01 - Assignment One

Web Development Ecosystem

***Complete the tasks below. If you need help, please use the learning content provided for each topic. Have fun!***

## HTTPS, HTTP, HTTP/2

### [ Learn: [1](https://codedelegance.com/blog/2017/07/02/Vanilla-Node-js-Web-Server-1/), [2](https://kinsta.com/learn/what-is-http2/), [3](https://developers.google.com/web/fundamentals/performance/http2), [4](https://frontendmasters.com/books/front-end-handbook/2019/#4) ]

1. In the following you can find the content of an HTTP Request. Answer to the following questions, indicating where (e.g., in which field) in the HTTP Request you can find the answer:  
    *GET /martignon/index.html HTTP/1.1*

*Host: cs.unibg.it*

*User Agent: Mozilla/5.0 (Macintosh; U; PPC Mac OS X; en) AppleWebKit/124*

*(KHTML, like Gecko) Safari/125*

*Accept: ext/xml, application/xml, application/xhtml+xml, text/html;q=0.9,*

*text/plain;q=0.8, image/png,\*,\*;q=0.5*

*Accept-Language: it*

*Keep-Alive: 300*

*Connection: keep-alive*

* 1. What is the requested URL? **cs.unibg.it/martignon/index.html (Host & GET)**
  2. Which version of HTTP is used? **1.1 (GET HTTP/1.1)**
  3. Does the browser ask for a persistent or a non-persistent connection? **Persistent (Connection: keep-alive)**
  4. What is, in your opinion, the utility in indicating the type (and version) of browser used by the client in the HTTP Request? **Browser type (and version) can differentiate the client experience based on desktop vs. mobile browsers. Keeping track of version can also assist with troubleshooting and debugging.**

1. An HTTP client sends the following message:  
    *GET http://cs.unibg.it /index.html HTTP/1.1*

*User-agent: Mozilla/4.0*

*Accept: text/html, image/gif, image/jpeg*

*If-modified-since: 27 Feb 2017 08:10:00*

* 1. Write down two feasible responses of the HTTP server (only the status line)
     1. **200 OK or 304 Not Modified**
  2. Assuming that the message is sent through a Proxy, specify the behavior of the Proxy itself. **The Proxy checks to see if the website has ever been accessed by the client. If it hasn’t been accessed, it will forward the message to the server. If it has been accessed, it will check to see if it was after 27 Feb 2017 08:10:00. If it was after the date/time, it will forward the message to the server. If it was before the date/time, it will access the offline cached version of the website on the client’s machine.**

## DNS & Domain Names

### [ Learn: [1](https://frontendmasters.com/books/front-end-handbook/2019/#4.3), [2](https://docs.microsoft.com/en-us/windows-server/administration/windows-commands/nslookup), [3](https://www.computerhope.com/nslookup.htm) ]

1. What is an Internet Standard, and which document defines the DNS protocol? **A technology or methodology applicable to the Internet (HTTP, Bluetooth, etc.). The Domain Name System defines the DNS protocol.**
2. Which design strategies enable the DNS to scale? **The hierarchical design of DNS allows for load balancing across multiple servers.**
3. Demonstrate the interaction of the resolver library with the DNS servers.
   1. Using **nslookup**, manually replay all name-server requests emitted by the resolver library and its primary name server when searching for an A record for the name unknown.tu-dresden.de.
   2. Did one of the DNS answers come from a name server's cache? If yes: How would the request-answer sequence have differed if all the participating name server's caches would have been empty? **Yes, it would re-direct to other servers in hierarchical order from root to domain-level name servers.**
4. Besides translating names to IP addresses, which other information is stored in DNS? **Zone Files and the different Record Types within (SOA, A & AAAA, CNAME, MX, NS, PTR and CAA).**
5. Try looking up a well-known address: type www.microsoft.com. Notice that the query returns several IP addresses (Microsoft load-balances Web traffic by using multiple servers in the same DNS record).
6. Try looking up a nonexistent hostwww: type www.fubijar.com. Notice that your server complains that it can't find the address. This is normal behavior.
7. Change the server to a nonexistent host (try making up a private IP address that you know isn't a DNS server on your network, like 10.10.10.10). Do this by typing server ipAddress. Nslookup will try to turn the IP address into a hostname. Eventually it will display a message telling you that the new default server is using the IP address you specified.
8. Try doing another lookup of a known DNS name. Type www.microsoft.com. Notice that nslookup is contacting the server you specified and that the lookup times out after a few seconds.

