**How Does The Browser Actually Render A Website**

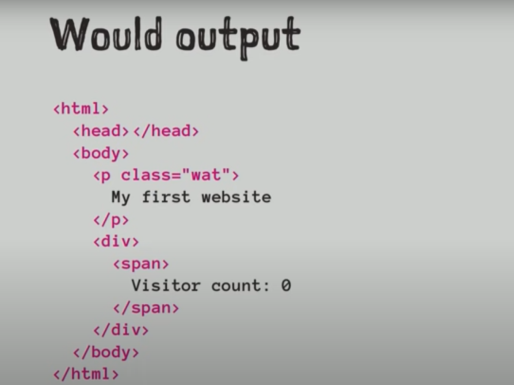
**Browser consists of**

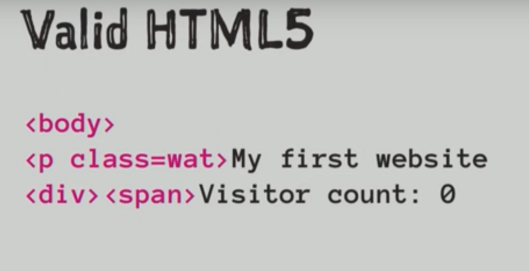
* Binding
* Rendering
  + Parsing
  + Layout
  + Painting
* Platform
* JavaScript VM

**High Level flow**

**HTML Parsing** - Creates DOM Tree

* HTML is forgiving by nature





**Parsing Flow**

* ***Tokeniser*** – Takes text and converts into Tokens
* ***Tree Constructor (Parse Tree)*** - One to one representation of HTML
* ***DOM Tree* –** interacting between Javascript in a page
* ***Script Execution* –** JS can interact with HTML and CSS and can alter the page

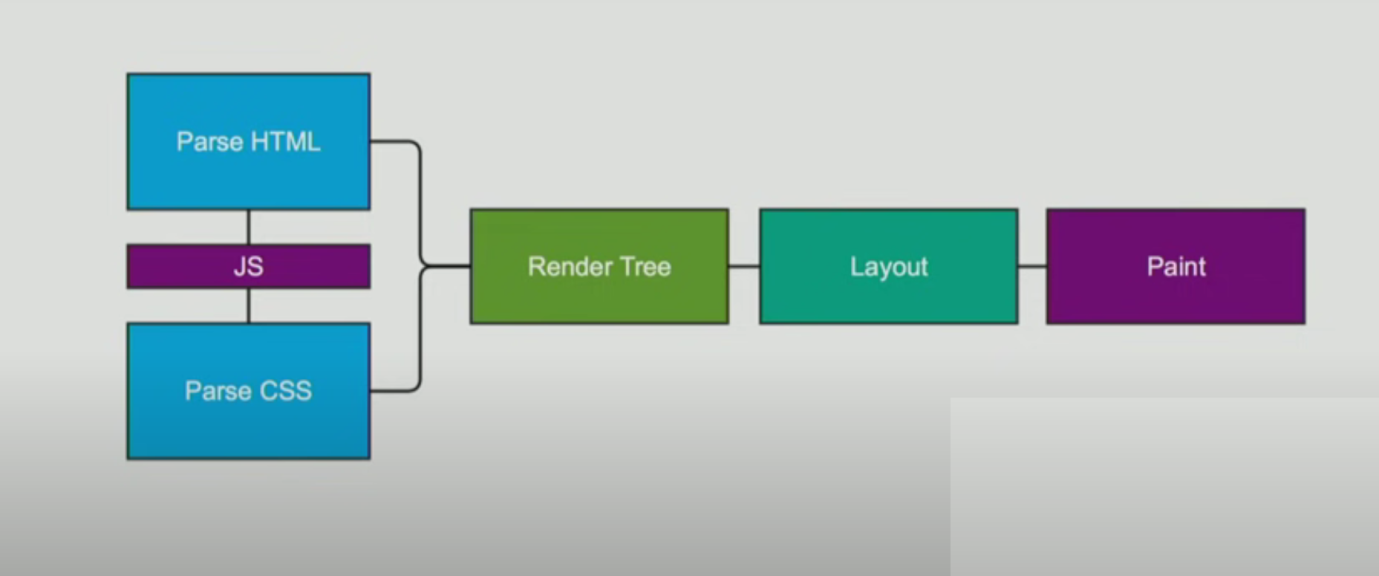
**HTML parsing** - can be halted

* <script>,<link> and <style> tags can stop HTML execution as they can alter the webpage

**Speculative parsing -** HTML parsing Will look ahead for external images, scripts, styles etc.

**Reentrant –** means HTML parsing can be interrupted by js

**CSS Paring –** Straight Forward parsing – Creates CSSOM



**Rendering or Frame Tree**

* Combination of Object Module and Style Resolution **(DOM +CSSOM)**
* This is the actual representation of what we see on the screen
* There are multiple tress in Rendering tree
  + **Render Object –** DOM Node itself
  + **Render style –** Style applied to DOM elements
  + **Render layers** – Placing elements on a page
  + **Line box** – text and its styles
* Not in render tree
  + Non visible elements – head tag, style, title etc
  + Nodes hidden by *display:none*

**DOM nodes to render Nodes :** DOM nodes are converted to render objects

* **It Consists of**
* Visual Outputs
* Geometric info
* Layout and paint
* Hold style and computed metrics

**Calculating visual properties**

* Combines all styles
* Default, external, style elements and inline
* Style computation

**Layout**

* Recursive process
* Traverse Render tree
* Layouts its child
* **Will Batch Layout** – render tree will flag itself as dirty if there is any change in DOM and the Browser will change it works as asynchronously
* **Immediate layer –** change in font size or window resize

**Paint –** will create a bitmap from render tree

**Performance Insights**

* Putting <script>,<link> and <style> tags at the bottom prevents parsing getting interrupted in middle
* Also helps in faster rendering
* Read in one go and write in one go