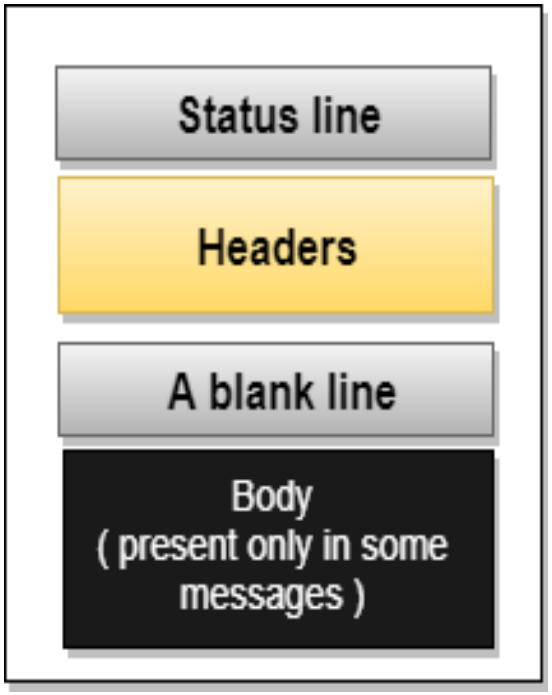
- Host: www.xyz.com
- Connection: Keep-Alive
- Accept: image/gif, image/jpeg, */*
- Accept-Language: us-en

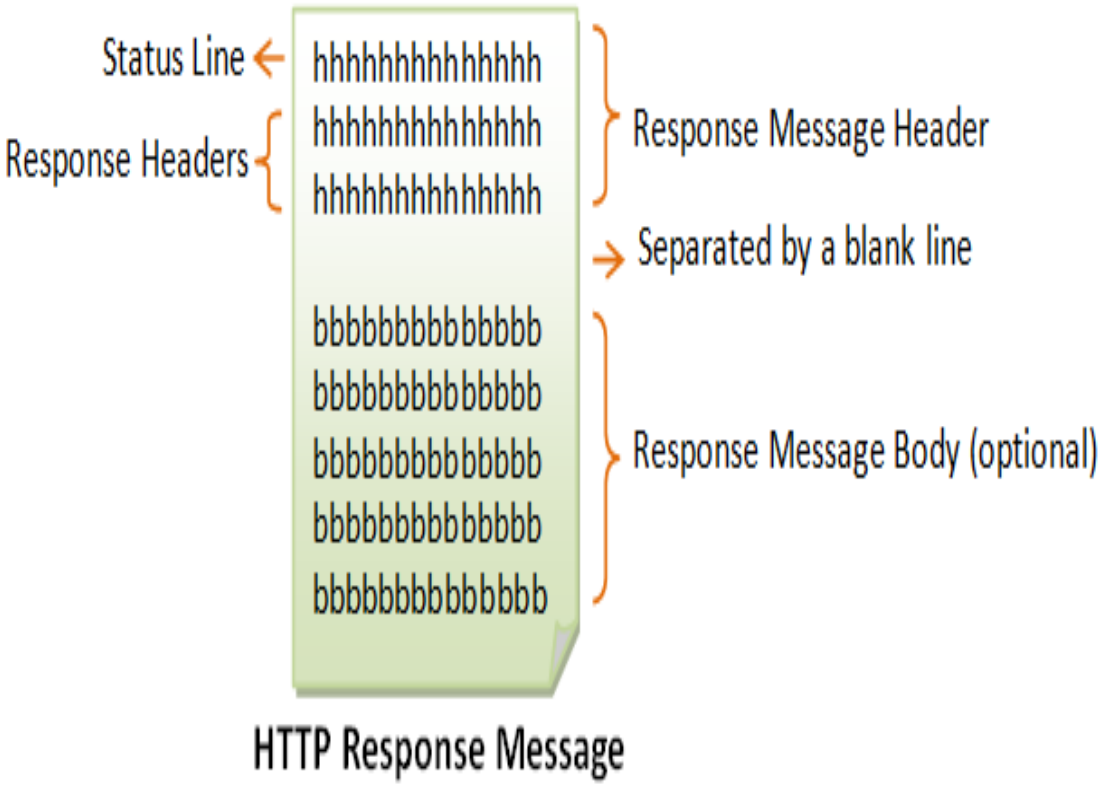
Example : The following shows a sample HTTP request message:



2. HTTP Response Message

- The format of the HTTP response message is as follows:





Status Line:

- The first line is called the *status line*, followed by optional response header(s).

