Homework-3

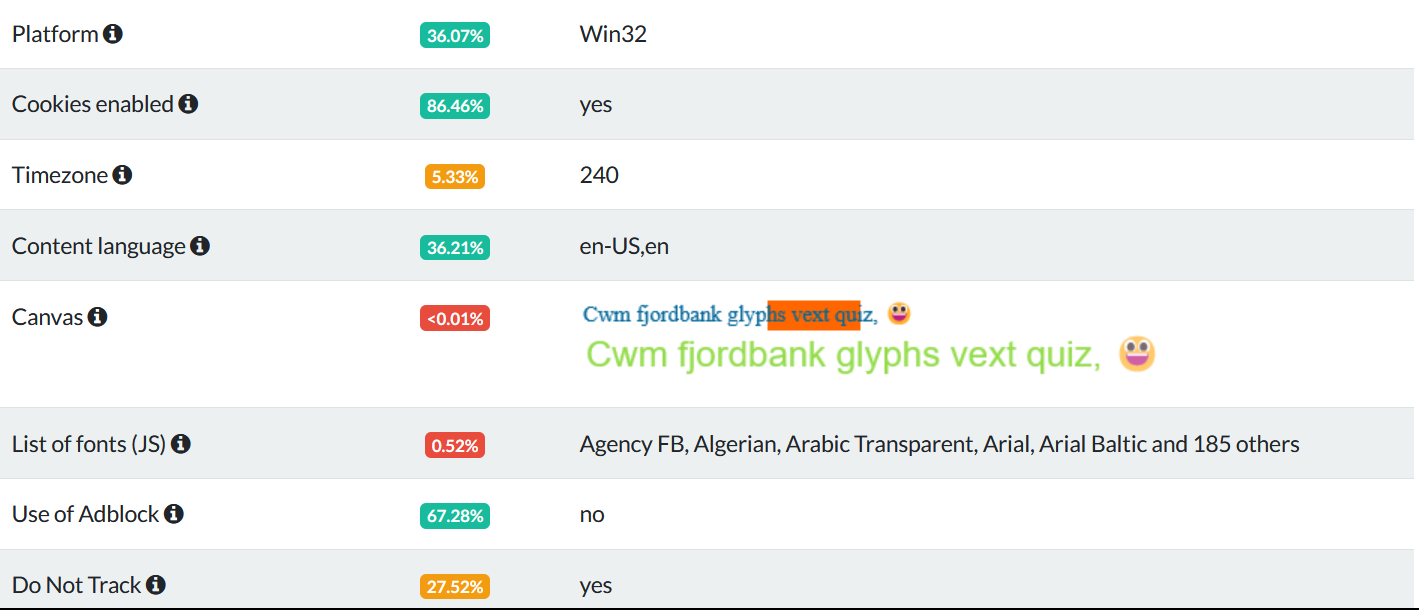
CSC 533 || Privacy in Digital Age

Question 1)

