1. **Does localStorage throw error after reaches maximum limits?**

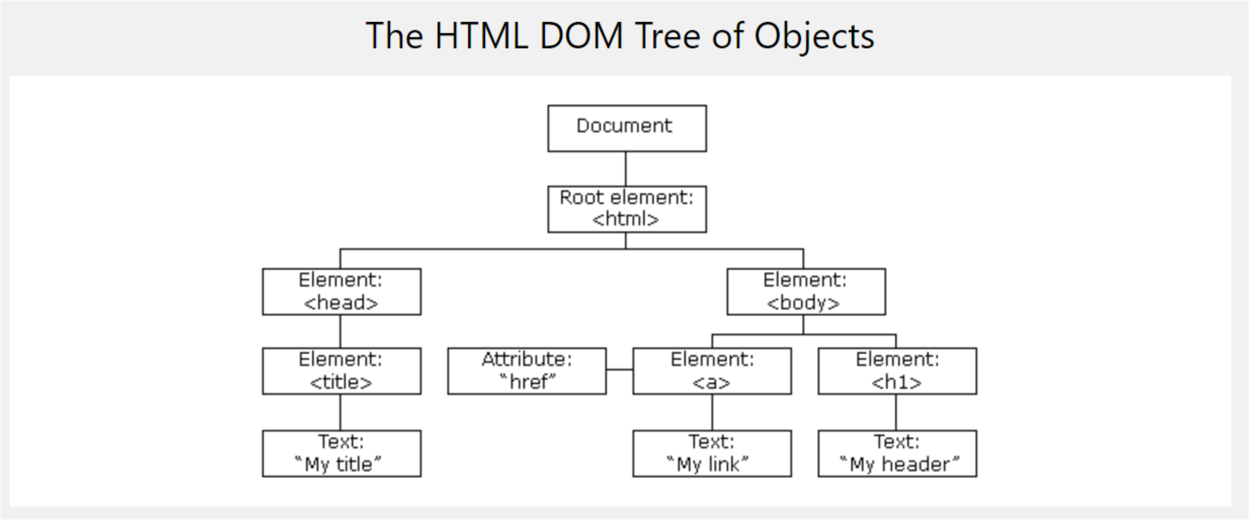
   Yes, it will give a QuotaExceeded Error that says: Failed to execute ‘setItem’ in ‘Storage: Setting the value of “key” exceeded the quota.

2. **What are the new form elements in HTML5?**

   <datalist> <keygen><output>

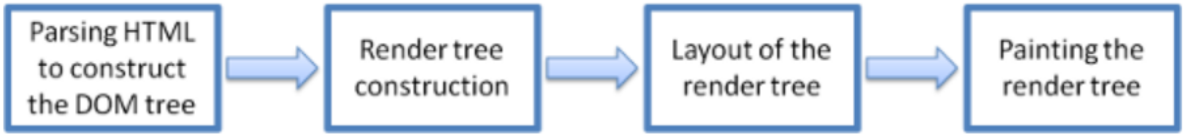
3. **What is the DOM? How does the DOM work?**

   Document Object Model(DOM) is a programming interface for HTML and XML documents. It represents the page so that programs can change the document structure, style, and content. The DOM represents the document as nodes and objects. That way, programming languages can connect to the page.



4. **How does the browser rendering engine work?**

   The rendering engine parses the chunks of HTML documents and converts the elements to DOM nodes in a tree called the “content tree” or the “DOM tree”. It also parses both the external CSS files as well as style elements.


Parsing HTML to construct the DOM tree
Render tree construction
Layout of the render tree
Painting the render tree

Rendering engine basic flow

5. **What does a <DOCTYPE html> do?**

   Doctype stands for Document Type Declaration. It is not a tag but a statement. It informs the web browser about the type and version of HTML used in building the web document. This helps the browser to handle and load it properly. While the HTML syntax for this statement is somewhat simple, you must note each version of HTML has its own rules.

