**STM\_ICT Project 2021 SP2 #28 Simulate Sensor Data and Dashboard**

**Configure UI Code Documentation**

Team member:

YuxuanLi, YifeiRan, Magura, Munyaradzi

CONTEXT

[Overview 3](#_Toc74443289)

[Resource 3](#_Toc74443290)

[User Interaction 3](#_Toc74443291)

[Create a table 4](#_Toc74443292)

[Import csv data 4](#_Toc74443293)

[Query data 5](#_Toc74443294)

[Delete data 5](#_Toc74443295)

[Other methods of operating the database: 6](#_Toc74443296)

[Code Implementation Components 6](#_Toc74443297)

# Overview

This document is used to describe the configuration and operation instructions of the database module in the project.

# Resource

This module uses the SQLite module that comes with python. Therefore, users do not need to install the library before using this module, just confirm whether the library has been imported in the py file.

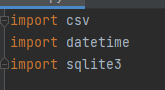
Choose SQLite as the project database type for the following reasons:

1. The main modules of the project are developed based on python. SQLite3 can be integrated with Python using the sqlite3 module.

2. SQLite is an in-process library. Unlike other databases, no user configuration is required in the system. At the same time, SQLite is also lightweight and has low system environmental requirements.

3. SQLite is self-sufficient, and the use of this database does not require users to generate external dependencies.

In addition, this module also uses the csv library and datetime library that comes with python.



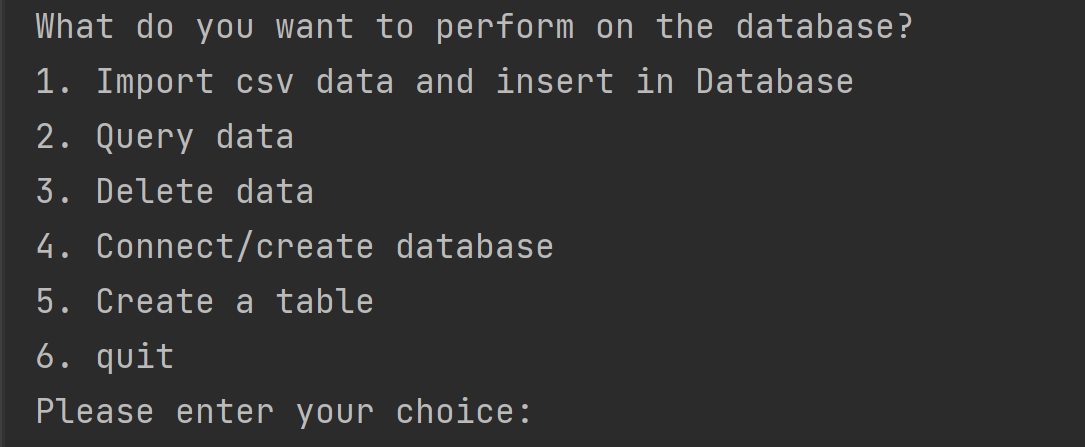
SQLite reference: <https://www.sqlite.org/index.html>

<https://docs.python.org/3/library/sqlite3.html>

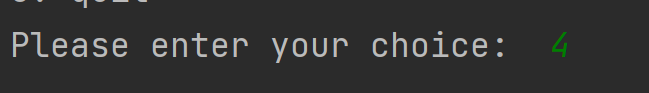
# User Interaction

This module provides users with preset some project database operation methods. Users can import csv files exported by other modules into the database through simple keyboard input.

After running, py file users can see the following output in the running window:



If the user is using the module for the first time. Please enter "4" to create the database.



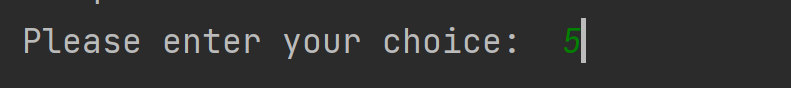
If there is no database file named "sim\_database.db" in the project path. The method will create a file in the project path.

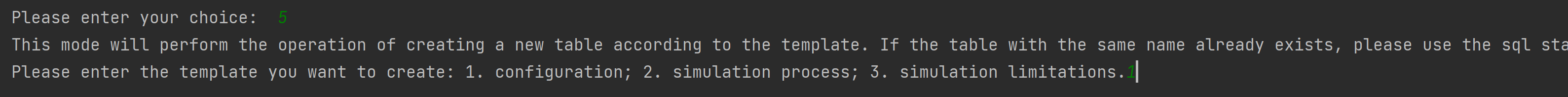
Note: If there is a file with the same name, this method will only connect to the database with that name. It will not overwrite the file with the same name.

The following is a prompt for successful creation:



## Create a table

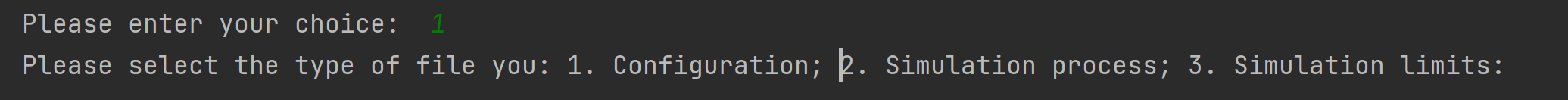




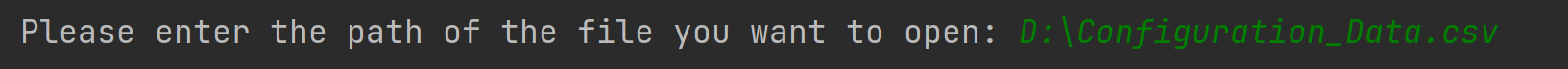
Before using the database, users need to create tables according to the template. After entering 5, the user enters "1", "2", and "3" respectively and press Enter to create a table based on the template in database.

## Import csv data

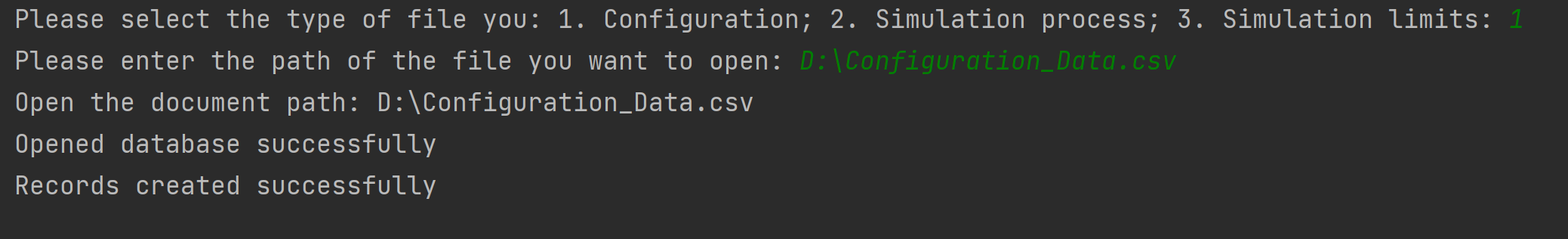
In the data import mode, users need to select the corresponding type according to the data they want to import. The wrong type selection will cause the import to fail.



The user needs to enter the path to import the csv file.