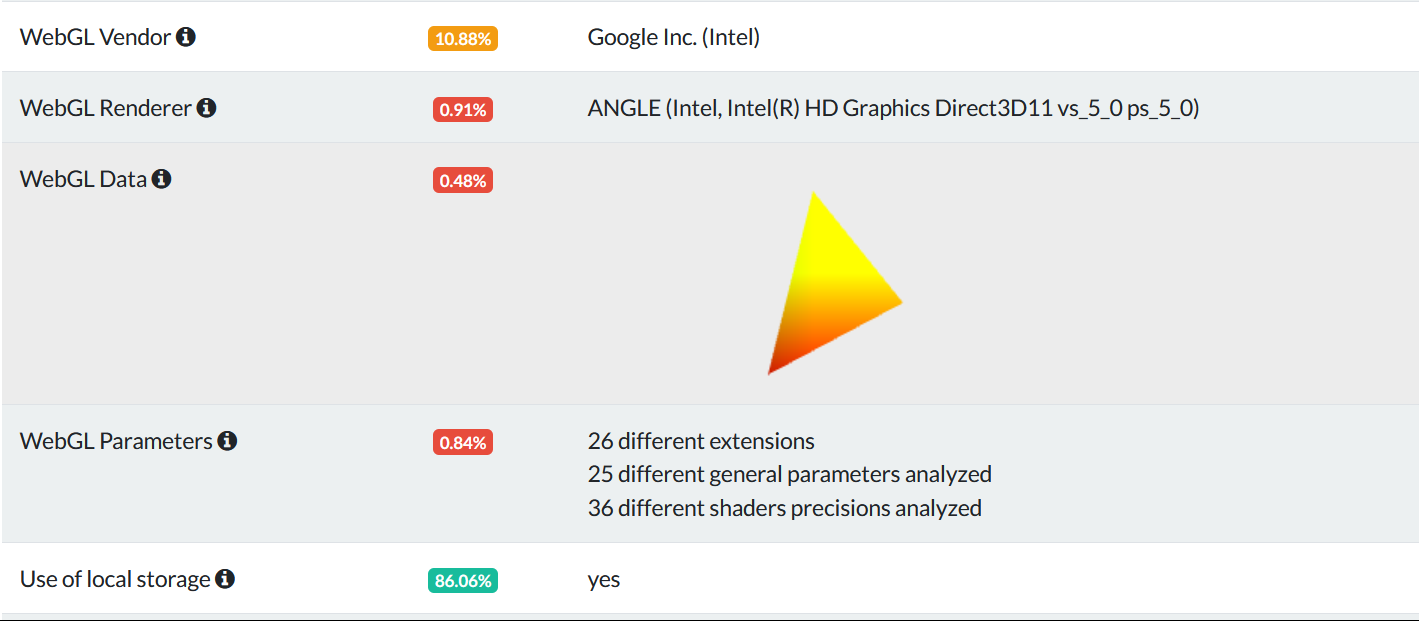
(a) **Major Differences between two finger browser-prints:** Here, the two browsers used are Firefox and Tor browser. The screenshots below highlight key differences between the two fingerprints. They are numbered in the same order, for both categories, that is, point (1) for Firefox and point (1) for Tor correspond to the same portion of the fingerprint, and so on, respectively.

* Mozilla Firefox:





