Sairam Soundararajan

SDEV300: Building Secure Python Applications

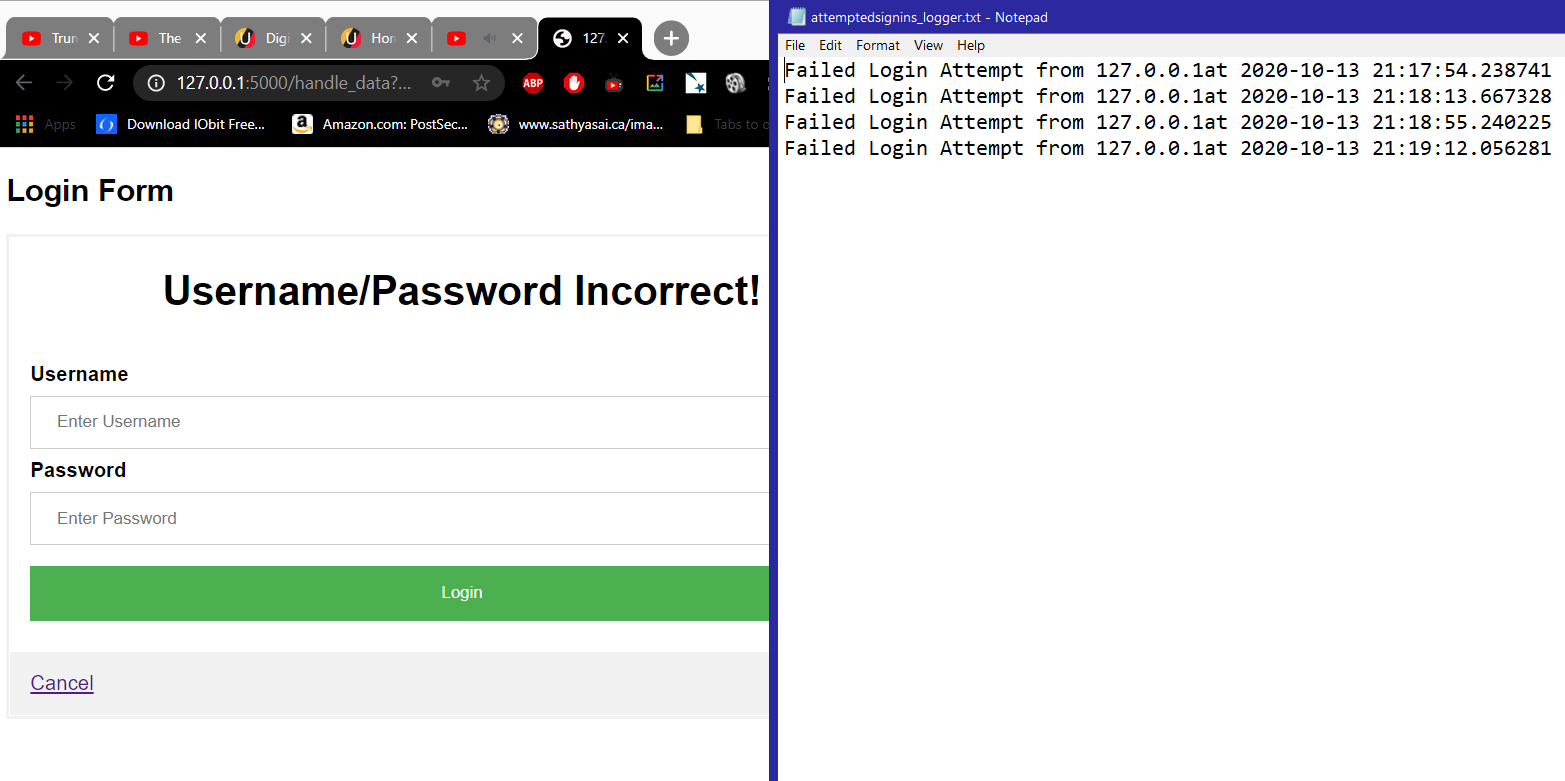
University of Maryland Global Campus

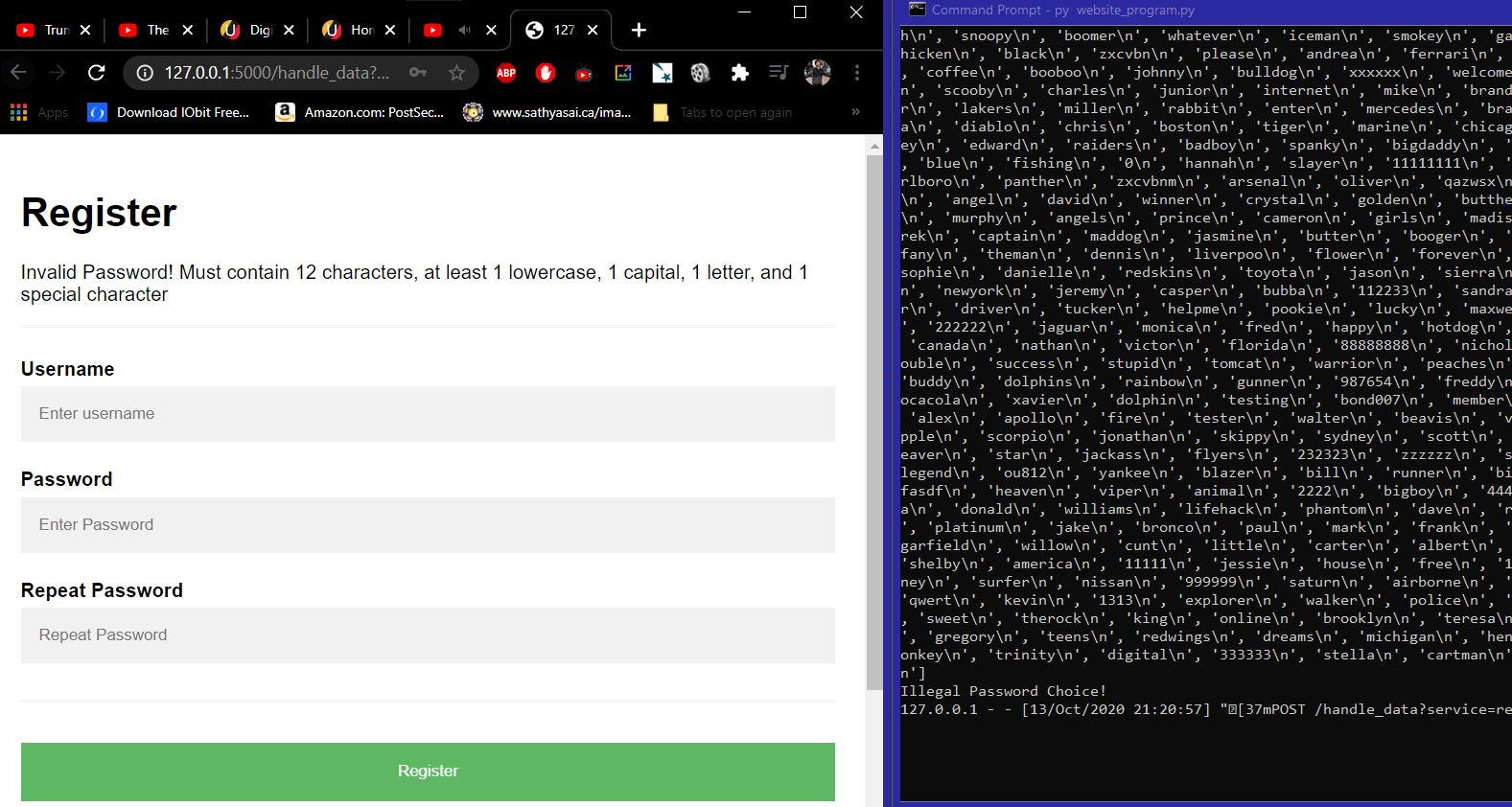
Professor Howards

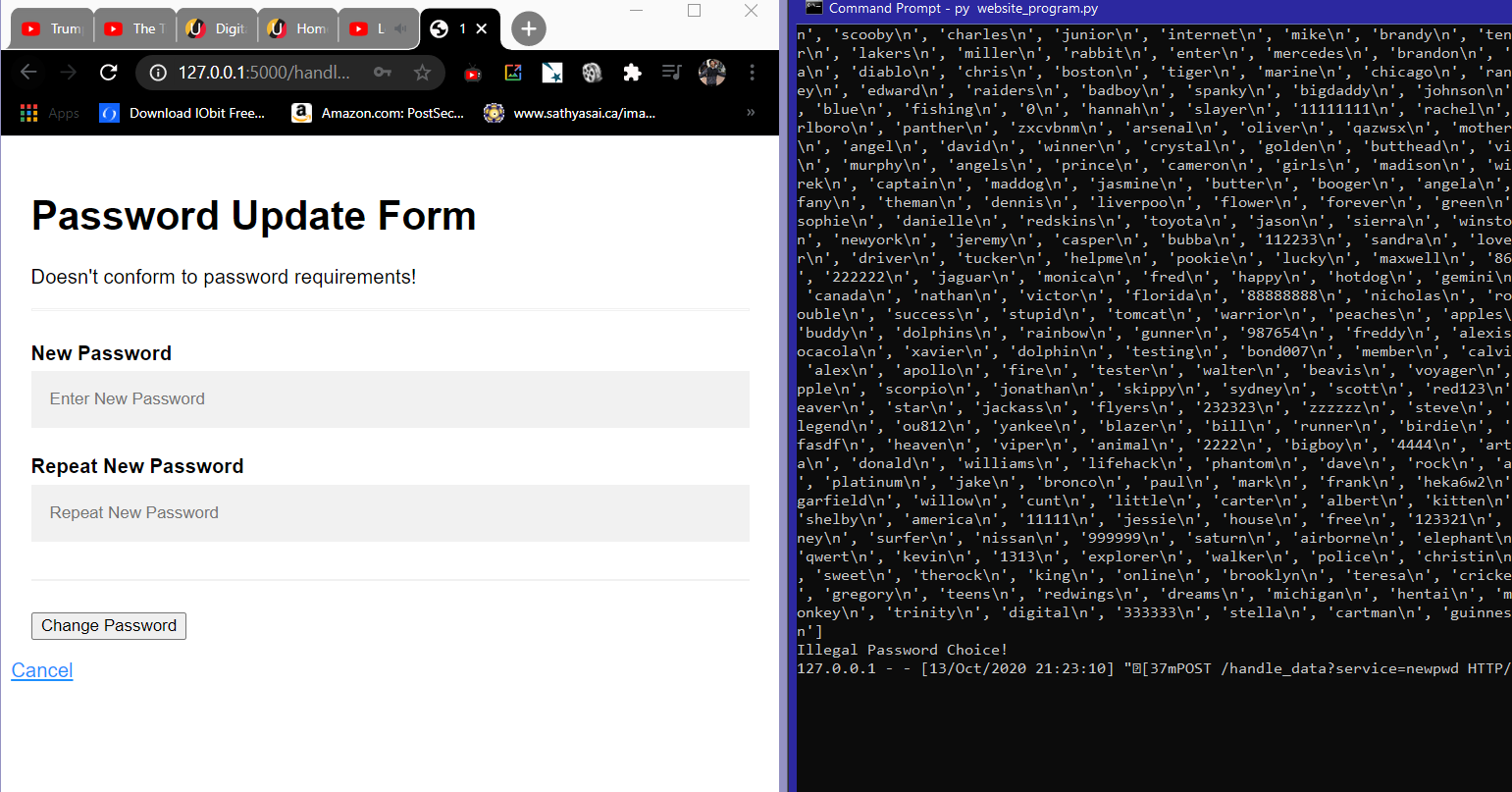
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test Case | Input | Expected Output | Actual Output | Pass? |
| 1 | This program does not contain user input within the code since this is an HTML program. | \* Serving Flask app "website\_program" (lazy loading)  \* Environment: production  WARNING: This is a development server. Do not use it in a production deployment.  Use a production WSGI server instead.  \* Debug mode: off  \* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit) | \* Serving Flask app "website\_program" (lazy loading)  \* Environment: production  WARNING: This is a development server. Do not use it in a production deployment.  Use a production WSGI server instead.  \* Debug mode: off  \* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit) | Yes |

Snapshots of Test Run:

The website will only operate during test run. Once the test run is over, the website will be inaccessible until the program is run again.