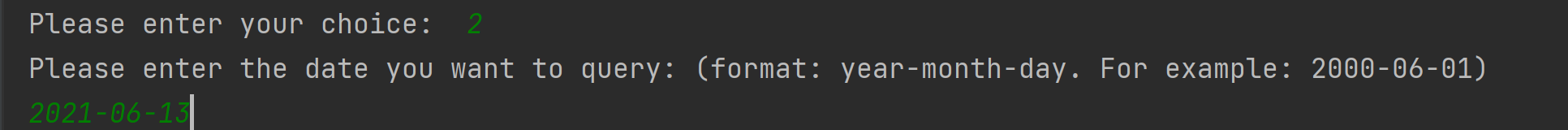
After the import is successful, the output box will display a message.

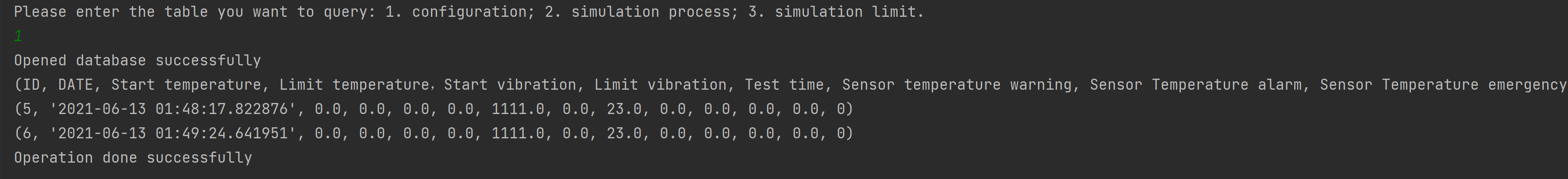


## Query data

Users can query the eligible data in this mode based on the year, month, and day information.

The format of year, month and day must meet the rule "year-month-day"



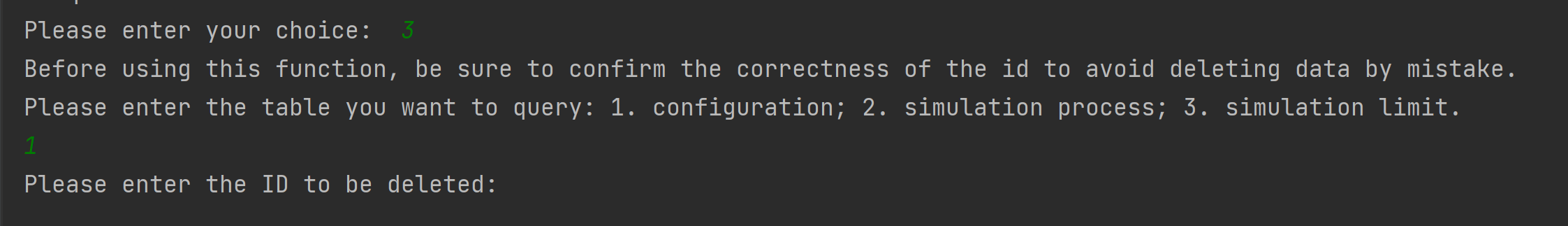


If only the title field is output, it means that the database does not have a record of that time.

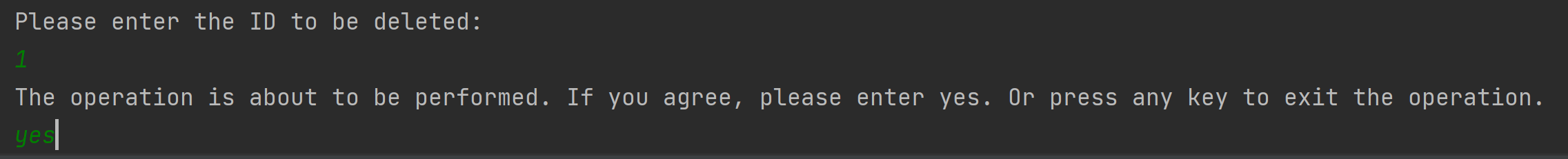
## Delete data

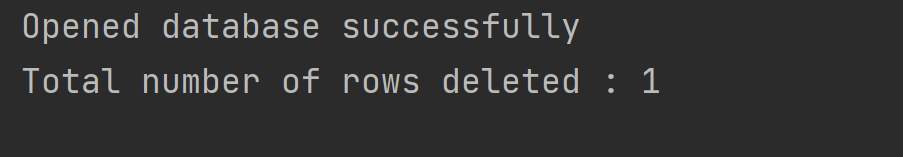
Please use this function with caution, once it is executed, it cannot be restored.

Before the operation, you need to obtain the id of the data you want to delete (you can use the search function to obtain the method)



Enter yes and press Enter to perform the operation.