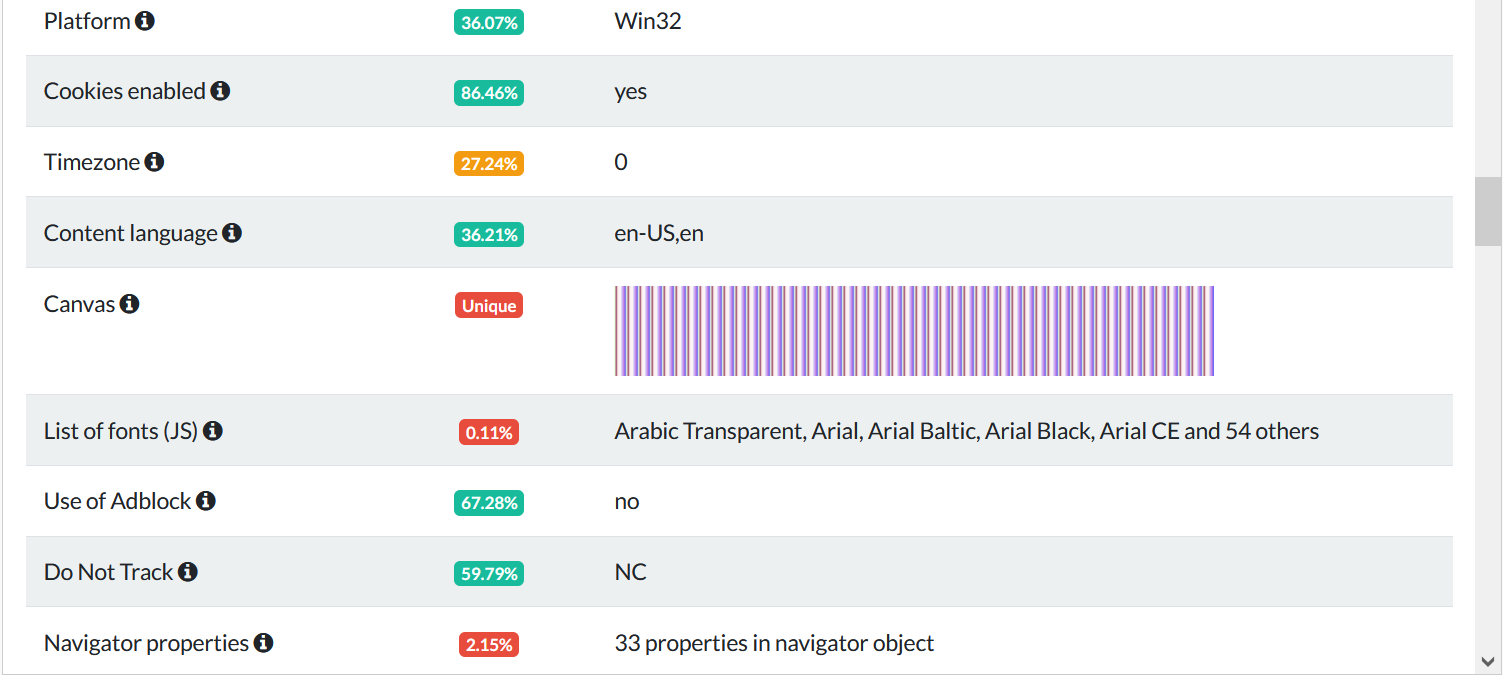




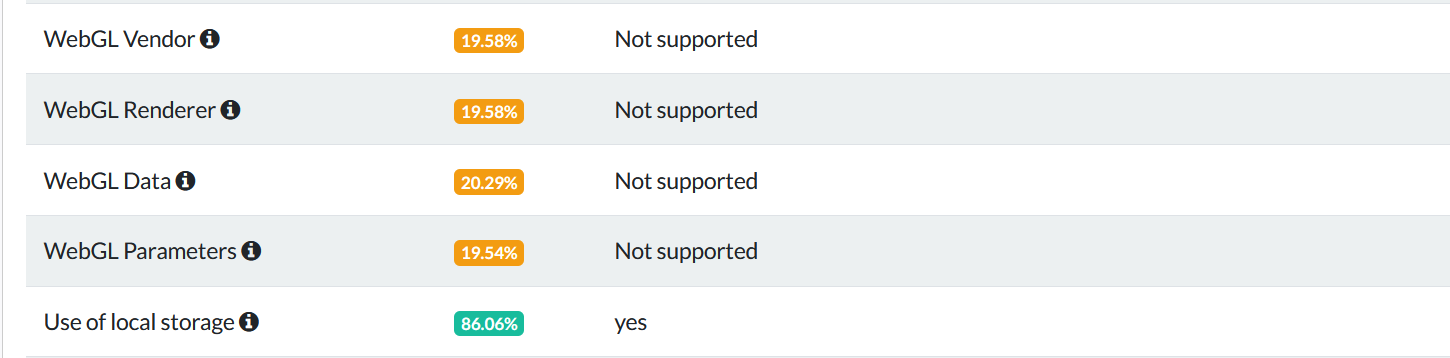


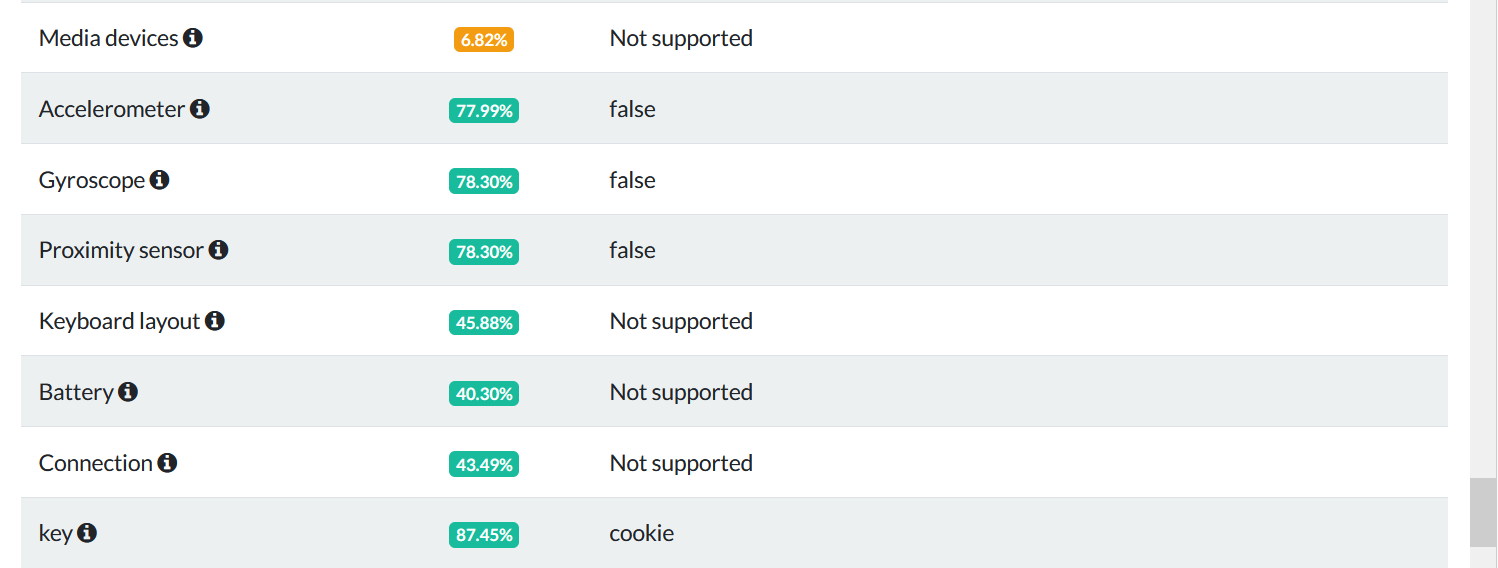
* Tor:











Analysis: There are five screenshots used that highlight major differences when it comes to fingerprinting, for which we have the following five points:

1. Browser Version and Time Zone (First Snapshot): The browser version for Firefox is 0.94% similar, while for Tor, it is 11.14% similar. This is a huge difference, which could imply many things. One conclusion is that a very few number of people use the current version of Firefox, making it highly identifiable, and lesser number of people use this particular version of Tor (built on top of Firefox, which is why it shows Firefox), which makes it less identifiable. Another can be that Tor is updated regularly, since it is used by people who are more concerned with privacy than Firefox, so using a newer version in Firefox might make one more identifiable. Without further data, these conclusions cannot be validated to a certainty.

Time Zone has 5.33% similarity index for Firefox while a 27.24% for Tor. This difference is also significant, which can imply that the number of people using Firefox in this particular time zone is lesser than the percentage of people using Tor, for the same.