- The status line has the following syntax:

HTTP-version status-code reason-phrase

- *HTTP-version*: The HTTP version used in this session. Either HTTP/1.0 and HTTP/1.1.

- *status-code*: a 3-digit number generated by the server to reflect the outcome of the request.

- *reason-phrase*: gives a short explanation to the status code.

Common status code and reason phrase are

- "200 OK"
- "404 Not Found"
- "403 Forbidden“
- "500 Internal Server Error"

Examples of status line are:

- HTTP/1.1 200 OK
- HTTP/1.0 404 Not Found
- HTTP/1.1 403 Forbidden

Response Headers:

- The response headers are in the form name:value pairs:

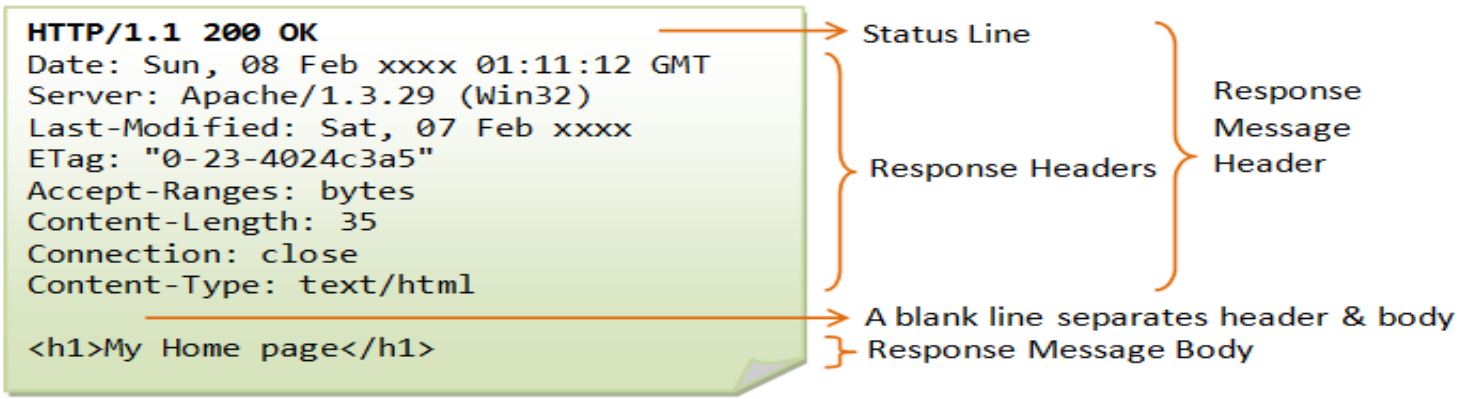
response-header-name: response-header-value1, response-header-value2, ...

Examples of response headers are:

- Content-Type: text/html
- Content-Length: 35
- Connection: Keep-Alive
- Keep-Alive: timeout=30, max=100

The response message body contains the resource data requested.

Example: The following shows a sample response message:



HTTP Request Methods :

- HTTP protocol defines a set of request methods.
- A client can use one of these request methods to send a request message to an HTTP server.