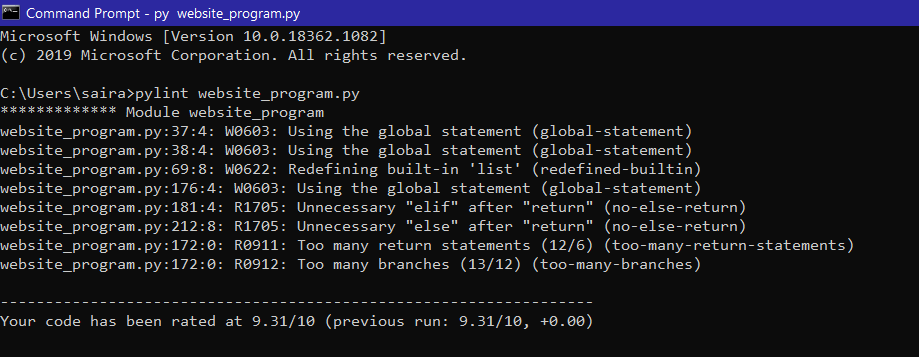






Pylint Discussion and Snapshots:

Before: Although I do not remember the values of the decimal precision after ‘8.’, my score before was 8.\*\*/10 (the asterisk are because I do not remember the values as mentioned). I had to rewrite the name of some variables and correct any subsequent errors to satisfy Pylint.

After: 

Although I have gotten rid of all the conventions and mitigated the warnings and refactors, I found it challenging to change the rest of the warnings and refactors. I know that 10 is the goal and I strongly apologize for not accomplishing that part. I was, however, able to improve the code to make the score closer to 10 as possible. As a result, my code has been rated at 9.31/10.

Log and NIST Discussion:

Whenever I sign in with an invalid username/password, a logfile(text file) gets created by the program and will display each login attempt with the date, time, and IP address following it.

In terms of NIST, I have made it possible for the program to know when a password being created in the Register or Password Update Form is matching any of the passwords in the CommonPasswords.txt. For every time a common password is attempted to be created, not only does the webpage mark it as invalid but the console will specifically say “Illegal Password!” as soon as the inputted password matches any one of the text file.

Cipher Code Decryption Portion of Lab 8:

Morse Code Message:

- .... .. ... / ... -.. . ...- / ...-- ----- ----- / -.-. .-.. .- ... ... / .... .- ... / ... --- -- . / ... - .-. .- -. --. . / .-. . --.- ..- . ... - ... .-.-.-

Morse Code Decryption:

THIS SDEV 300 CLASS HAS SOME STRANGE REQUESTS.

Base64 Message:

U28gdGhpcyBpcyBiYXNlNjQuIE5vdyBJIGtub3cu

Base64 Decryption:

So this is base64. Now I know.