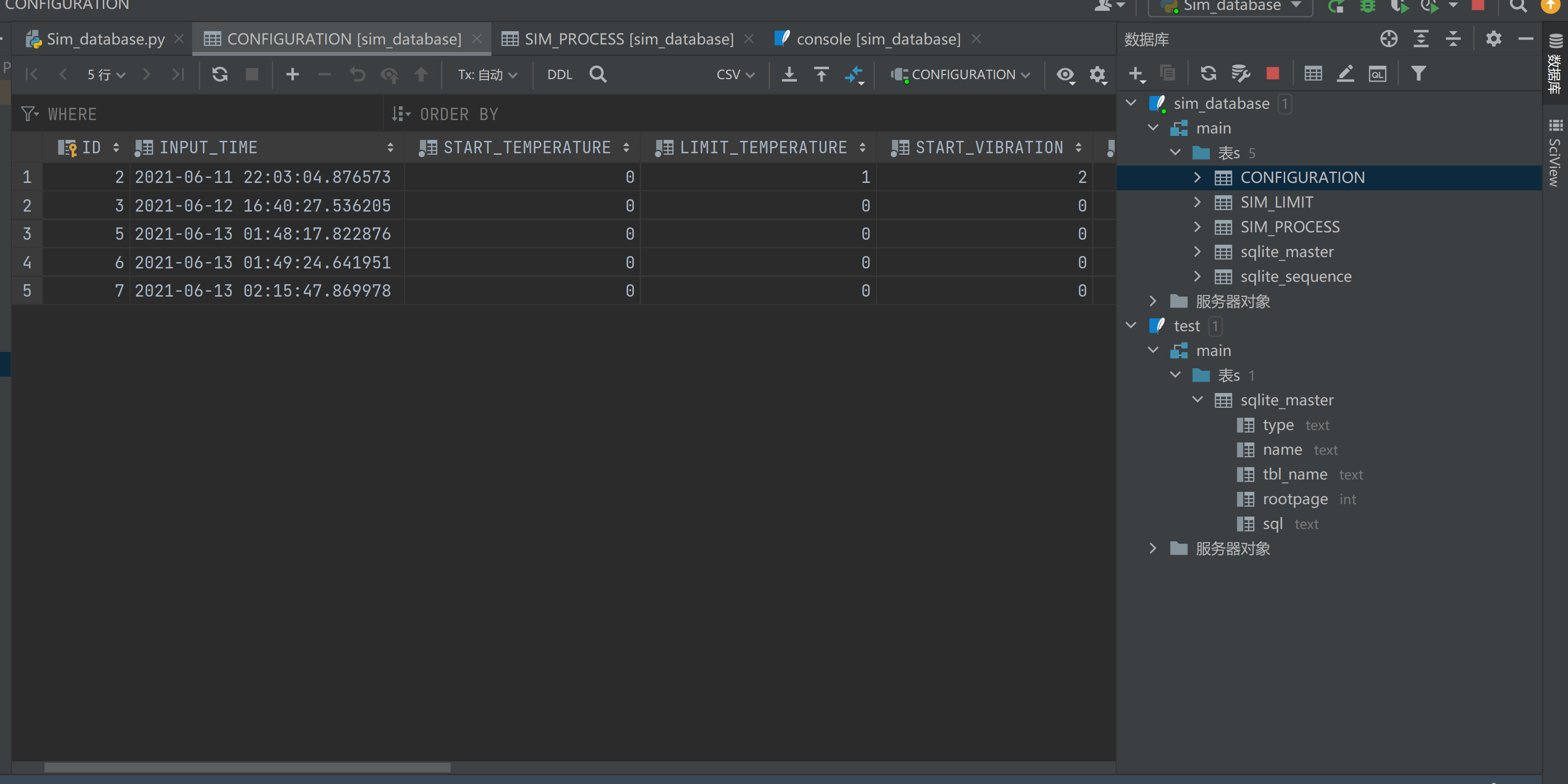




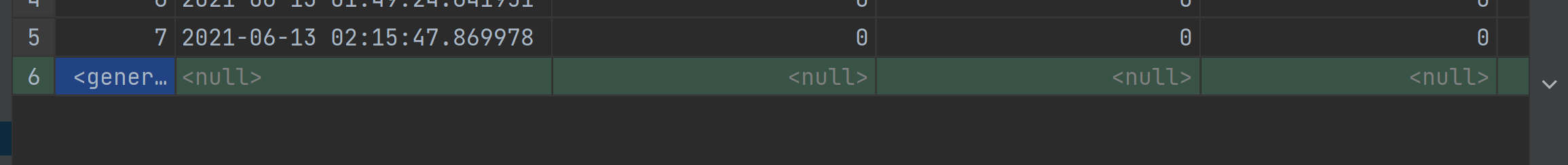
## Other methods of operating the database:

**Pychram:**

Using pychram, you can directly use the graphical interface to perform operations on each table in the database (such as adding, deleting, and modifying data).



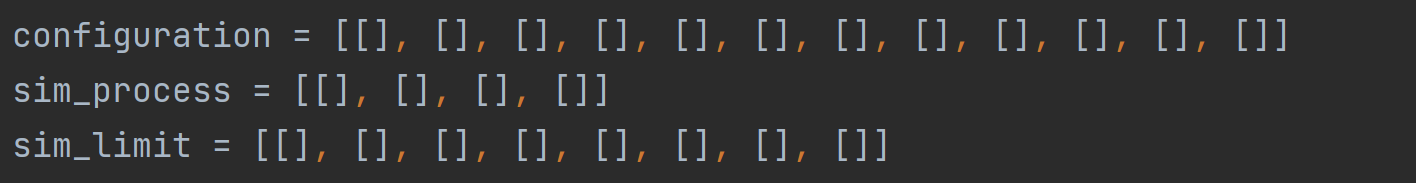
Enter data directly on the graphical interface to add it to the table.



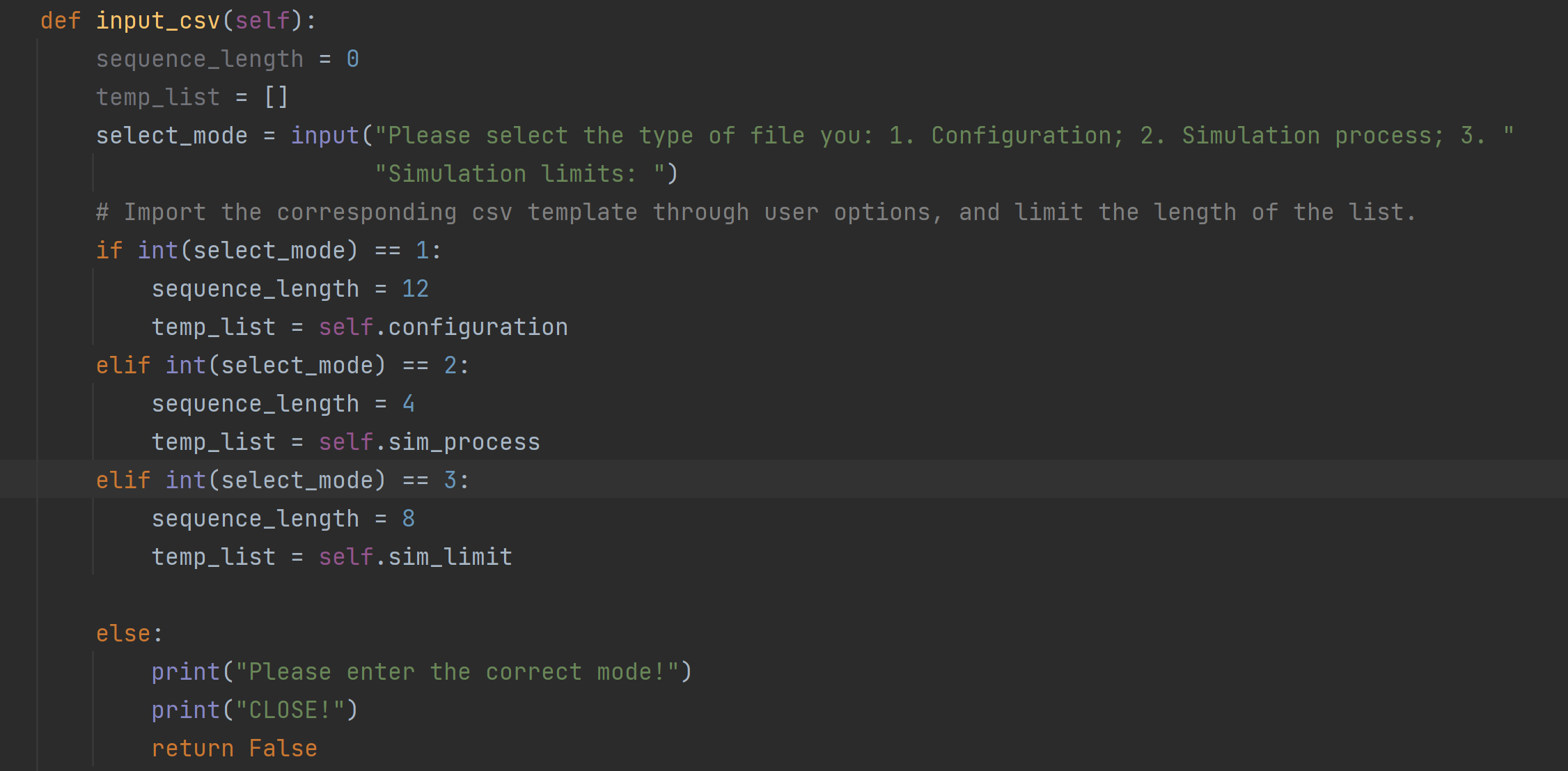
**Console& Database operating software**

