|  |
| --- |
| import sqlite3 |
|  |  |
|  | # Sqlite database and table |
|  | def sqllitefile(): |
|  | global conn |
|  | conn = sqlite3.connect('login\_data.db') |
|  | global c |
|  | c = conn.cursor() |
|  | c.execute('''CREATE TABLE IF NOT EXISTS login\_details |
|  | (USER\_ID TEXT PRIMARY KEY NOT NULL, |
|  | PASSWORD TEXT NOT NULL);''') |
|  |  |
|  |  |
|  | # For choosing between register and login |
|  | def choose(): |
|  | sqllitefile() |
|  | nc = True |
|  | while nc: |
|  | print('Register or Login') |
|  | lor = input('Type Login or Register to continue: ') |
|  | if lor == 'login' or lor == 'Login' or lor == 'LOGIN': |
|  | nc = False |
|  | return login() |
|  | elif lor == 'register' or lor == 'Register' or lor == 'REGISTER': |
|  | nc = False |
|  | return register() |
|  | else: |
|  | print('---------------------------------------------') |
|  | print('Please only choose between Login and Register') |
|  |  |
|  |  |
|  | # Instructions for registration |
|  | def useridins(): |
|  | print('--------------------------------------------------------------------------------') |
|  | print('email/username Instructions:') |
|  | print('1. email/username should have "@" and followed by "."') |
|  | print('2. It should not be like this Eg:- @gmail.com') |
|  | print('3. There should not be any "." immediate next to "@" Eg:- my@.in') |
|  | print('4. It should not start with special characters and numbers Eg:- 123#@gmail.com') |
|  |  |
|  |  |
|  | def passwordins(): |
|  | print('------------------------------------------------------------------') |
|  | print('Password Instructions:') |
|  | print('1. Password length must be between 5 and 15 \n2. Password must contain atleast one special character') |
|  | print('3. Password must contain atleast one number') |
|  | print('4. Password must contain atleast one upper case character') |
|  | print('5. password must contain atleast one lower case character') |
|  |  |
|  |  |
|  | # For conditions in user id for registration |
|  | def user\_id\_check\_func(u): |
|  | numbers = '0123456789' |
|  | sc = """~`!@#$%^&\*()\_-+={[}]|\:;"'<,>.?/""" |
|  | ati, doti = 0, 0 |
|  | for i in range(len(u)): |
|  | if u[i] == '@': |
|  | ati = i |
|  | elif u[i] == '.': |
|  | doti = i |
|  | if ((u[0] not in numbers) and (u[0] not in sc)) and (doti > (ati + 1)): |
|  | return True |
|  | else: |
|  | return False |
|  |  |
|  |  |
|  | # For conditions in password for registration |
|  | def password\_check\_func(p): |
|  | sc = """~`!@#$%^&\*()\_-+={[}]|\:;"'<,>.?/""" |
|  | numbers = '0123456789' |
|  | uc = 'ABCDEFGHIJKLMNOPQRSTUVWXYZ' |
|  | lc = 'abcdefghijklmnopqrstuvwxyz' |
|  | scb, numbers\_b, ucb, lcb = False, False, False, False |
|  | for i in range(len(p)): |
|  | if p[i] in sc: |
|  | scb = True |
|  | elif p[i] in numbers: |
|  | numbers\_b = True |
|  | elif p[i] in uc: |
|  | ucb = True |
|  | elif p[i] in lc: |
|  | lcb = True |
|  | if scb == numbers\_b == ucb == lcb == True: |
|  | return True |
|  | else: |
|  | return False |
|  |  |
|  |  |
|  | # For Registeration |
|  | def register(): |
|  | # User Name |
|  | nu = True |
|  | while nu: |
|  | useridins() |
|  | print('---------------------------------------') |
|  | user\_id = input('Enter your email/username to register: ') |
|  | user\_id\_check = user\_id\_check\_func(user\_id) |
|  | if user\_id\_check == False: |
|  | print('--------------------------------------') |
|  | print('Read the below instructions carefully:\n') |
|  | else: |
|  | nu = False |
|  |  |
|  | # If user\_name already exists |
|  | exist = conn.execute("select USER\_ID from login\_details where USER\_ID like ?", (user\_id,)).fetchone() |
|  | if exist: |
|  | print('------------------------------------------') |
|  | print('user\_name not available') |
|  | print('Please use different user name to register') |
|  | register() |
|  |  |
|  | # Password |
|  | np = True |
|  | while np: |
|  | passwordins() |
|  | print('----------------------------') |
|  | password = input('Enter password to register: ') |
|  | password\_check = password\_check\_func(password) |
|  | if 5 < len(password) < 16 and password\_check: |
|  | np = False |
|  | else: |
|  | print('--------------------------------------') |
|  | print('Read the below instructions carefully:') |
|  |  |
|  | # File Handling |
|  | c.execute("INSERT INTO login\_details (USER\_ID, PASSWORD) \ |
|  | VALUES (?, ?)", (user\_id, password)) |
|  | conn.commit() |
|  | c.close() |
|  | conn.close() |
|  | print('------------------------') |
|  | print('congratulation....') |
|  | print('Registration is complete') |
|  |  |
|  |  |
|  | # For Password in Login |
|  | def login\_password(): |
|  | # Password |
|  | print('-------------------------------------------------------------------------------------') |
|  | print("Enter any number other than 1 to type password or Enter 1 if you forgot your password") |
|  | pc = input() |
|  | if pc == '1': |
|  | password\_id = c.execute("select password from login\_details \ |
|  | where user\_id = ?", (user\_id,)).fetchone() |
|  | print('-----------------------------------------------------------------') |
|  | print(f'Password for the mentioned username - {user\_id} is :') |
|  | print(password\_id[0]) |
|  | else: |
|  | print('--------------------------') |
|  | password = input('Enter Password for login: ') |
|  | password\_id\_exists = c.execute("select password from login\_details \ |
|  | where user\_id = ? and password = ?", (user\_id, password,)).fetchone() |
|  | if password\_id\_exists: |
|  | print('------------------') |
|  | print('Login Successfull \nWelcome....') |
|  | else: |
|  | fl = 0 |
|  | print('--------------------------------') |
|  | print('Password is Incorrect, Try again') |
|  | login\_password() |
|  |  |
|  |  |
|  | # For Username in Login |
|  | def login(): |
|  | # User Name |
|  | global unchoice |
|  | unchoice, re = 1, 1 |
|  | global user\_id |
|  | print('------------------------------------') |
|  | user\_id = input('Enter username/email to login: ') |
|  | user\_id\_exists = c.execute("select USER\_ID from login\_details where USER\_ID like ?", (user\_id,)).fetchone() |
|  | if not user\_id\_exists: |
|  | print('------------------------------------------------------------------') |
|  | print('user name does not exist, Please Register first') |
|  | print('choose 1 to register or choose any other number to try login again') |
|  | unlchoice = input() |
|  | if unlchoice != '1': |
|  | login() |
|  | re = 0 |
|  | else: |
|  | unchoice = 0 |
|  | register() |
|  | # password |
|  | if unchoice and re: |
|  | login\_password() |
|  |  |
|  |  |
|  | if \_\_name\_\_ == '\_\_main\_\_': |
|  | choose() |