The methods are:

1. GET: A client can use the GET request to get a web resource from the server.

2. HEAD: A client can use the HEAD request to get the header that a GET request would have obtained.

1. POST: Used to post data up to the web server.

2. PUT: Ask the server to store the data.

5. DELETE: Ask the server to delete the data.

6. TRACE: Ask the server to return a diagnostic trace of the actions it takes.

7. OPTIONS: Ask the server to return the list of request methods it supports.

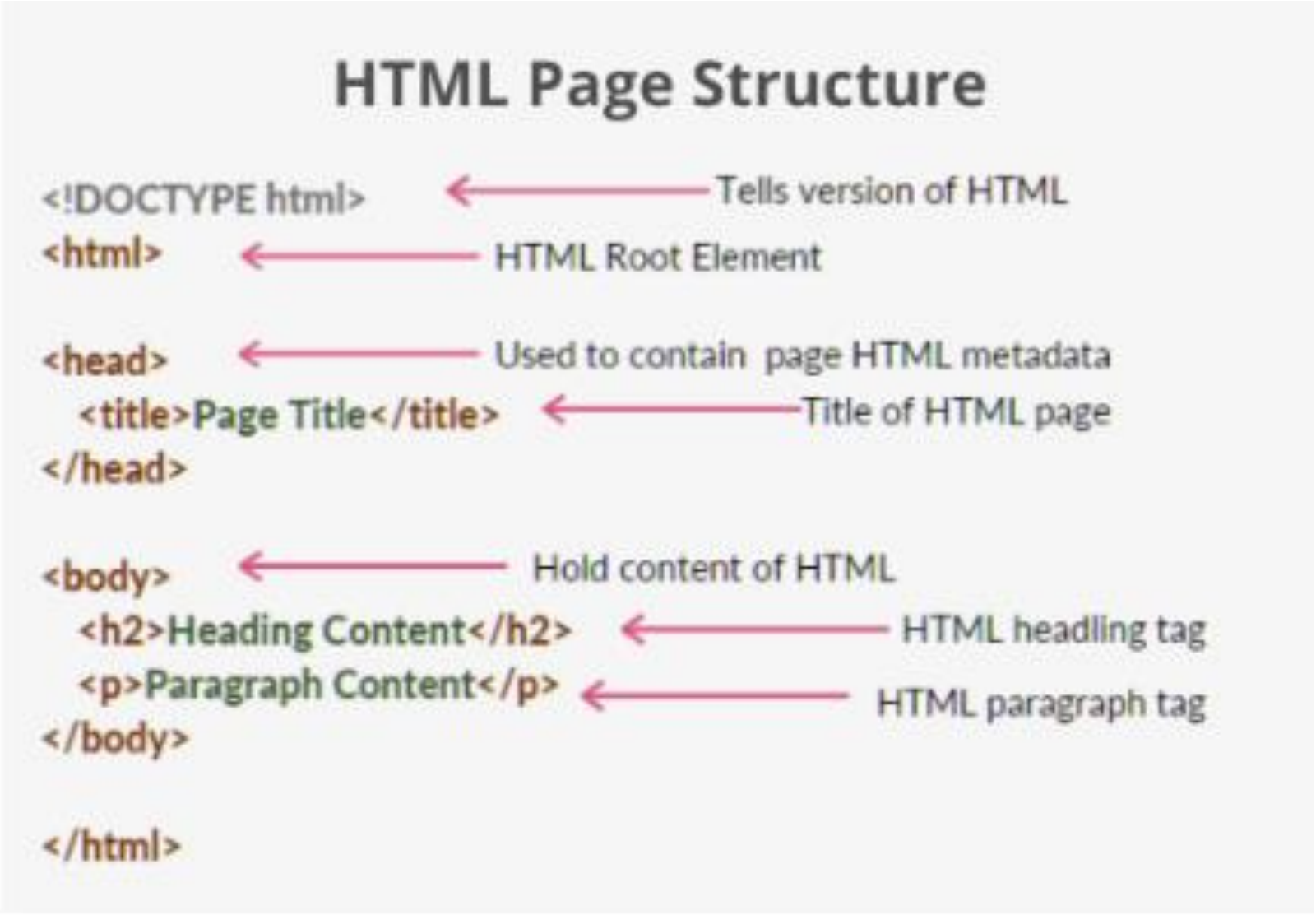
8. CONNECT: establishes a connection to the server identified by a given URI.