6. **What happens when DOCTYPE is not given?**
   a. The browser enters Quirks mode.
   b. New features in HTML5 like <header>, <nav> will not be supported.
   c. Case insensitive for CSS classname/id/name.

7. **What is the difference between standards mode and quirks mode?**

When the web standards were made at W3C, browsers could not just start using them, as doing so would break most existing sites on the web. Browsers therefore introduced two modes to treat new standards compliant sites differently from old legacy sites.

In quirks mode, layout emulates nonstandard behavior in Navigator 4 and Internet Explorer 5. This is essential in order to support websites that were built before the widespread adoption of web standards.

In standards mode, the behavior is (hopefully) the behavior described by the HTML and CSS specifications.

8. **What is the difference between HTML and XHTML?**

HTML and XHTML are both languages in which web pages are written. XHTML is a stricter, more XML-based version of HTML.

The most important differences from HTML:

   a. <!DOCTYPE> is mandatory
   b. The xmlns attribute in <html> is mandatory
   c. <html>, <head>, <title>, and <body> are mandatory
   d. Elements must always be properly nested
   e. Elements must always be closed
   f. Elements must always be in lowercase
   g. Attribute names must always be in lowercase
   h. Attribute values must always be quoted
   i. Attribute minimization is forbidden

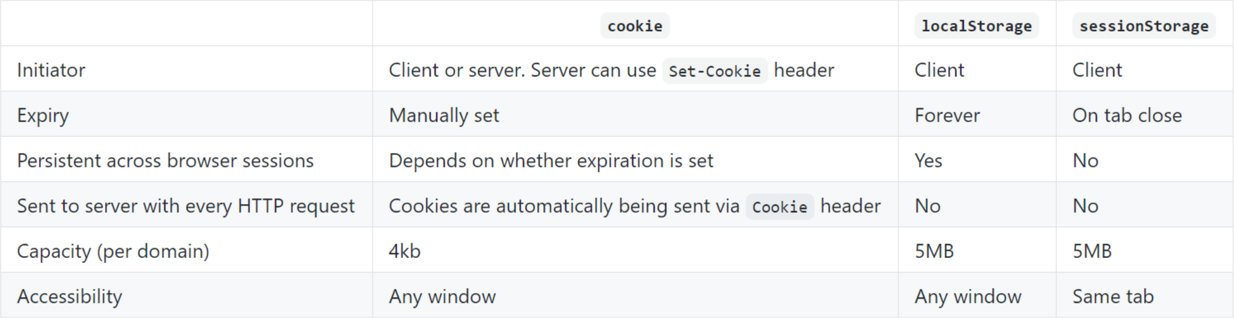
9. **What are the building blocks of HTML5?**

   a. Semantics: allowing you to describe more precisely what your content is.
   b. Connectivity: allowing you to communicate with the server in new and innovative ways.
   c. Offline and storage: allowing webpages to store data on the client-side locally and operate offline more efficiently.
   d. Multimedia: making video and audio first-class citizens in the Open Web.
   e. 2D/3D graphics and effects: allowing a much more diverse range of presentation options.
   f. Performance and integration: providing greater speed optimization and better usage of computer hardware.
   g. Device access: allowing for the usage of various input and output devices.
   h. Styling: letting authors write more sophisticated themes.

Reference: MDN

10. **Describe the difference between a cookie, sessionStorage, and localStorage?**

All the technologies are key-value storage mechanisms on the client side. They are only able to store values as strings.



	`cookie`	`localStorage`	`sessionStorage`
Initiator	Client or server. Server can use `Set-Cookie` header	Client	Client
Expiry	Manually set	Forever	On tab close
Persistent across browser sessions	Depends on whether expiration is set	Yes	No
Sent to server with every HTTP request	Cookies are automatically being sent via `Cookie` header	No	No
Capacity (per domain)	4kb	5MB	5MB
Accessibility	Any window	Any window	Same tab

11. **What is the Critical Rendering Path?**

The Critical Rendering Path is the sequence of steps the browser goes through to convert the HTML, CSS, and JavaScript into pixels on the screen. Optimizing the critical render path improves render performance.The critical rendering path includes the Document Object Model (DOM), CSS Object Model (CSSOM), render tree and layout.

