import sqlite3

conn = sqlite3.connect("users.db")

cursor = conn.cursor()

cursor.execute("""

CREATE TABLE IF NOT EXISTS users (

id INTEGER PRIMARY KEY AUTOINCREMENT,

username TEXT UNIQUE NOT NULL,

password TEXT NOT NULL

)

""")

conn.commit()

conn.close()

import bcrypt

import sqlite3

def register\_user():

username = input("Enter username: ")

password = input("Enter password: ")

hashed\_password = bcrypt.hashpw(password.encode('utf-8'), bcrypt.gensalt())

conn = sqlite3.connect("users.db")

cursor = conn.cursor()

try:

cursor.execute("INSERT INTO users (username, password) VALUES (?, ?)", (username, hashed\_password))

conn.commit()

print("User registered successfully!")

except sqlite3.IntegrityError:

print("Username already exists!")

conn.close()

def login\_user():

username = input("Enter username: ")

password = input("Enter password: ")

conn = sqlite3.connect("users.db")

cursor = conn.cursor()

cursor.execute("SELECT password FROM users WHERE username = ?", (username,))

result = cursor.fetchone()

conn.close()

if result:

stored\_password = result[0]

if bcrypt.checkpw(password.encode('utf-8'), stored\_password.encode('utf-8')):

print("Login successful!")

else:

print("Incorrect password.")

else:

print("Username not found.")

def main():

while True:

print("\n1. Register")

print("2. Login")

print("3. Exit")

choice = input("Enter your choice: ")

if choice == "1":

register\_user()

elif choice == "2":

login\_user()

elif choice == "3":

print("Exiting program. Goodbye!")

break

else:

print("Invalid choice. Please try again.")

if \_\_name\_\_ == "\_\_main\_\_":

main()