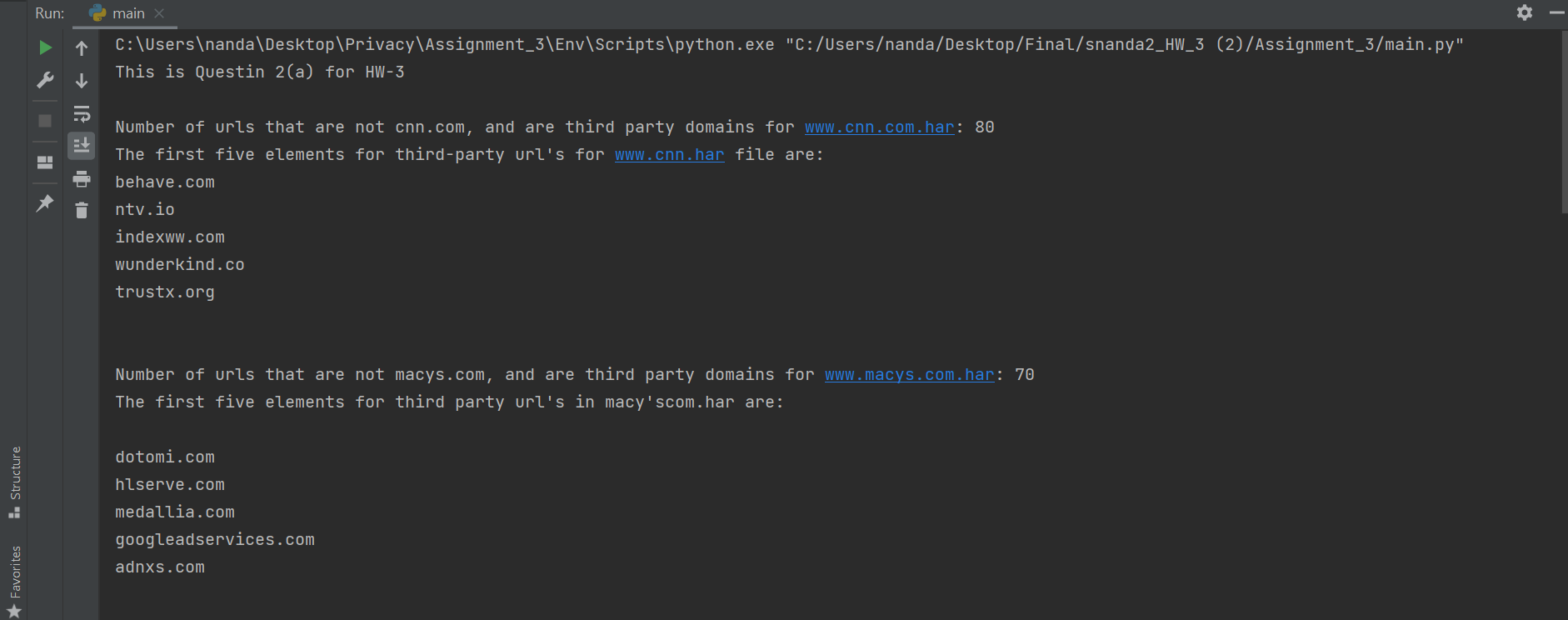
1. Canvas, List of Fonts, Do not Track (Second Snapshot): Canvas is highly unique for Firefox (<0.01%) and completely unique in Tor, which is not desirable. It could imply that from graphical rendering of html 5 canvas, one could completely identify Tor, and to a large degree, Firefox. List of Fonts are 0.52 and 0.11 on similarity scale, which means that JS could get more test attributes on Firefox, based on installed list of fonts for fingerprinting in Firefox than on Tor. Also, list of fonts used in Firefox is higher, which could also be a factor for higher similarity. The “Do not track” attribute has 27.52 % and 59.79% similarity indices for Firefox’s DNT signal setting set to “Standard” by default, and Tor’s set to “NC”, by default, which means that it’s hard to fingerprint Tor based on its default setting than it is to fingerprint Tor. In this aspect, Tor seems to be a clear winner (in absence of more data).
2. Screen Width, Screen Height, Screen Available Width, Screen Available Height (Third Snapshot): The respective values are {3.12,1.91,0.28,3.04} for Firefox and {0.03,0.13,0.13,0.03} for Tor. Based on this, we can say that based on screen dimensions and pixels of free space available on window of browser, Tor is more identifiable or easily finger printable than Firefox. It is a common issue, which is why it’s always recommended to never maximize your Tor window.
3. WebGl Vendor,Renderor,Parameters, Data (Snapshot 4): The similarity indices for the four attributes for Firefox ={0.88,0.91,0.48,0.84} and Tor={19.58,19.58,19.54,20.29}. Based on this, it seems that it’s much more easier to fingerprint Firefox from the information regarding the graphics card, and for Tor, this information is not directly accessible (as seen from the right most column, value =”Not Supported”).
4. Media Devices (Snapshot 5): The similarity indices for media devices in Firefox and Tor are “unique” and 6.82%. This difference seems significant, and seems to indicate that it’s easier to fingerprint Firefox based on hardware attributes like cameras, microphone array, etc, than in a Tor browser.

