Practice Exercise

- When we verify a user using a password, the most important requirement is that you MUST not store the password in plaintext.

  If an attacker gets access to the database of plaintext passwords, he / she can use this information to exploit the data.

  One common way of storing a password in a database is by hashing the password and storing its hashed value in the database.

  You, as a developer, have to write a Python program. In this program you will use the module **hashlib** to generate a hash of the password.

  For this you have to do the following activities…

  a. Write a function **hash_password(password)** - this function will generate the hash value of the plaintext password passed to this function.

  b. Write a function **store_user_details(username, password)** - A user will call this function and pass the username and plaintext password. Use a dictionary data structure to store the username (key) and its corresponding hashed password (as value). You will not store the plaintext password.

  c. Write a function, **verify(username, password)** - A user will call this function and pass the username and plaintext password. This function will first check whether a specified username is valid or not. If a username is valid, then it will validate the password. You have to show an appropriate message when the user and/or password validates or invalidates.