3. Hyper Text Markup Language(HTML)

- HTML is the standard language for creating web pages and web applications.
- At present, HTML5 version is used which is an extension of HTML4.01 version.

- HTML is used to format web pages with different tags available.
- Hypertext means links that connect web pages together.
- Markup means HTML tags specifying type of text.
- HTML uses markup to define text, images and other content to display in a web browser.

- HTML page/document structure



- An HTML element is represented by "tags", which consist of the name surrounded by angular brackets "<" and ">".
- The name of an element inside a tag is case insensitive but recommended practice is to write tags in lowercase.

```
<!DOCTYPE html>
<html>

   <head>
      <title>This is document title</title>
   </head>

   <body>
      <h1>This is a heading</h1>
      <p>Document content goes here.....</p>
   </body>

</html>
```

Simple Tags:

1. **<!DOCTYPE...> :**This tag defines the document type and HTML version.
2. **<html> :** This tag encloses the complete HTML document and mainly comprises of document header which is represented by <head>...</head> and document body which is represented by <body>...</body> tags.
3. **<head> :** This tag represents the document's header which can keep other HTML tags like <title>, <link> etc.
4. **<title> :** The <title> tag is used inside the <head> tag to mention the document title.
5. **<body> :** This tag represents the document's body which keeps other HTML tags like <h1>, <p> etc.
6. **<h1> :** This tag represents the heading.
7. **<p> :** This tag represents a paragraph.

1. The <!DOCTYPE> Declaration

- The <!DOCTYPE> declaration tag is used by the web browser to understand the version of the HTML used in the document.
- Current version of HTML is 5 and it makes use of the following declaration −

<!DOCTYPE html>

```
<!DOCTYPE html>
<html>
<body>

<h1>This is AI&ML class</h1>

<p>This is our first paragragh.</p>

</body>
</html>
```

This is AI&ML class

This is our first paragragh.

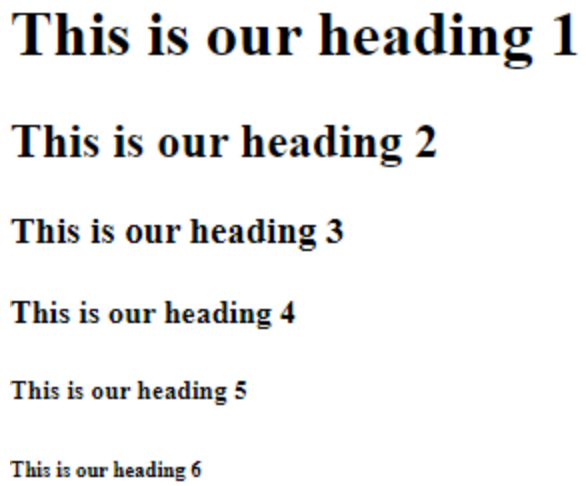
2. Heading Tags

- Any document starts with a heading.
- HTML also has six levels of headings, which use the elements **<h1>, <h2>, <h3>, <h4>, <h5>,** and **<h6>**.
- While displaying any heading, browser adds one line before and one line after that heading.

```
<!DOCTYPE html>
<html>
<body>

<h1>This is our heading 1</h1>
<h2>This is our heading 2</h2>
<h3>This is our heading 3</h3>
<h4>This is our heading 4</h4>
<h5>This is our heading 5</h5>
<h6>This is our heading 6</h6>

</body>
</html>
```



3. Paragraph Tag

- The ** ** tag offers a way to structure your text into different paragraphs.
- Each paragraph of text should go in between an opening and a closing tag as shown below in the example −

```
<!DOCTYPE html>
<html>
<body>

<p>It is our first paragraph.</p>
<p>It is our another paragraph.</p>

</body>
</html>
```

It is our first paragraph.