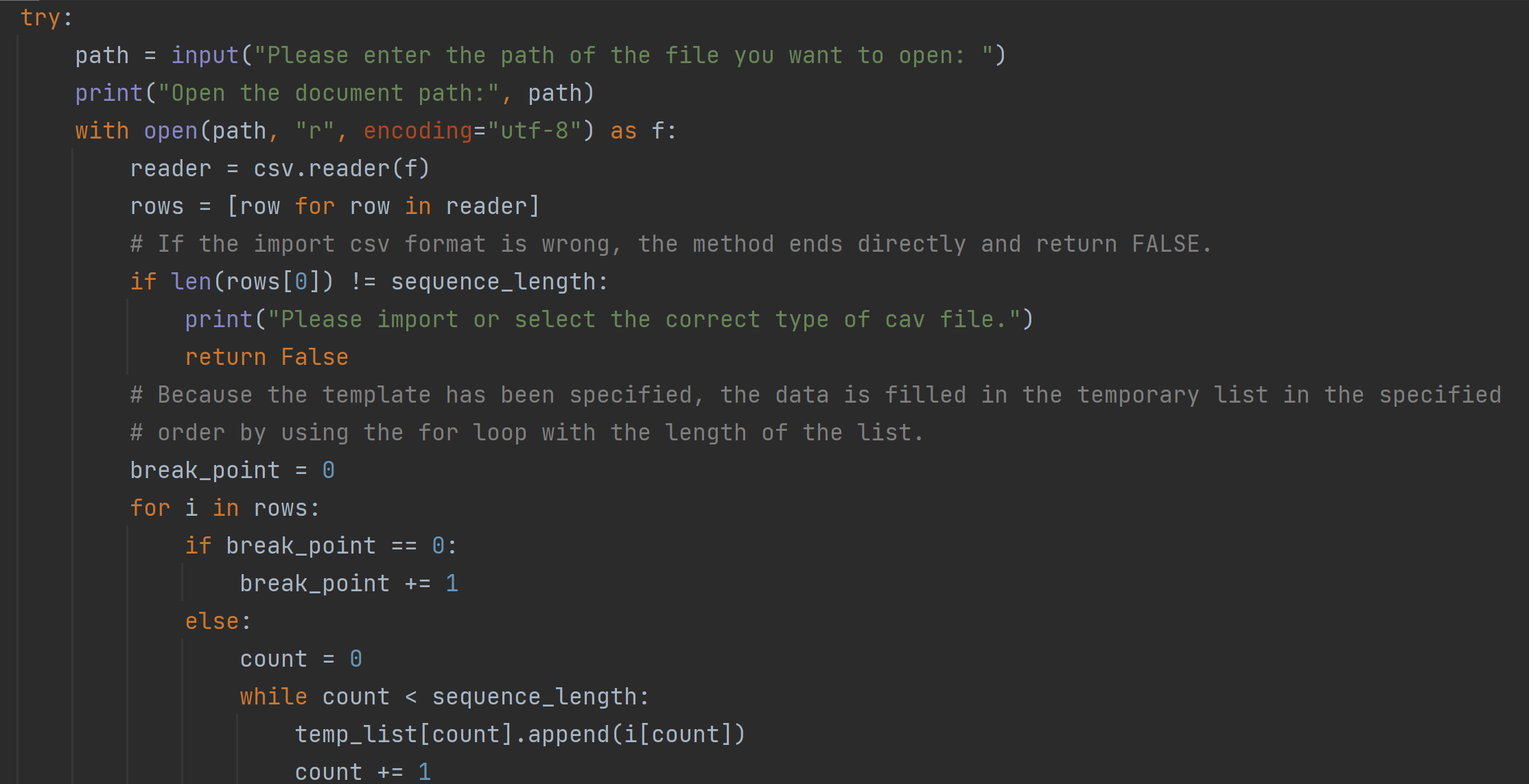
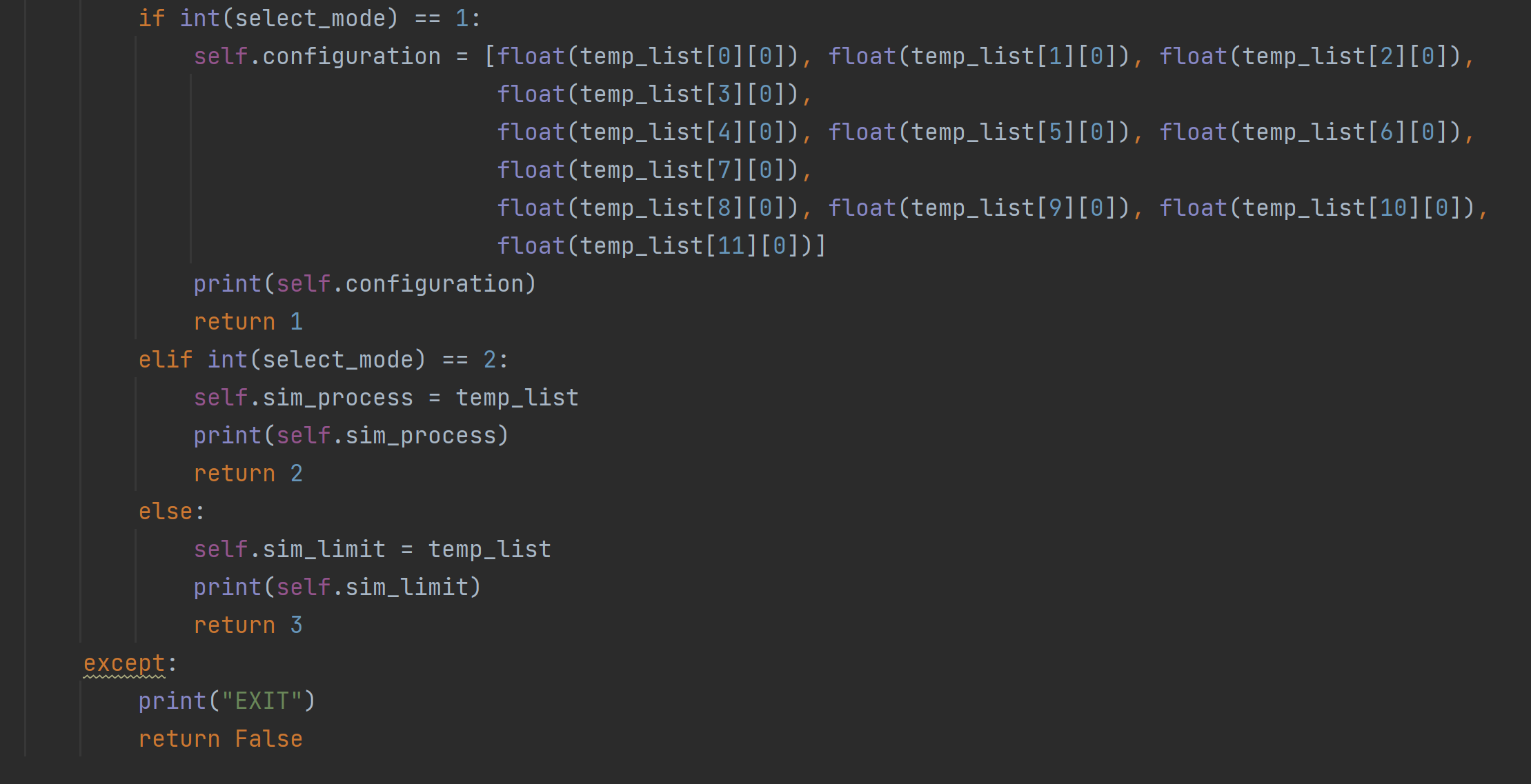
In the pychram database console, you can directly use sql statements to operate the database. Users can also use software such as oracle to operate the database.