(b) Basic Takeaway in terms of tracking capability of two browsers: From above, the takeaway seems to be, that while Tor is better at hiding the Graphic Card, Camera, Microphone and other hardware related information, it is not good at hiding away the rendering components of a page, including Screen Height,Width, and free space for available height and width. This is a major concern, and most online resources warn against fingerprinting based on this, and a common solution that’s offered is to not maximize the Tor window without some sort of a fix. With regards to page rendering, the HTML 5 canvas is completely unique in Tor, while almost unique in Firefox, which again means that while Tor is good at hiding away hardware based specifications, it has some shortcomings when it comes to page rendering. The values for other rendering attributes like fonts for JS are comparable, but as expected, more unique in Tor than in Firefox. Also, when it comes to default DNT settings, Tor fairs way higher than Firefox, in that it’s much more common for DNT to be set to NC in Tor than in Firefox, which is expected, as Tor users are usually more concerned about being tracked and the default set to NC is expected among a large proportion of users. The data related to timezone could lead to a multitude of conclusions, one of which is that the given timezone has more number of Tor users or at least, more number of Tor users than Mozilla users who do not use VPN (plus those whose VPN points to the given destination). Finally, the browser versions for Tor are more similar, in that more number of people use the latest version. This could imply that Tor users update more regularly than Firefox users, as one conclusion, or that Tor users are more concerned about privacy, so they update frequently to avoid security concerns. These conclusions can be validated better with more data, where we can find out more or unexpected factors for the same.

Question 2)

(a) Number of unique third party domains that are visited:

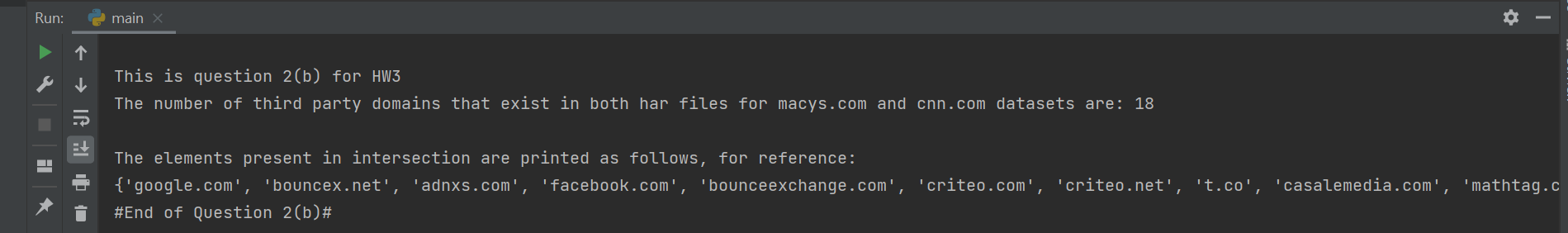
From the code output, the number of unique third party domains visited while visiting [www.cnn.com](http://www.cnn.com) is 80, while that for [www.macys.com](http://www.macys.com) is 70. The first five elements for the two sets (sets were used instead of lists for uniqueness, as python sets only allow unique elements) are also printed in the output below for reference:



(b) Number of unique third party domains in both macys.com and cnn.com:

The number of unique third-party domains that occur in both [www.macys.com](http://www.macys.com) and [www.cnn.com](http://www.cnn.com) from part (a), are calculated as 18, as per the output screen below.

The domains are also printed to the output, for reference.

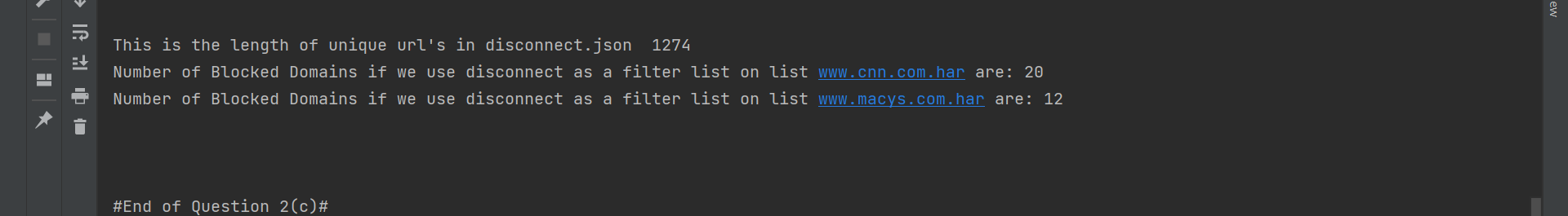


(c) Number of Requests blocked if we were to use Disconnect.json:

