# Career Services Assignment 9 – API Flash Cards

**Points possible:** 50

|  |  |  |
| --- | --- | --- |
| Category | Criteria | % of Grade |
| Completeness | All requirements of the assignment are complete. | 100 |

**Instructions:** Research common interview questions online revolving around **HTML, CSS, and AJAX** and create 20 flash cards from the information you find. Study your flash cards regularly to better prepare for interviews. Fill out the table below with the information you put on each of your flash cards.

|  |  |
| --- | --- |
| **Front of Card** | **Back of Card** |
| 1. What is AJAX and explain about it? | AJAX (Asynchronous JavaScript and XML) is a group of technologies used to load the page content in the background without disturbing or changing the existing state of the web page. DOM (Document Object Model) API (Application Programming Interface) in the browser tree can be manipulated to provide the dynamic functionalities or to change the content of the web page dynamically. Asynchronous communication is another great feature of the AJAX and that can be achieved by using XMLHTTPRequest object which is an API present in the form of an object which a standard is given by WHATWG (Web Hypertext Application Technology Working Group) and was later moved to W3C (World Wide Web Consortium). |
| 2. What are the different technologies used in AJAX? | AJAX involves with different technologies such as HTML (and XHTML), CSS (for presentation layer), JSON, XML or XSLT (for data exchange between a web server and client), JavaScript and other APIs for asynchronous communication. [AJAX](https://www.educba.com/what-is-ajax/) is used for creating dynamic web pages. The examples of the websites those used AJAX are Google, Gmail, YouTube, Facebook etc. AJAX is a standard based on Internet Standards to comply with all the browsers and networks across the globe. |
| 3. What is a synchronous request in AJAX? | The synchronous and Asynchronous type requests in AJAX are used based on the request priority of the web server request and should be carefully configured to make the server respond to the user based on the user requirement. The synchronous request waits for the server’s response after making a request to proceed with next part of the script execution which is crucial in fast response mechanism and this should be avoided frequent times to make server high responsive. Wherever the synchronous request mechanism is essential or inevitable, there should it be used. |
| 4. What is asynchronous request in AJAX? | An Asynchronous request is defined as the script execution which allows proceeding with the next line of execution irrespective of the response from the web server after a request, which can be handled later upon the receipt of the response from web server. The asynchronous is the most import mechanism in providing the responses to the user without any delay where a user can still interact with the web pages irrespective of the responses from the server for the previous requests. |
| 5. What are different ready states in AJAX? | A ready state is a property present in the XMLHttpRequest object which holds the status of the XMLHTTPRequest. The different types of the status changes are available from 0 to 4 which are defined as 0 (Request not initialized), 1 (Established connection with Server), 2 (Request Received), 3 (Processing request) 4 (Request finished and a response is ready). When the status is 4, then the status can be described as HTTP status 200 (OK) which means the request is success and response is ready |
| 6. What are the different stages and processes in AJAX ready states? | The different states and its processes are initialization, requesting, processing and finished in AJAX. When an open method is called on an object, then the status changes to 1. When a send method is called then status changes to 2. After receiving the response from the server, when a request ends, the status changes to 4. |
| 7. What is XMLHTTPRequest Object? | XMLHTTPRequest (XHR) is a browser level API (Application Programming Interface) which processes the server-client requests using JavaScript. The XMLHTTPRequest object is present in the form of API to establish communication between a server and the browser client. XMLHTTPRequest is used to update the page content without reloading, request and receive data responses from the server even after completely loading the page, request and receive the responses from the server in the background without any interruptions to the user. |
| 8. What is Semantic HTML? | Semantic HTML is a coding style. It is the use of HTML markup to reinforce the semantics or meaning of the content. For example: In semantic HTML tag is not used for bold statement as well as tag is used for italic. Instead of these we use and tags. |
| 9. How to create a nested webpage in HTML? [↑](https://www.adaface.com/blog/html-css-interview-questions/#questionsindex) | The HTML iframe tag is used to display a nested webpage. In other words, it represents a webpage within a webpage. The HTML <iframe> tag defines an inline frame. |
| 10. Why is a URL encoded in HTML? [↑](https://www.adaface.com/blog/html-css-interview-questions/#questionsindex) | An URL is encoded to convert non-ASCII characters into a format that can be used over the Internet because a URL is sent over the Internet by using the ASCII character-set only. If a URL contains characters outside the ASCII set, the URL has to be converted. The non-ASCII characters are replaced with a "%" followed by hexadecimal digits. |
| 11. What is SVG? [↑](https://www.adaface.com/blog/html-css-interview-questions/#questionsindex) | HTML SVG is used to describe the two-dimensional vector and vector/raster graphics. SVG images and their behaviors are defined in XML text files. So as XML files, you can create and edit an SVG image with the text editor. It is mostly used for vector type diagrams like pie charts, 2-Dimensional graphs in an X, Y coordinate system. |
| 12. What is the difference between DIV and SPAN in HTML? [↑](https://www.adaface.com/blog/html-css-interview-questions/#questionsindex) | The difference between span and div is that a span element is in-line and usually used for a small chunk of HTML inside a line,such as inside a paragraph. Whereas, a div or division element is block-line which is equivalent to having a line-break before and after it and used to group larger chunks of code. |