# Code Implementation Components



Standardize the format of the imported data. The standardized format can be classified for different csv and database formats. For example: It can be judged that the data should be imported into the table of the simulation process through the import list of length 4.

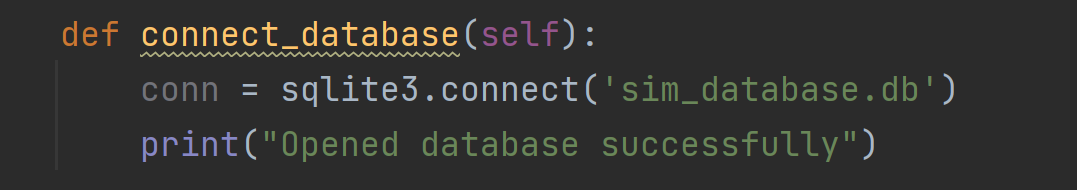


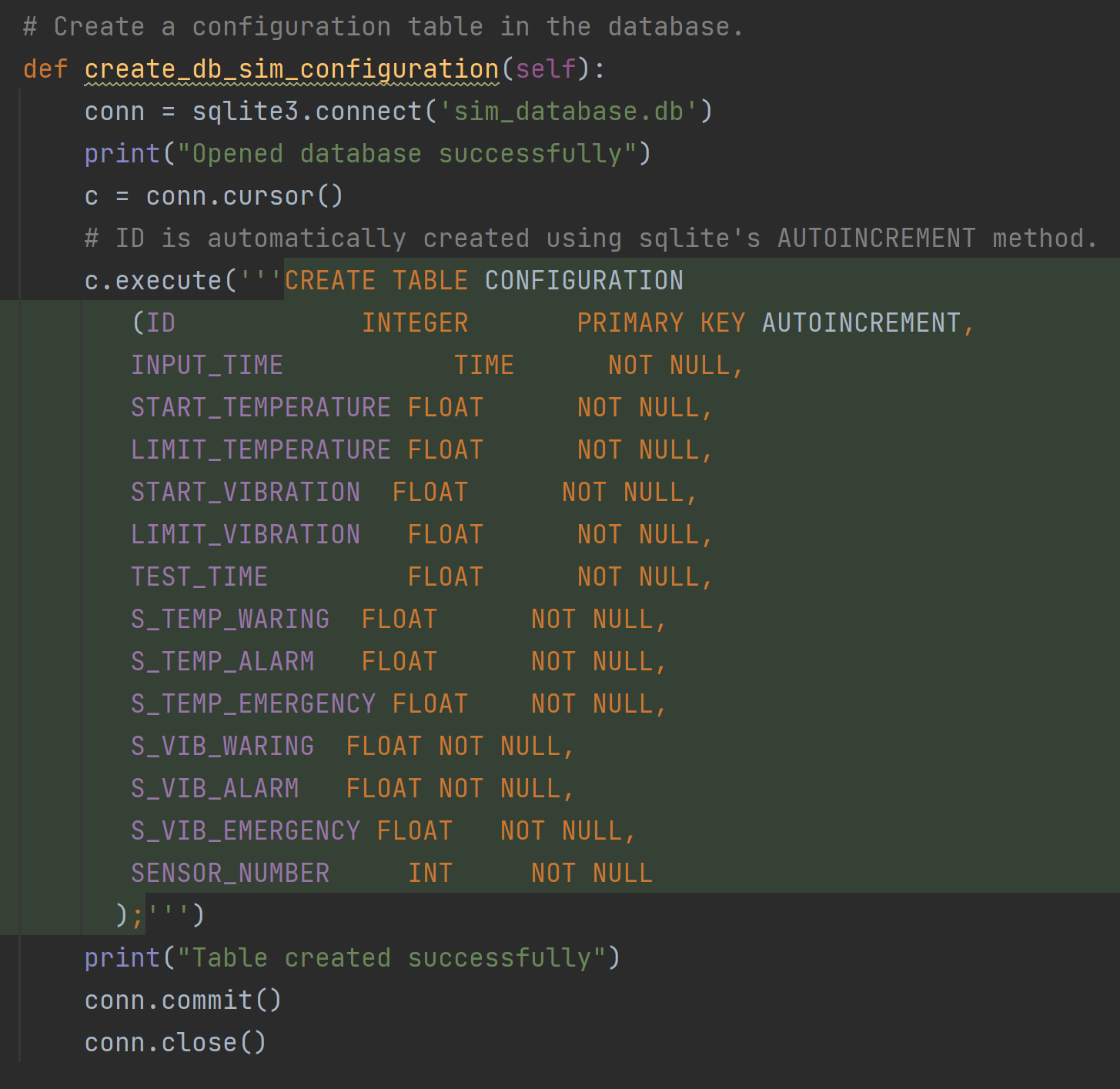
This class uses input() to guide the user to input the format specification of the imported data (csv).

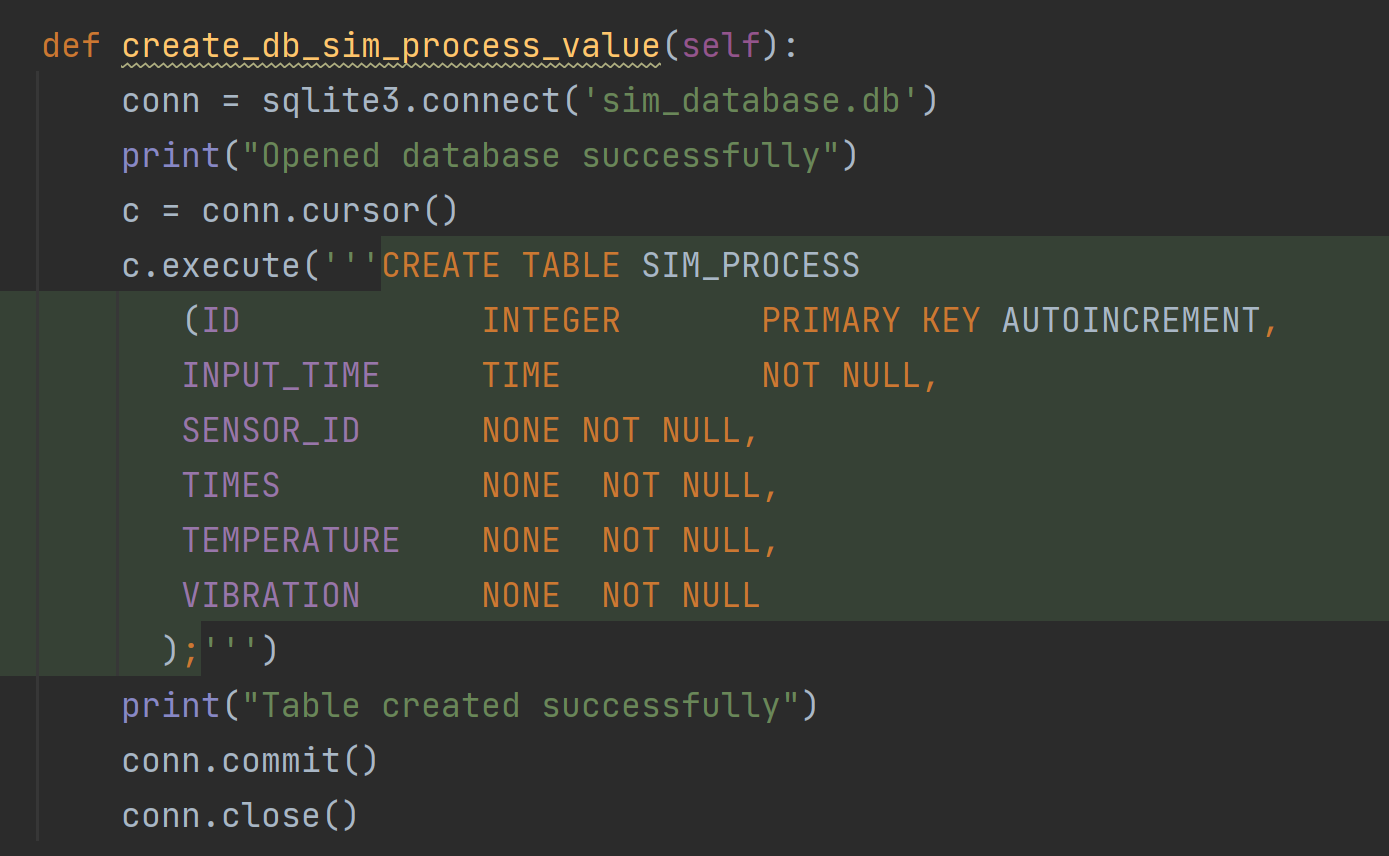
The method uses the open and reader methods to read the csv of the local path.