The number of third-party domains that would be blocked, if we use disconnect.json as a filter list are:

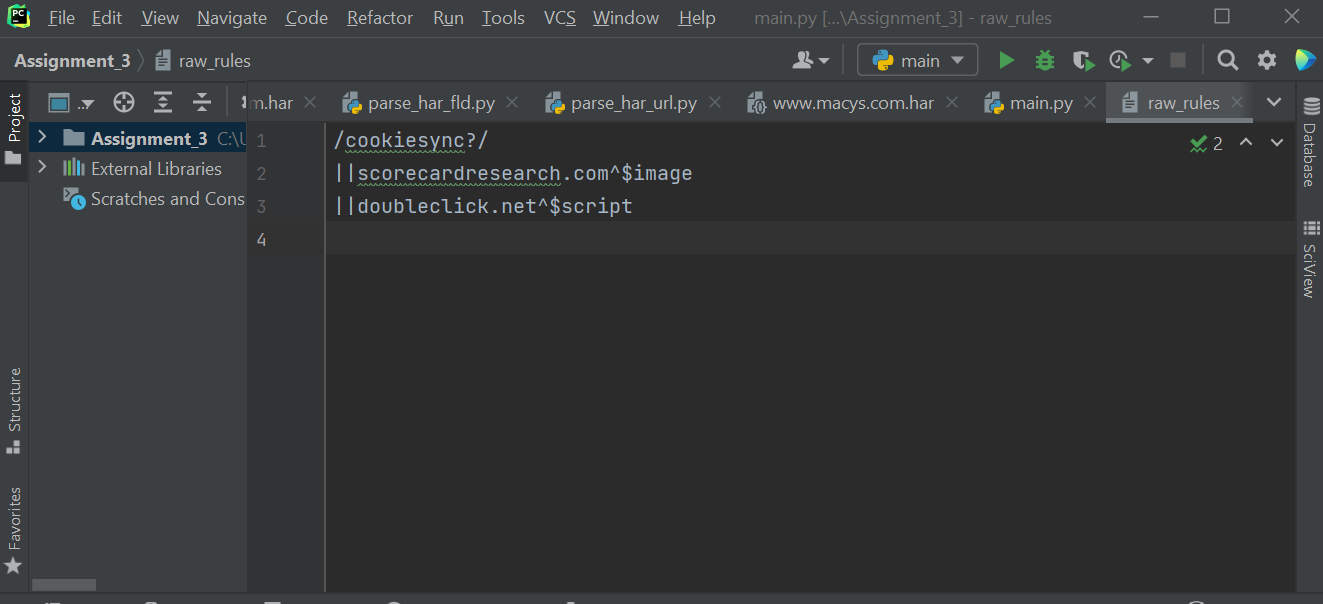
1. 20 : For [www.cnn.com](http://www.cnn.com)
2. 12 : For [www.macys.com](http://www.macys.com)

These are also produced to output, as shown below:



Question 3)

Writing and Testing adblock rules: We are given three adblock rules, which are used as ‘raw-rules’ in the files labelled raw-rules.txt (Also present in the zip file). Below is the screenshot for the three raw rules that were used:



Based on these raw rules, number of domains that were blocked in [www.cnn.com.har](http://www.cnn.com.har) file were 8, and these are also printed in the specified format mentioned in the question. Please note that repetition here does not denote the same url, since we are getting fld’s for the corresponding url’s (from get\_fld() function in the tld package), and two different third party websites can have the same adblock rule and the same third-party domain, that is, there is no actual repetition.