Reference: Critical rendering path, Understanding the Critical Rendering Path

12. **What are the Benefits of Server Side Rendering (SSR) Over Client Side Rendering (CSR)?**

Client-Side Rendering will download a minimal HTML page, then render the JavaScript and fills the content into it. Server-side rendering, on the other hand, renders JS on the server. In this case, the output is HTML already filled with content, so there is no need to fetch any JS.

The benefits for SSR are:

a. Performance – the loaded app is sent down from the server on the first request.
b. Search Engine Optimization – Google favors sites with fast load times.

13. **What is the difference between a <span> and a <div>?**

<span>

- Inline element.
- Wrap small portions of text, images, etc.
- Used to stylize texts.

<div>

- Block-level element.
- Used to wrap sections of a document.
- Used while creating CSS based layouts in html.

14. **Name 5 common block-level and inline HTML elements?**

<div> <p> <h1> <form> <ul> <li>

15. **What are semantic and non-semantic elements?**

Semantic elements are elements we could under stand what they are from the tag name. Like <table>, <form>. While non-semantic elements are not, like <div>, <span>.

16. **What is the purpose of the main element?**

The <main> element represents the main content section of the body of a document or application. The <main> content section of a document includes content that is unique to that document and excludes content that is repeated across a set of documents such as site navigation links, copyright information, site logos and banners and search forms (unless the document or applications main function is that of a search form).

17. **Define semantic markup. What are the semantic meanings for <section>, <article>, <aside>, <nav>, <header>, <footer> and when/how should each be used in structuring html markup?**
    - <section>
      Represents a generic standalone section of a document, which doesn't have a more specific semantic element to represent it. Sections should always have a heading, with very few exceptions.
    - <article>
      Represents a self-contained composition in a document, page, application, or site, which is intended to be independently distributable or reusable (e.g., in syndication).
    - <aside>
      Represents a portion of a document whose content is only indirectly related to the document's main content. Asides are frequently presented as sidebars or call-out boxes.
    - <nav>
      Represents a section of a page whose purpose is to provide navigation links, either within the current document or to other documents. Common examples of navigation sections are menus, tables of contents, and indexes.
    - <header>
      Represents introductory content, typically a group of introductory or navigational aids. It may contain some heading elements but also a logo, a search form, an author name, and other elements.
    - <footer>
      Represents a footer for its nearest sectioning content or sectioning root element. A <footer> typically contains information about the author of the section, copyright data or links to related documents.

    Reference: Using HTML sections and outlines
18. **When should you use section, div, or article?**

    If the content within the element is not semantically related, then use a <div>. If the semantically related content is also able to be self-contained, then use an <article>. Otherwise, use a <section>.

    Reference: Sectioning Content in HTML5 - div or section or article?
19. **What is Character Encoding?**

    A character encoding tells the computer how to interpret raw zeroes and ones into real characters. It usually does this by pairing numbers with characters. Words and sentences in text are created from characters and these characters are grouped into a character set. There are many different types of character encodings floating around at present, but the ones we deal most frequently with are ASCII, 8-bit encodings, and Unicode-based encodings.

    Reference: What is character encoding
20. **What is the purpose of meta tags?**

Meta tags provide information about the webpage in the HTML of the document. This information is called "metadata" and while it is not displayed on the page itself, it can be read by search engines and web crawlers.

21. **What do async and defer refer to in a script tag? Describe the difference between <script>, <script async> and <script defer>**

    The <script> element has two attributes, async and defer, that can give us more control over how and when external files are fetched and executed.