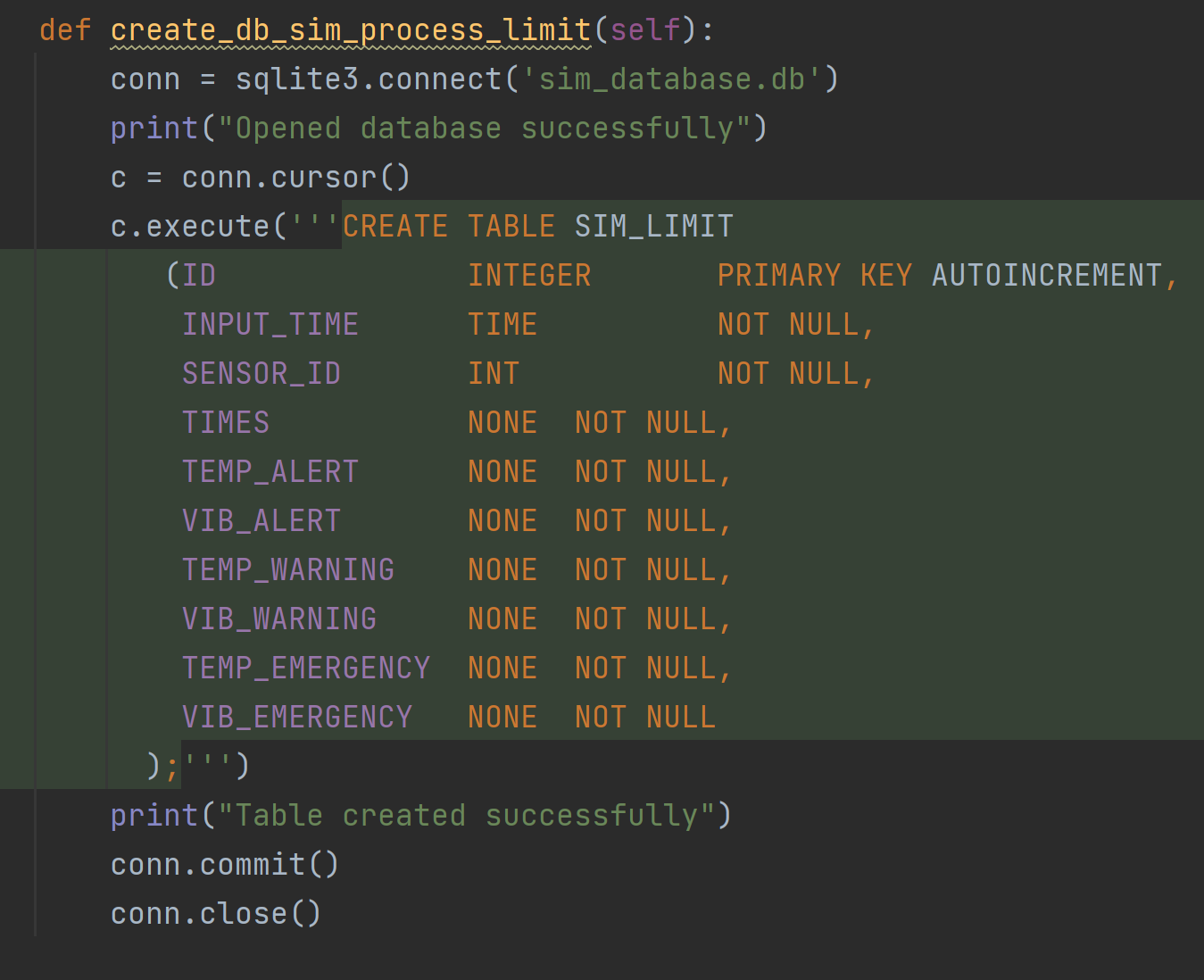
By standardizing the length, the csv data is imported into the list in sequence in units of columns. And write to the class variable. The return value is FALSE (import failed) and the number 123 (representing the imported csv type).



Sqlite3.connect() method can connect to the db file of the same name sqlite3 in the project directory. If there is no database with the same name, create a new database.







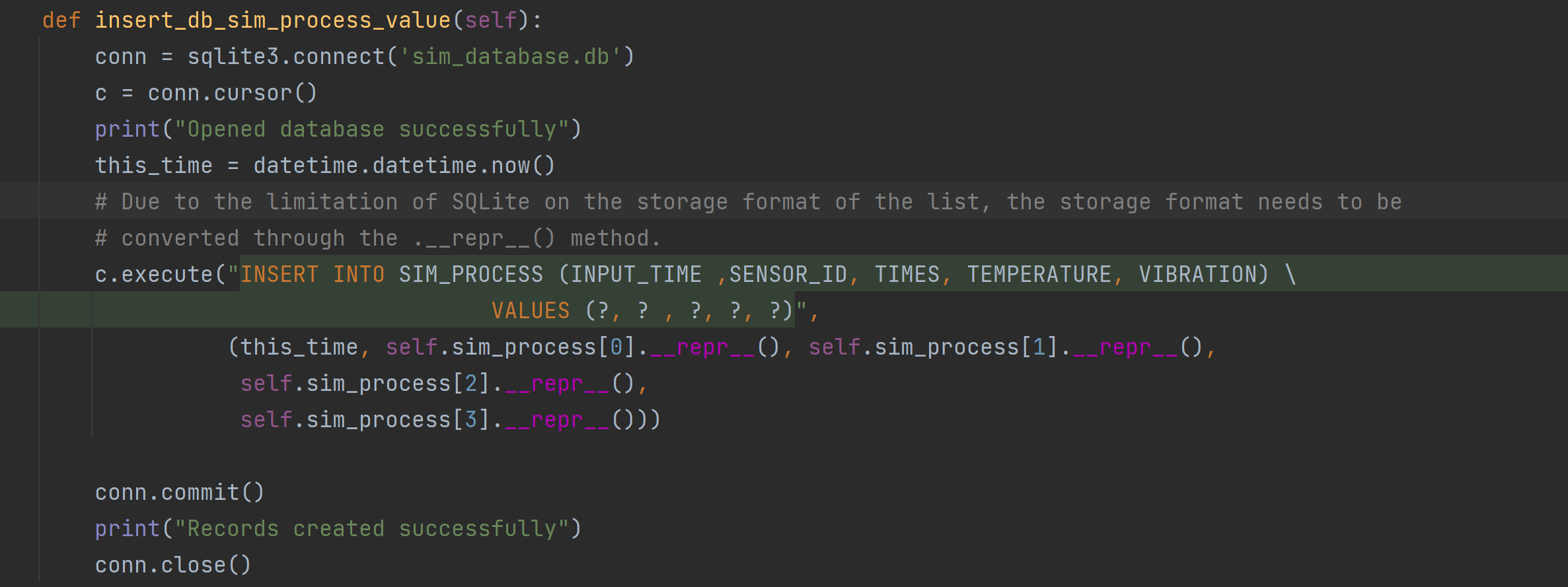
The method is used to create a table of a specific template in the database for the user.