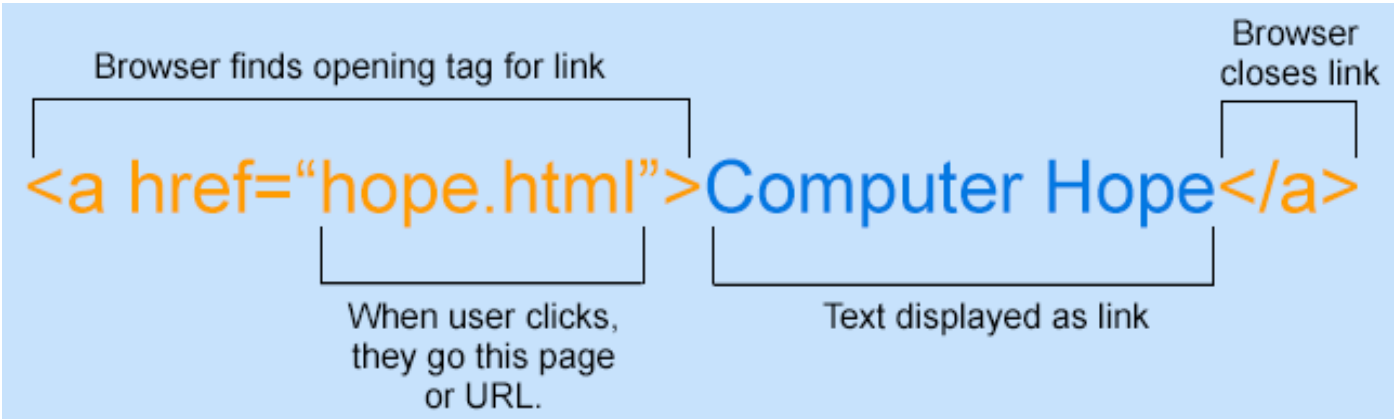
It is our another paragraph.

HTML Attributes:

- All HTML elements can have **attributes.**
- Attributes provide **additional information** about elements
- Attributes usually come in name/value pairs like: **name="value".**

<a> tag:

- The <a> tag defines a hyperlink which links a web page to another web page.
- The href attribute specifies the URL of the page the link goes to:



```
<!doctype html>
<html>
<head>
<title>
hi
</title>

</head>
<body>
<h1>
<a href="file:///C:/Users/LENOVO/Desktop/second.html">This is first html document</a>
</h1>
</body>
</html>
```

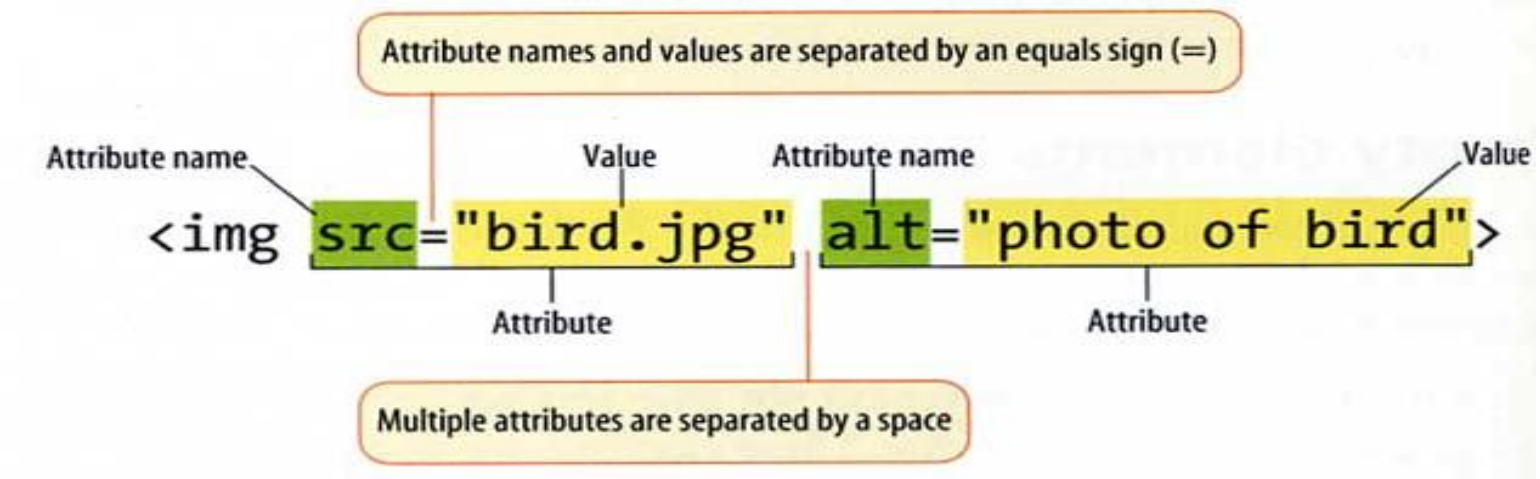
This is first html document

```
<html>
this is my second html doc
</html>
```

