Python + MySQL: Terminal User Registration Explained

import mysql.connector

Imports the MySQL connector module to allow Python to communicate with a MySQL database. Without this, database operations like insert, update, or select would not be possible.

import getpass

Imports getpass module to safely accept password input without displaying it in the terminal. Without it, the password will be visible while typing, reducing security.

username = input("Enter your username: ")

Prompts the user to enter a username. This input will be stored in the 'username' variable.

password = getpass.getpass("Enter your password: ")

Prompts the user to enter a password in hidden mode. Uses getpass for security.

conn = mysql.connector.connect(...)

Establishes a connection to the MySQL database. Replace host, user, password, and database as needed. Without this, you can't run any queries on the MySQL database.

cursor = conn.cursor()

Creates a cursor object used to execute SQL statements. Without it, you cannot run SQL commands from Python.

query = "INSERT INTO users (username, password) VALUES (%s, %s)"

Defines the SQL query to insert the provided username and password into the database.

values = (username, password)

Creates a tuple of values to pass into the SQL query placeholders (%s). Helps prevent SQL injection.

Python + MySQL: Terminal User Registration Explained

cursor.execute(query, values)

Executes the SQL query using the cursor. Submits the data to the database engine for insertion.

conn.commit()

Commits the transaction. Without this line, the data won't be saved permanently to the database.

print("(Success) User registered successfully!")

Displays a confirmation message if everything worked correctly.

except mysql.connector.Error as err:

Begins an exception block to catch and handle any MySQL-related errors.

print(f"(Error) Error: {err}")

Displays the specific error message encountered during the database operation.

finally:

Ensures that cleanup operations run regardless of success or failure above.

if conn.is_connected():

Checks if the connection to MySQL is still active before trying to close it.

cursor.close() conn.close()

Closes the cursor and database connection to release resources. Not doing this can cause memory leaks or lock database resources.