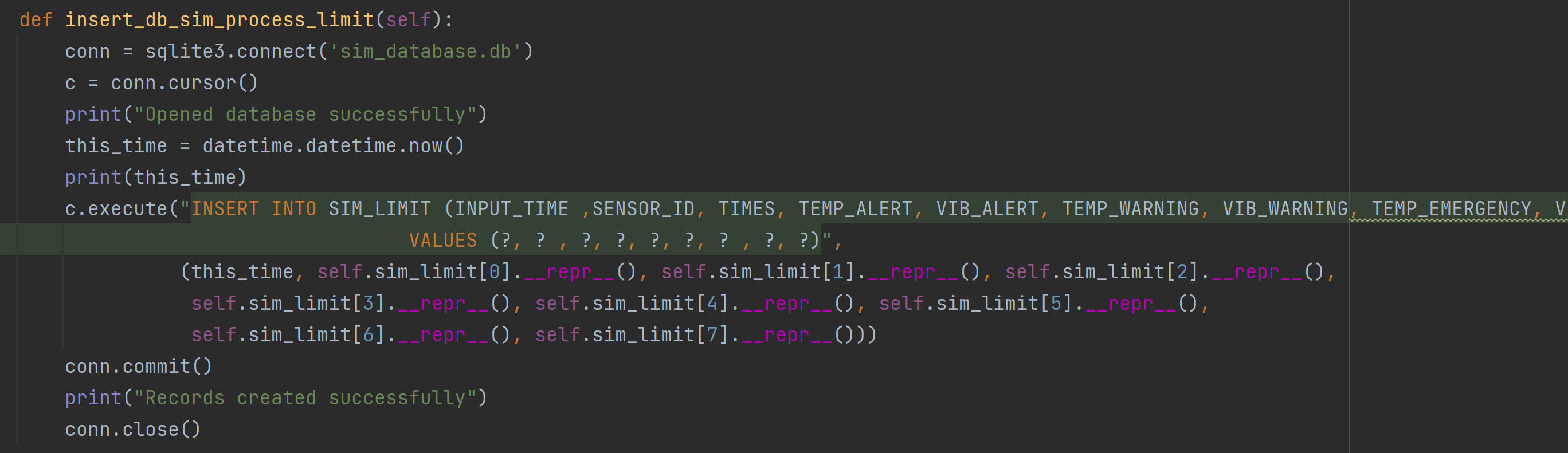
**cursor()** creates a cursor for this routine. After that, **execute** (sql [, optional parameters]) executes the statement in it through the cursor. Note that the statements inside are SQLite statements. Users need to use "?" and ":" placeholders to import parameters.

**.commit()** is used to commit the current transaction to make changes to the database.

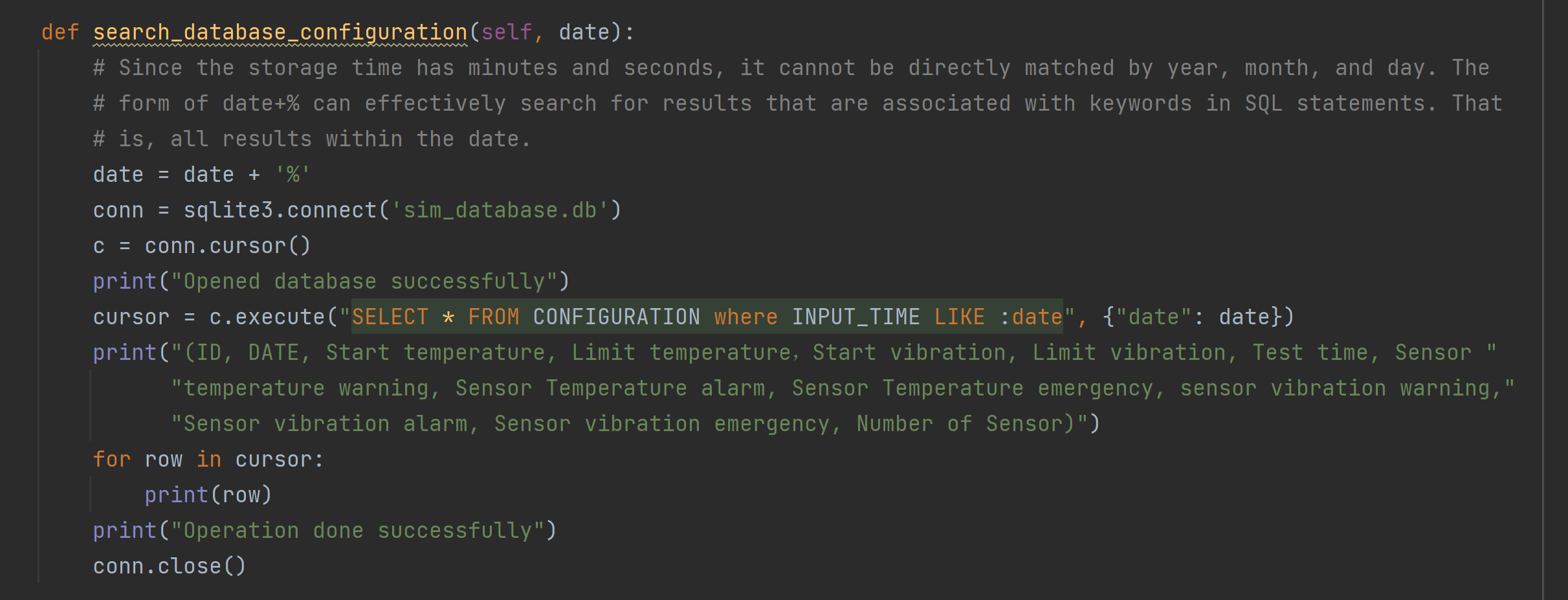
In addition, the parameter **AUTOINCREMENT** in SQLite can make the primary key in the table add ID to the data in order as new data is inserted.