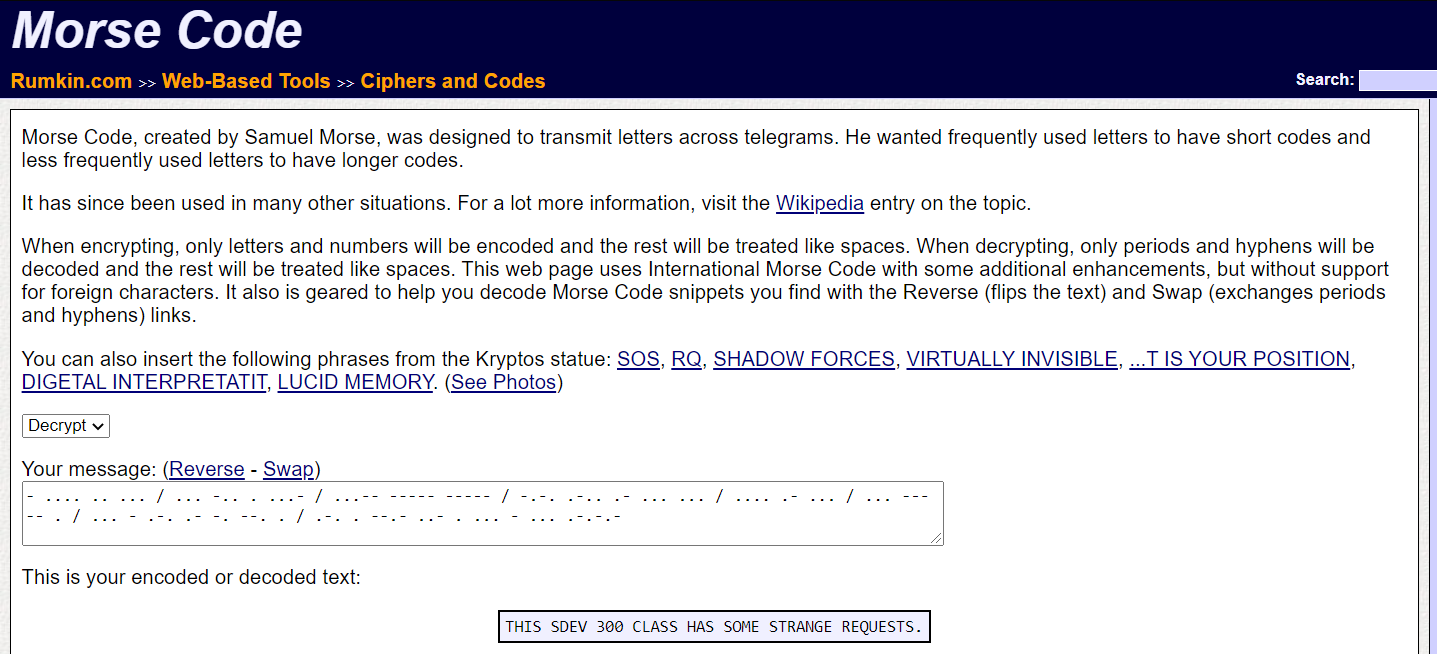
Caesar Cipher Message:

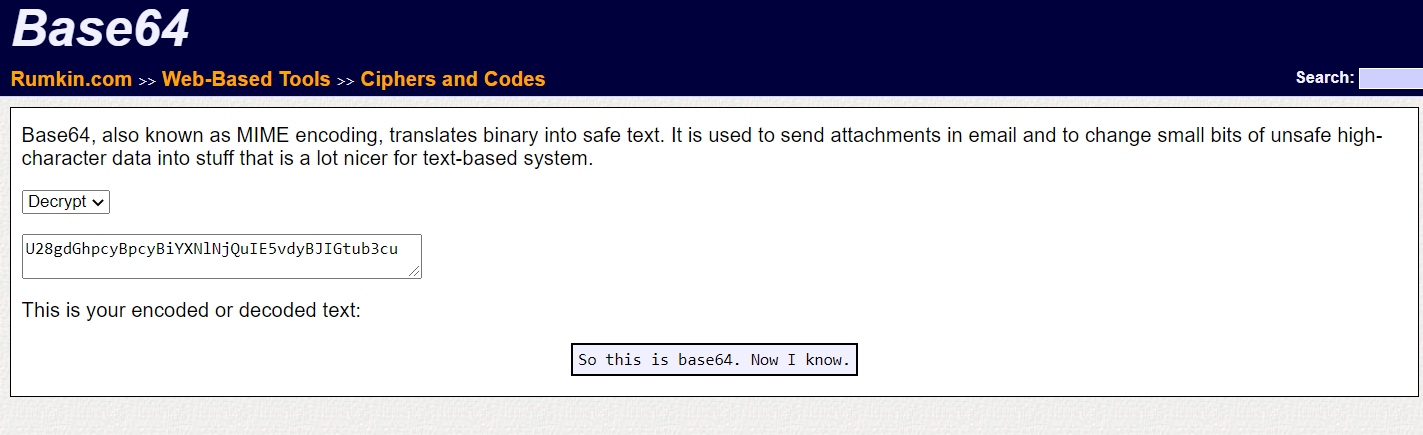
--- Psuwb Ysm ---- W oa gc qzsjsf. Bc cbs qcizr dcggwpzm twuifs hvwg cih. --- Sbr Ysm ---