| 13. Is there any way to keep list elements straight in an HTML file? [↑](https://www.adaface.com/blog/html-css-interview-questions/#questionsindex) | By using indents, you can keep the list elements straight. If you indent each sub nested list in further than the parent list, you can easily determine the various lists and the elements that it contains. |
| 14. How Do I Design For Backward Compatibility Using Style Sheets? [↑](https://www.adaface.com/blog/html-css-interview-questions/#questionsindex) | Existing HTML style methods (such as <font SIZE> and <b>) may be easily combined with style sheet specification methods. Browsers that do not understand style sheets will use the older HTML formatting methods, and style sheets specifications can control the appearance of these elements in browsers that support CSS1. |
| 15. What Can Be Done With Style Sheets That Can Not Be Accomplished With Regular Html? [↑](https://www.adaface.com/blog/html-css-interview-questions/#questionsindex) | Many of the recent extensions to HTML have been tentative and somewhat crude attempts to control document layout. Style sheets go several steps beyond, and introduces complex border, margin and spacing control to most HTML elements. It also extends the capabilities introduced by most of the existing HTML browser extensions. Background colors or images can now be assigned to ANY HTML element instead of just the BODY element and borders can now be applied to any element instead of just to tables. For more information on the possible properties in CSS, see the Index DOT Css Property Index. |
| 16. Why Shouldn't I Use Fixed Sized Fonts ? [↑](https://www.adaface.com/blog/html-css-interview-questions/#questionsindex) | Only in very rare situations we will find users that have a "calibrated" rendering device that shows fixed font sizes correct. This tells us that we can never know the real size of a font when it's rendered on the user end. Other people may find your choice of font size uncomfortable. A surprisingly large number of people have vision problems and require larger text than the average. Other people have good eyesight and prefer the advantage of more text on the screen that a smaller font size allows. What is comfortable to you on your system may be uncomfortable to someone else. Browsers have a default size for fonts. If a user finds this inappropriate, they can change it to something they prefer. You can never assume that your choice is better for them. So, leave the font size alone for the majority of your text. If you wish to change it in specific places (say smaller text for a copyright notice at the bottom of page), use relative units so that the size will stay in relationship to what the user may have selected already. Remember, if people find your text uncomfortable, they will not bother struggling with your web site. Very few (if any) web sites are important enough to the average user to justify fighting with the author's idea of what is best. |
| 17. What are the benefits of CSS sprites? [↑](https://www.adaface.com/blog/html-css-interview-questions/#questionsindex) | If a web page has large no. of images that takes a longer time to load because each image separately sends out an HTTP request. The concept of CSS sprites is used to reduce the loading time for a web page because it combines the various small images into one image. It reduces the number of HTTP requests and hence the loading time.  CSS sprites combine multiple images into one image, limiting the number of HTTP requests a browser has to make, thus improving load times. Even under the new HTTP/2 protocol, this remains true.  Under HTTP/1.1, at most one request is allowed per TCP connection. With HTTP/1.1, modern browsers open multiple parallel connections (between 2 to 8) but it is limited. With HTTP/2, all requests between the browser and the server are multiplexed on a single TCP connection. This means the cost of opening and closing multiple connections is mitigated, resulting in a better usage of the TCP connection and limits the impact of latency between the client and server. It could then become possible to load tens of images in parallel on the same TCP connection.  However, according to benchmark results, although HTTP/2 offers 50% improvement over HTTP/1.1, in most cases the sprite set is still faster to load than individual images.  To utilize a spritesheet in CSS, one would use certain properties, such as background-image, background-position and background-size to ultimately alter the background of an element. |
| 18. What is tweening? [↑](https://www.adaface.com/blog/html-css-interview-questions/#questionsindex) | It is the process of generating intermediate frames between two images. It gives the impression that the first image has smoothly evolved into the second one. It is an important method used in all types of animations. In CSS3, Transforms (matrix, translate, rotate, scale etc.) module can be used to achieve tweening. |
| 19. What are the merits and demerits of embedded style sheets? [↑](https://www.adaface.com/blog/html-css-interview-questions/#questionsindex)  . | This is the basic CSS3 interview questions that have been frequently asked in an interview. Following is the merit and demerit are as follows:  **Merits:**   * Multiple tag types can be created in a single document. * Styles, in a complex situation, can be applied by using the Selector and grouping methods. * Extra download in unnecessary.   **Demerits:**   * Multiple documents cannot be controlled. |
| 20. What are the advantages and disadvantages of using a CSS pre-processor? | The usage depends on the type of project but the following advantages/disadvantages come with a preprocessor.  Advantages:   * CSS is made more maintainable. * Easy to write nested selectors. * Variables for consistent theming. Can share theme files across different projects. * Mixins to generate repeated CSS. * Sass features like loops, lists, and maps can make configuration easier and less verbose. * Splitting your code into multiple files. CSS files can be split up too but doing so will require an HTTP request to download each CSS file.   Disadvantages:   * Requires tools for preprocessing. Re-compilation time can be slow. * Not writing currently and potentially usable CSS. For example, by using something like postcss-loader with webpack, you can write potentially future-compatible CSS, allowing you to use things like CSS variables instead of Sass variables. Thus, you’re learning new skills that could pay off if/when they become standardized. |