    - <script>
      HTML parsing is blocked, the script is fetched and executed immediately, HTML parsing resumes after the script is executed.
    - <script async>
      The script will be fetched in parallel to HTML parsing and executed as soon as it is available (potentially before HTML parsing completes). Use async when the script is independent of any other scripts on the page, for example, analytics.
    - <script defer>
      The script will be fetched in parallel to HTML parsing and executed when the page has finished parsing. If there are multiple of them, each deferred script is executed in the order they were encountered in the document. If a script relies on a fully-parsed DOM, the defer attribute will be useful in ensuring that the HTML is fully parsed before executing. There's not much difference in putting a normal <script> at the end of <body>. A deferred script must not contain document.write.
      Reference: Github

22. **Can you describe the difference between progressive enhancement and graceful degradation?**

    Progressive enhancement is a more sophisticated and at the same time stable way of assuring that but it takes more time and effort. Graceful degradation can be used more easily as a patch for an already existing product; it means harder maintenance later on, but requires less initial work.

    Reference: W3

23. **What is the purpose of cache-busting and how can you achieve it?**

Cache busting solves the browser caching issue by using a unique file version identifier to tell the browser that a new version of the file is available. Therefore the browser doesn't retrieve the old file from cache but rather makes a request to the origin server for the new file.

Reference: Cache busting

24. **Name 3 ways to decrease page load?**

    - Choose a performance-optimized hosting solution.
    - Compress and optimize your images.
    - Reduce your redirects.

    Reference:https://blog.hubspot.com/marketing/how-to-reduce-your-websites-page-speed

25. **What ARIA and screenreaders are, and how to make a website accessible?**

Accessible Rich Internet Applications (ARIA) is a set of attributes that define ways to make web content and web applications (especially those developed with JavaScript) more accessible to people with disabilities.

Reference: Accessibility|MDN, How to Optimize Your Website for Disabled Users

26. What is the purpose of the alt attribute on images?

The ALT text adds a text description to an image on a Web page, and should be used for all images, graphical bullets, and graphical horizontal rules. ALT text is accessed by screen reader users to provide them with a text equivalent of images.

27. Explain some of the pros and cons of CSS animations versus JavaScript animations?

CSS is more simple than JavaScript. But JS has significantly more power than CSS.

28. What does CORS stand for and what issue does it address?

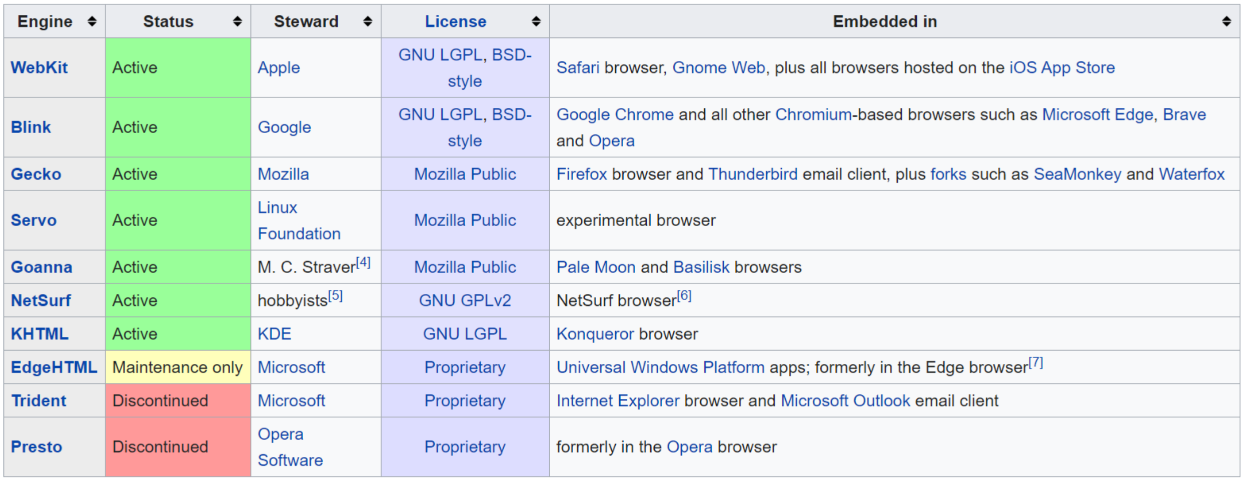
Cross-Origin Resource Sharing (CORS) is an HTTP-header based mechanism that allows a server to indicate any other origins (domain, scheme, or port) than its own from which a browser should permit loading of resources. CORS also relies on a mechanism by which browsers make a “preflight” request to the server hosting the cross-origin resource, in order to check that the server will permit the actual request. In that preflight, the browser sends headers that indicate the HTTP method and headers that will be used in the actual request.