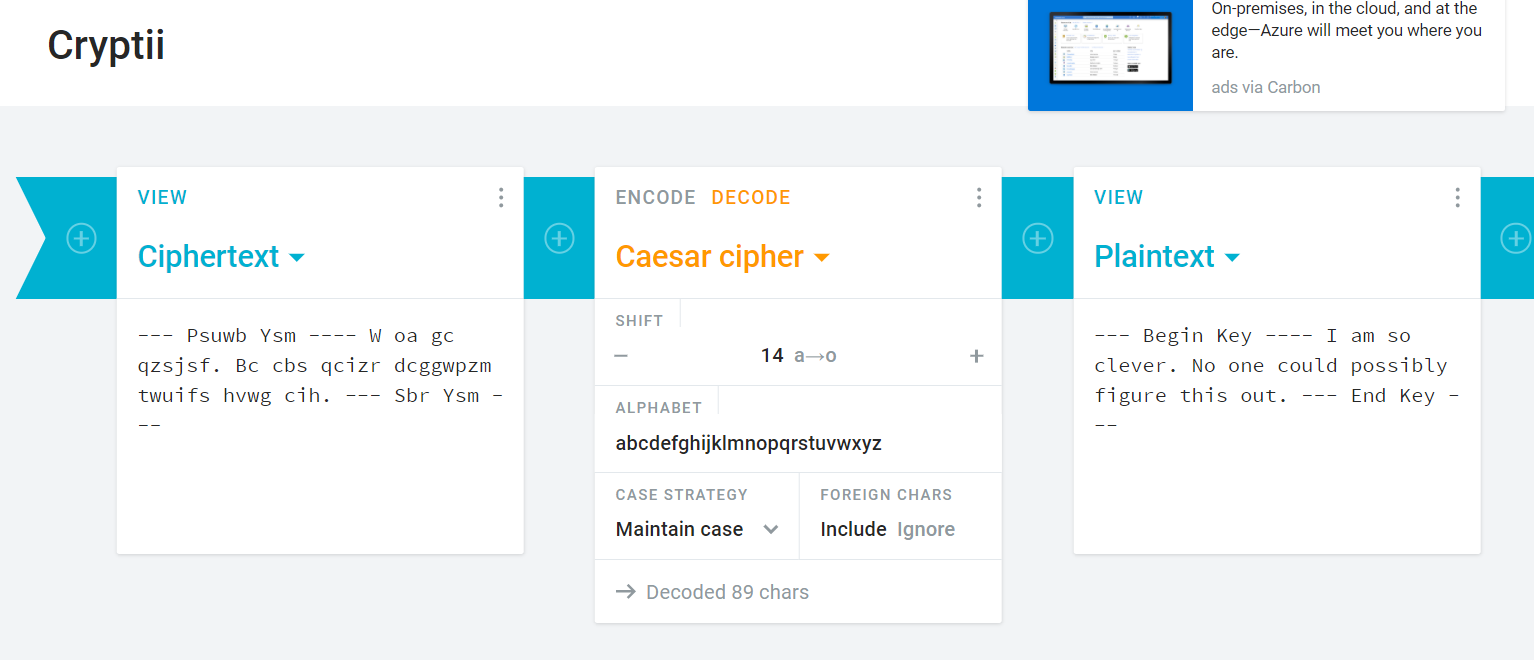
Shift: 14

Caesar Cipher Decryption:

--- Begin Key ---- I am so clever. No one could possibly figure this out. --- End Key ---







Conclusion:

Thank you Professor Howards for your time and everything. I enjoyed learning python and found the code to be easier to follow. I am grateful for the experience I had while taking this course. It has been a pleasure taking this course and having you as my professor. Have a great rest of the fall!

Regards,

Sairam