this is my second html doc

<img> tag:

- The <img> tag is used to embed an image in an HTML page.
- The src attribute specifies the path to the image to be displayed:
- The alt attribute specifies an alternate text for an image, if the image for some reason cannot be displayed.
- The width and height attributes, specifies the width and height of the image (in pixels).



first - Notepad

File Edit Format View Help

```
<!DOCTYPE html>
<html>
<body>
<h2>HTML Images</h2>
<p>Images are defined by img tag</p>
<img src="AIMLimage.png" alt="aiml.com" width="150" height="150">
</body>
</html>
```

File | C:/Users/LENOVO/Desktop/first.html

HTML Images

Images are defined by img tag

The style Attribute

Syntax: <*tagname* style="*property*:*value*;">

- The style attribute is used to add styles to an element, such as color, font, size, and more.

```
<!DOCTYPE html>
<html>
<body>
<h2>The style Attribute</h2>
<p>The style attribute is applied to elements</p>
<h1>
<p style="color:green;">This is a green paragraph.</p>
</h1>
</body>
</html>
```

The style Attribute

The style attribute is applied to elements

This is a green paragraph.

Background Color:

- The style= background-color attribute defines the background color for an HTML element.

```
<!DOCTYPE html>
<html>
<body style="background-color:pink;">

<h1>How are you AI&ML students?</h1>
<p>This is CS&AI Department</p>
</boody>
</html>
```

Text Color

- The CSS color property defines the text color for an HTML element
- Fonts
- The CSS font-family property defines the font to be used for an HTML element:
- Text Size
- The CSS font-size property defines the text size for an HTML element:
- Text Alignment
- The CSS text-align property defines the horizontal text alignment for an HTML element:

HTML Formatting Elements:

Formatting elements were designed to display special types of text:

1. <b> - Bold text
2. - Important text
3. <i> - Italic text
4. - Emphasized text
5. <mark> - Marked text
6. <small> - Smaller text
7. <del> - Deleted text
8. <ins> - Inserted text
9. <sub> - Subscript text
10. <sup> - Superscript text

HTML Comment Tag:

Comments to your HTML source by using the following syntax:

<!-- Write your comments here -->

HTML Styles – CSS

- CSS stands for Cascading Style Sheets.
- CSS saves a lot of work by controlling the layout of multiple web pages all at once.

What is CSS?

- Cascading Style Sheets (CSS) is used to format the layout of a webpage.

- With CSS, the color, font, the size of text, the spacing between elements, how elements are positioned and laid out, background images or background colors to be used are controlled.

- CSS can be added to HTML documents in 3 ways:

1. **Inline -** by using the style attribute inside HTML elements.
2. **Internal -** by using a <style> element in the <head> section.
3. **External -** by using a <link> element to link to an external CSS file.

- The most common way to add CSS, is to keep the styles in external CSS files.