29. Ways to improve website performance

- SEO Best Practices
- Enable compression
- Minify CSS, JavaScript, and HTML
- Reduce redirects
- Remove render-blocking JavaScript

Reference: https://moz.com/learn/seo/page-speed

30. Comparison of browser engines like Chrome, Firefox, Internet Explorer, Safari?



Engine	Status	Steward	License	Embedded in
WebKit	Active	Apple	GNU LGPL, BSD-style	Safari browser, Gnome Web, plus all browsers hosted on the iOS App Store
Blink	Active	Google	GNU LGPL, BSD-style	Google Chrome and all other Chromium-based browsers such as Microsoft Edge, Brave and Opera
Gecko	Active	Mozilla	Mozilla Public	Firefox browser and Thunderbird email client, plus forks such as SeaMonkey and Waterfox
Servo	Active	Linux Foundation	Mozilla Public	experimental browser
Goanna	Active	M. C. Straver[4]	Mozilla Public	Pale Moon and Basilisk browsers
NetSurf	Active	hobbyists[5]	GNU GPLv2	NetSurf browser[6]
KHTML	Active	KDE	GNU LGPL	Konqueror browser
EdgeHTML	Maintenance only	Microsoft	Proprietary	Universal Windows Platform apps; formerly in the Edge browser[7]
Trident	Discontinued	Microsoft	Proprietary	Internet Explorer browser and Microsoft Outlook email client
Presto	Discontinued	Opera Software	Proprietary	formerly in the Opera browser

Reference: https://en.wikipedia.org/wiki/Comparison_of_browser_engines

31. What does the lang attribute in HTML do?

The lang (or sometimes the xml:lang) attribute specifies the natural language of the content of a web page. An attribute on the html tag sets the language for all the text on the page.

32. What is the desktop first and mobile-first design approach?

Desktop first designing is for full sized screens means building for the highest specs to display and communicate as much as possible. In this design type it's not the mobile but the desktop experience that needs to be detailed and dynamic.

Mobile first is designed mainly for functionality. It's a very focused approach starting with putting out a core function and layering the extras incrementally for higher resolutions (often called progressive enhancement). The mobile-first strategy is often referred to as a content-first strategy because your website's content must be carefully researched and structured to be categorized based on importance. This method can be much more expensive due to the more extended preparation phase and the expertise needed to create the platform's versatility.

Reference: D VS M

33. **What are data- attributes good for?**

    Before JavaScript frameworks became popular, front end developers used data-attributes to store extra data within the DOM itself, without other hacks such as non-standard attributes, extra properties on the DOM. It is intended to store custom data private to the page or application, for which there are no more appropriate attributes or elements.

34. **Explain the difference between layout, painting, and compositing?**
    - Layout:
      Once browser know which rules to apply to an element it will determine how much space each element takes up and where to place it
    - Painting:
      This is the process of filling in pixels. It involves drawing out elements on multiple layers
    - Compositing:
      Browser draws the different layer to the screen in the correct order so the page renders correctly

35. **Explain about HTML Canvas?**

    <canvas> is an HTML element which can be used to draw graphics via scripting (usually JavaScript). This can, for instance, be used to draw graphs, combine photos, or create simple (and not so simple) animations.

