

* **When we enter an URL in the browser**

1. Browser checks cache for DNS entry to find the corresponding IP address of website.  
   It looks for following cache. If not found in one, then continues checking to the next until found.
   1. Browser Cache
   2. Operating Systems Cache
   3. Router Cache
   4. ISP Cache
2. If not found in cache, DNS server initiates a DNS query to find IP address of server that hosts the domain name.  
   The requests are sent using small data packets that contain information content of request and IP address it is destined for.
3. Browser initiates a TCP connection with the server using synchronize(SYN) and acknowledge(ACK) messages.
4. Browser sends an HTTP request to the web server.
5. Server on the host computer handles that request and sends back a response. It assembles a response in some format like JSON, XML and HTML.
6. Server sends out an HTTP response along with the status of response.
7. Browser displays HTML content

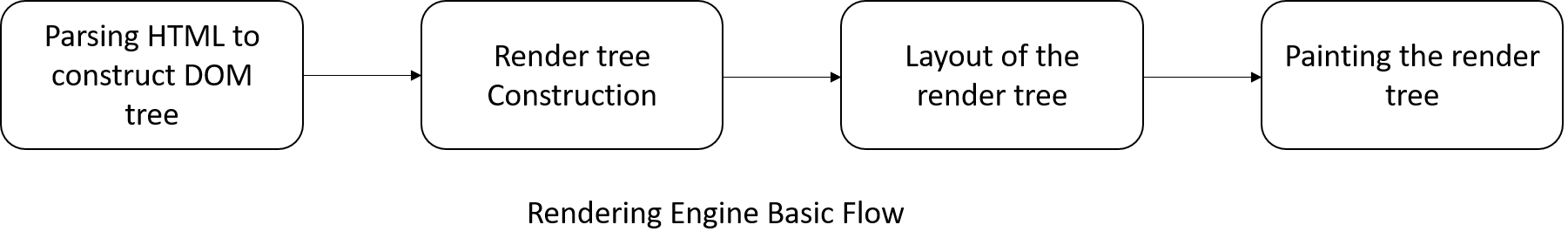
**Browser Components**

1. **The User Interface**: The user interface is the space where User interacts with the browser. It includes the address bar, back and next buttons, home button, refresh and stop, bookmark option, etc. Every other part, except the window where requested web page is displayed, comes under it.
2. **The Browser Engine**: The browser engine works as a bridge between the User interface and the rendering engine. According to the inputs from various user interfaces, it queries and manipulates the rendering engine.
3. **Rendering Engine:**

 The rendering engine, as the name suggests is responsible for rendering the requested web page on the browser screen. The rendering engine interprets the HTML, XML documents and images that are formatted using CSS and generates the layout that is displayed in the User Interface. However, using plugins or extensions, it can display other types data also. Different browsers user different rendering engines:  
\* Internet Explorer: Trident  
\* Firefox & other Mozilla browsers: Gecko  
\* Chrome & Opera 15+: Blink  
\* Chrome (iPhone) & Safari: Webkit

1. **Networking**: Component of the browser which retrieves the URLs using the common internet protocols of HTTP or FTP. The networking component handles all aspects of Internet communication and security. The network component may implement a cache of retrieved documents in order to reduce network traffic.
2. **Data Persistence/Storage:** This is a persistence layer. Browsers support storage mechanisms such as localStorage, IndexedDB, WebSQL and FileSystem. It is a small database created on the local drive of the computer where the browser is installed. It manages user data such as cache, cookies, bookmarks and preferences.

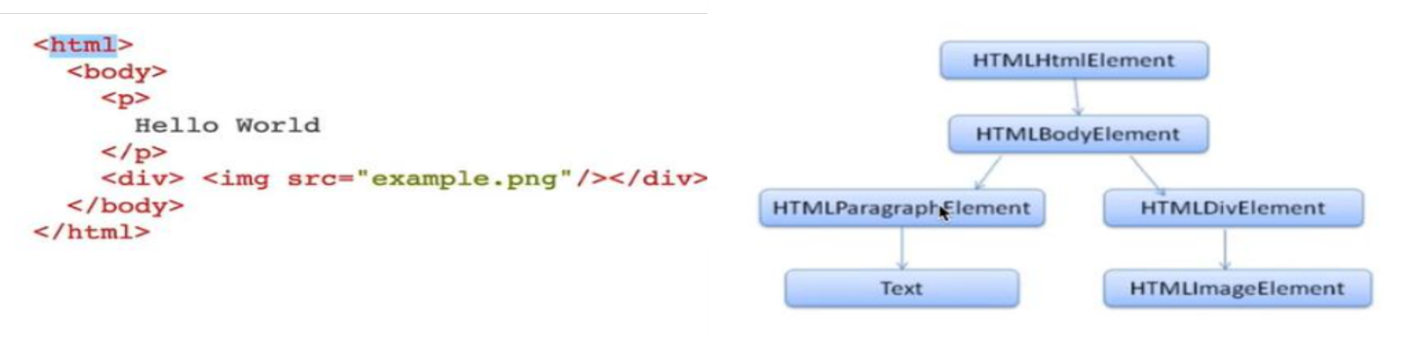
**Rendering Engine:**

**Parser**

The job of the HTML parser is to parse the HTML markup into a parse tree.

**Tree construction:**

When the parser is created the Document object is created. During the tree construction stage the DOM tree with the Document in its root will be modified and elements will be added to it.



Output tree is a tree of DOM element and attribute nodes.

**Layout**: After the construction of the render tree, it goes through a “**layout process**” of the render tree. When the renderer is created and added to the tree, it does not have a position and size. The process of calculating these values is called layout or reflow.

**Painting**: In the painting stage, the render tree is traversed and the renderer’s “paint()” method is called to display content on the screen. Painting uses the UI backend layer.