## Serving Web Assets

### [ Learn: [1](https://developer.mozilla.org/en-US/docs/Learn/Common_questions/What_is_a_web_server), [2](https://www.youtube.com/watch?v=JhpUch6lWMw), [3](https://www.youtube.com/watch?v=1ndlRiaYiWQ) ]

1. Nginx
   1. Install [**Nginx for Windows**](https://www.maketecheasier.com/install-nginx-server-windows/)
2. Apache
   1. Install [**Apache for Windows**](https://httpd.apache.org/docs/2.4/platform/windows.html)
3. Node.js
   1. Install [**HTTP Server**](https://www.npmjs.com/package/http-server) using NPM
4. 200 OK!
   1. Install [**Web Server for Chrome**](https://chrome.google.com/webstore/detail/web-server-for-chrome/ofhbbkphhbklhfoeikjpcbhemlocgigb?hl=en)
5. Abyss
   1. Install [**Abyss Web Server**](https://aprelium.com/abyssws/download.php)

## Hosting Providers

### [ Learn: [1](https://aws.amazon.com/free/?all-free-tier.sort-by=item.additionalFields.SortRank&all-free-tier.sort-order=asc&awsf.Free%20Tier%20Types=*all&awsf.Free%20Tier%20Categories=*all), [2](https://try.digitalocean.com/freetrialoffer/), [3](https://www.vultr.com/promo/try50/), [4](https://www.youtube.com/playlist?list=PLYxzS__5yYQk7h6aoN5_rvvvC8WUMxAaB) ]

1. AWS
   1. Create an Amazon Web Services account
2. Digital Ocean
   1. Create an account at Digital Ocean
3. Vultr
   1. Create an account at Vultr

## SSL Certificates

### [ Learn: [1](https://aprelium.com/abyssws/articles/self-signed-cert.html), [2](https://www.sslforfree.com/), [3](https://zeropointdevelopment.com/how-to-get-https-working-in-windows-10-localhost-dev-environment/) ]

1. Using ANY of the servers from the objectives above, install an [**SSL certificate**](https://www.sslforfree.com/) for localhost (127.0.0.1).
   1. **Screenshot.**

## Project Layout

1. Create a template directory that mirrors [**THIS ONE**](https://1drv.ms/t/s!AnLYPxKmUDfciLIFQSGgWgEjQaJQLA?e=R8ydd1).
   1. **Screenshot of top folder structure.**

## Chrome Development Tools

### [ Learn: [0](https://www.udemy.com/course/chrome-devtools-web-developers-tutorial/), [1](https://www.codecademy.com/articles/use-devtools), [2](https://developer.chrome.com/docs/devtools/), [3](https://www.youtube.com/watch?v=x4q86IjJFag), [4](https://www.youtube.com/watch?v=Y3u2groOG-A), [5](https://developers.google.com/web/tools/lighthouse), [6](https://www.youtube.com/watch?v=x4q86IjJFag&t=2s) ]

1. Developer Console
   1. Complete [**THESE EXERCISES**](http://masteringdevtools.com/exercises)
      1. **I completed #1 through #4 and provided screenshots.**
2. Extensions
   1. Install all [**THESE CHROME EXTENSIONS**](https://wpastra.com/chrome-developer-extensions/)
      1. **I installed all extensions that were able to be installed.**
3. Google Lighthouse
   1. Use Google lighthouse to measure five (5) websites. Compare and contrast with [**THIS TOOL**](https://web.dev/measure/).
      1. **Screenshots provided.**