36. **Explain about HTML Layout Engines used by browsers?**

    A browser engine (also known as a layout engine or rendering engine) is a core software component of every major web browser. The primary job of a browser engine is to transform HTML documents and other resources of a web page into an interactive visual representation on a user's device.

37. **What are the semantic tags available in html5?**

    <article><aside><details><figcaption><figure><footer><header><main><mark><nav><section><summary><time>

38. **Why you would like to use a semantic tag?**

    More easy for other developers to understand.

39. **How to make a page responsive?**

    To create a responsive website, add the following <meta> tag to set the viewpoint:<meta name="viewport" content="width=device-width, initial-scale=1.0">

40. **What is the difference between span tag and div tag?**

<span>

- Inline element.
- Wrap small portions of text, images, etc.
- Used to stylize texts.

<div>

- Block-level element.
- Used to wrap sections of a document.
- Used while creating CSS based layouts in html.

41. What are optional closing tags?

<html><head><body><p><li><dt><dd><option><thead><th><tbody><tr><td><tfoot><colgroup>

42. What is a self-closing tag?

A self-closing tag is a shorthand notation for an opening and closing tag in one. It's used to communicate lack of content in between the opening and closing tags. So, rather than typing <p></p> (with no space at all in between), you'd be able write <p/>.

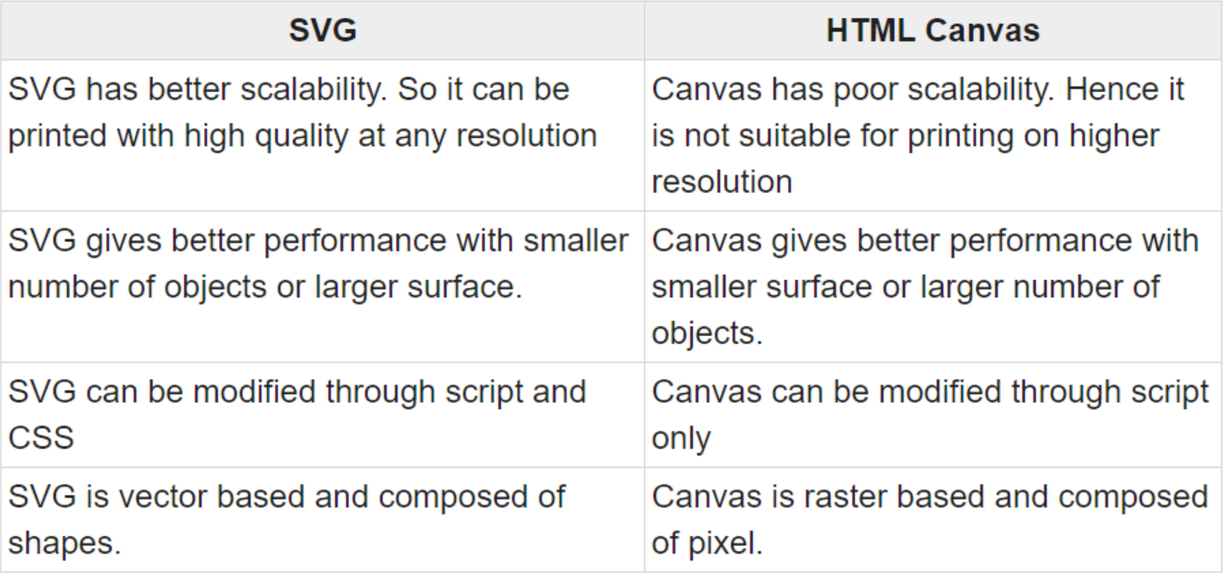
43. Does the following trigger HTTP request at the time of page load?

