**Build A Privacy-Preserving Web Service for User Credential Verification**

I purchased four large files from the black market which contain millions of breached user emails and passwords. These files were stolen from a large social media company. As shown below, the breached user emails and passwords, separated by ':', were from real users. For example:

* *jscockburn@yahoo.com:tammy&ivy*
* *amatorydesigns@yahoo.com:529092349974921*
* *lauraric@aol.com:708405616084772*
* *c\_bradford84@yahoo.com:Fordumb132*
* *pnjbiluv@yahoo.com:248033933788426*

Since people tend to reuse the same password in many places, it can be dangerous for anyone to continue to use such breached email/password, since hackers may well have access to these passwords and test them against important websites such as your bank account. Therefore, it will be prudent if a user is provided with a web service that they can use to verify that their new password does NOT appear in these breached files before they decide to adopt it.

To build such a web service, we have decided to store all email and password pairs in a database table. Since the database can be hacked too, we have decided not to store the email and password as plain text directly; instead, we will compute the cryptographic hash for each and store the hash values instead. Similarly, when a user wants to verify that their email and password do not appear in our database, we do not send them as plain text in wire either; instead, we send their hash values to the database server for comparison. Anyone who cares to pause to think about this for one second will realize that the use of hashing will not only work but also be essential as nobody will use the provided service once they realize that their password will be sent, in plain text, to a remote server that they cannot trust.

You will work in a team of two to complete this project in two parts:

* **Part 1**: load the email and password in hash values into a database table named 'users';
* **Part 2**: build a simple website where a user can verify if their email and password appear in the database table. Use the Flask web framework.

**References:**

1. <https://www.tutorialspoint.com/flask/index.htm>
2. <https://flask.palletsprojects.com/en/3.0.x/tutorial/>
3. Cryptographic hashing with [hashlib](https://www.geeksforgeeks.org/hashlib-module-in-python/) module in Python