1. Inline CSS

- An inline CSS is used to apply a unique style to a single HTML element.
- An inline CSS uses the style attribute of an HTML element.
- The following example sets the text color of the <h1> element to purple, and the text color of the <p> element to blue:

```
<!DOCTYPE html>
<html>
<body>

<h1 style="color:purple;">This is an AI&ML class room</h1>

<p style="color:blue;">It is for III sem students</p>

</body>
</html>
```

This is an AI&ML class room

It is for III sem students

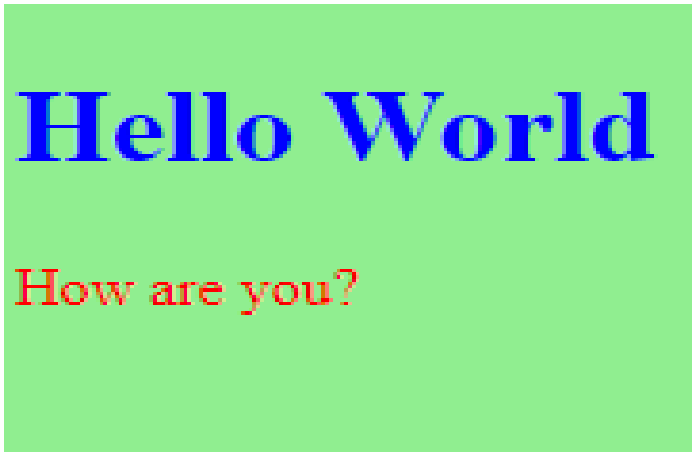
2. Internal CSS

- An internal CSS is used to define a style for a single HTML page.
- An internal CSS is defined in the <head> section of an HTML page, within a <style> element.
- The following example sets the text color of all the <h1> element (on that page) to blue, and the text color of all the <p> element to red.
- In addition, the page will be displayed with a “lightgreen" background color:

```
<!DOCTYPE html>
<html>
<head>
<style>
body {background-color:lightgreen;}
h1   {color: blue;}
p    {color: red;}
</style>
</head>
<body>

<h1>Hello World</h1>
<p>How are you?</p>

</body>
</html>
```



3. External CSS

- An external style sheet is used to define the style for many HTML pages.
- To use an external style sheet, add a link to it in the <head> section of each HTML page:

```
<html>
<head>
<link rel="stylesheet" href="ext.css">
</head>
<body>
<h1>HTML Images</h1>
<p>Images are defined by img tag</p>
</body>
</html>
```

```
body
{
background-color: aqua;
}
h1{
color:blue;
}
p{
color:green;
}
```

HTML Tables

- HTML tables allow web developers to arrange data into rows and columns.

Define an HTML Table

- A table in HTML consists of table cells inside rows and columns.

Table Cells

- Each table cell is defined by a <td> and a </td> tag.
- td : table data.
- Everything between <td> and </td> are the content of the table cell.
- A table cell can contain all sorts of HTML elements:
  text, images, lists, links, other tables, etc.

Table Rows

- Each table row starts with a <tr> and ends with a </tr> tag.
- tr : table row.
- As many rows can be defined in a table; the number of cells are the same in each row.

Table Headers

- Sometimes you want your cells to be table header cells. In those cases use the <th> tag instead of the <td> tag:
- th : table header.
- By default, the text in <th> elements are bold and centered, which can be changed with CSS.

```
<!DOCTYPE html>
<html>
<body><style>
table,th, td {
  border:2px solid red;
}</style>
<h2>A Simple table in HTML</h2>
<table style="width:50%">
  <tr>
    <th>Roll No.</th>
    <th>Name</th>
    <th>Age</th>
  </tr>
  <tr>
    <td>1</td>
    <td>AAA</td>
    <td>15</td>
  </tr>
  <tr>
    <td>2</td>
    <td>BBB</td>
    <td>16</td>
  </tr>
</table>
</body></html>
```

A Simple table in HTML

Roll No.	Name	Age
1	AAA	15
2	BBB	16