Check: https://www.w3.org/TR/REC-html40/index/attributes.html

44. How Geolocation API works in html5?

HTML5 geolocation detects latitude and longitude coordinates by using the device's GPS (if available on the device) or the device's mobile/WIFI signal (if GPS is not available. The mobile/WIFI signals are triangulated to work out the latitude and longitude.

45. What is the difference between SVG and Canvas?



SVG	HTML Canvas
SVG has better scalability. So it can be printed with high quality at any resolution	Canvas has poor scalability. Hence it is not suitable for printing on higher resolution
SVG gives better performance with smaller number of objects or larger surface.	Canvas gives better performance with smaller surface or larger number of objects.
SVG can be modified through script and CSS	Canvas can be modified through script only
SVG is vector based and composed of shapes.	Canvas is raster based and composed of pixel.

46. Explain Drag and Drop in HTML5?

Drag and Drop (DnD) is powerful User Interface concept which makes it easy to copy, reorder and deletion of items with the help of mouse clicks. This allows the user to click and hold the mouse button down over an element, drag it to another location, and release the mouse button to drop the element there.

47. **Why use IndexedDB instead of WebSQL in HTML5?**

    IndexedDB is the successor to both LocalStorage and WebSQL, designed to replace them as the “one true” browser database. It exposes an asynchronous API that supposedly avoids blocking the DOM, but as we’ll see below, it doesn’t necessarily live up to the hype. Browser support is extremely spotty, with only Chrome and Firefox having fully usable implementations.

48. **Explain Application Cache in HTML5. OR What is a manifest file in HTML?**

    The manifest file is a simple text file, which tells the browser what to cache (and what to never cache).

49. **Explain Microdata in HTML5?**

    Microdata is a standardized way to provide additional semantics in your web pages.

50. **List the API available in HTML5?**

    Geolocation, drag and drop, web storage,Server-Sent Events

51. **What are the different new form element types provided by HTML5?**

    <datalist> <keygen><output>

52. **What are the HTML tags which deprecated in HTML5?**

    Some attributes from HTML4 are no longer allowed in HTML5 at all and they have been removed completely. img and iframe. caption, iframe, img, input, object, legend, table, hr, div, h1, h2, h3, h4, h5, h6, p, col, colgroup, tbody, td, tfoot, th, thead and tr. table, tr, td, th and body.

53. **How you can Use Modernizr in HTML5?**

    Modernizr is a JavaScript library that detects which HTML5 and CSS3 features your visitor's browser supports. In detecting feature support, it allows developers to test for some of the new technologies and then provide fallbacks for browsers that do not support them.

    Use <script src="modernizr.js"></script> and Modernizr.canvas

54. **What is the use of WebSocket API?**

    The WebSocket API is an advanced technology that makes it possible to open a two-way interactive communication session between the user's browser and a server. With this API, you can send messages to a server and receive event-driven responses without having to poll the server for a reply.

55. **What does enctype='multipart/form-data’ mean?**

    It is an encoding type that allows files to be sent through a POST. Quite simply, without this encoding the files cannot be sent through POST. If you want to allow a user to upload a file via a form, you must use this enctype.

56. **What is progressive rendering?**

    Progressive rendering is the name given to techniques used to improve the performance of a webpage (in particular, improve perceived load time) to render content for display as quickly as possible.

    It used to be much more prevalent in the days before broadband internet but it is still used in modern development as mobile data connections are becoming increasingly popular (and unreliable)!

57. **What is the difference between Select and Datalist?**

For the select element, the user is required to select one of the options you've given. For the datalist element, it is suggested that the user select one of the options you've given, but he can actually enter anything he wants in the input.

58. **What are data- attributes good for?**

    data-* attributes allow us to store extra information on standard, semantic HTML elements without other hacks such as non-standard attributes, extra properties on DOM, or Node.setUserData().