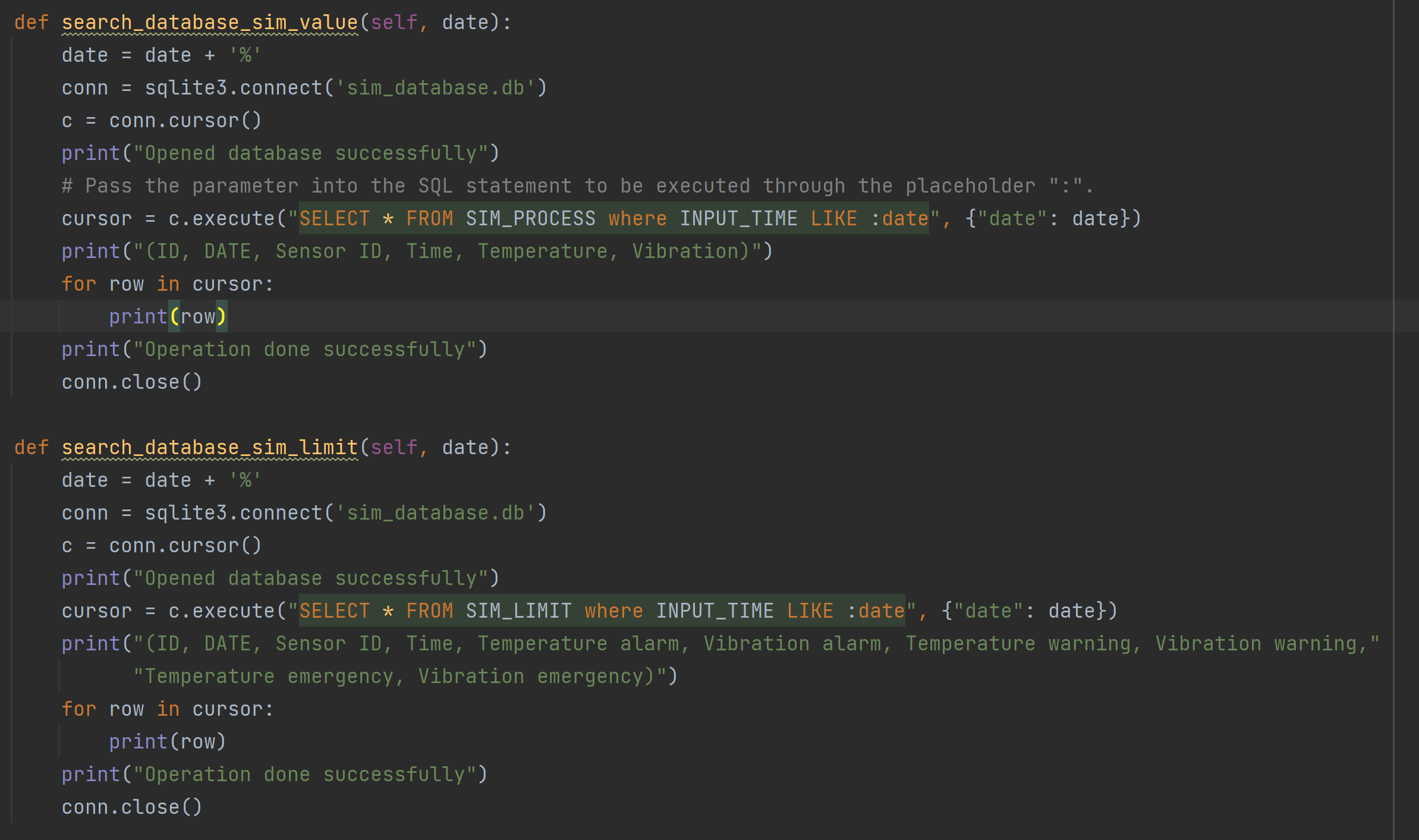




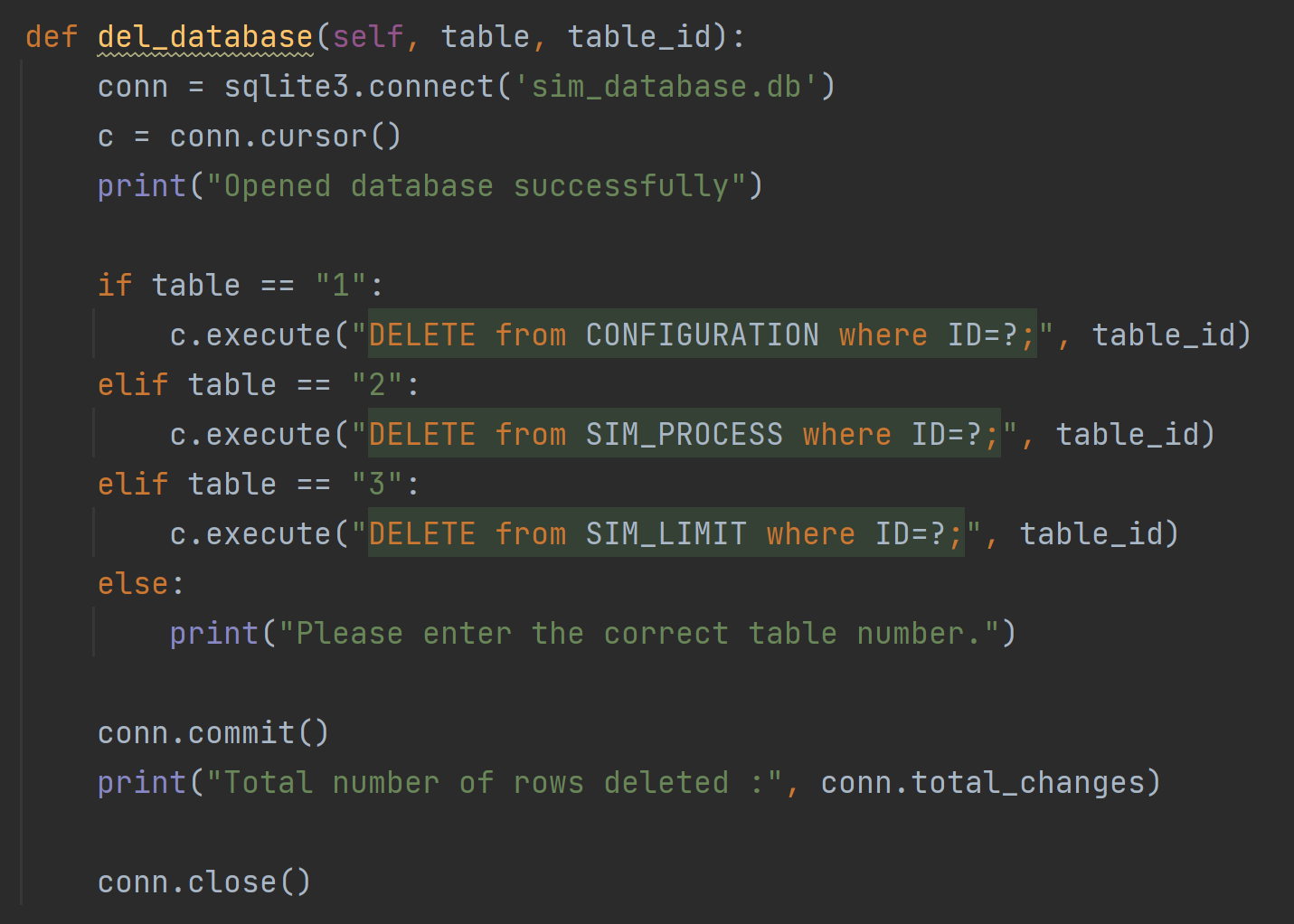


The function of the insert method is to import the csv data variable in the standardized format into the database. For the list, you need to use the .\_\_repr\_\_() method to make it conform to the SQLite storage variable specification.

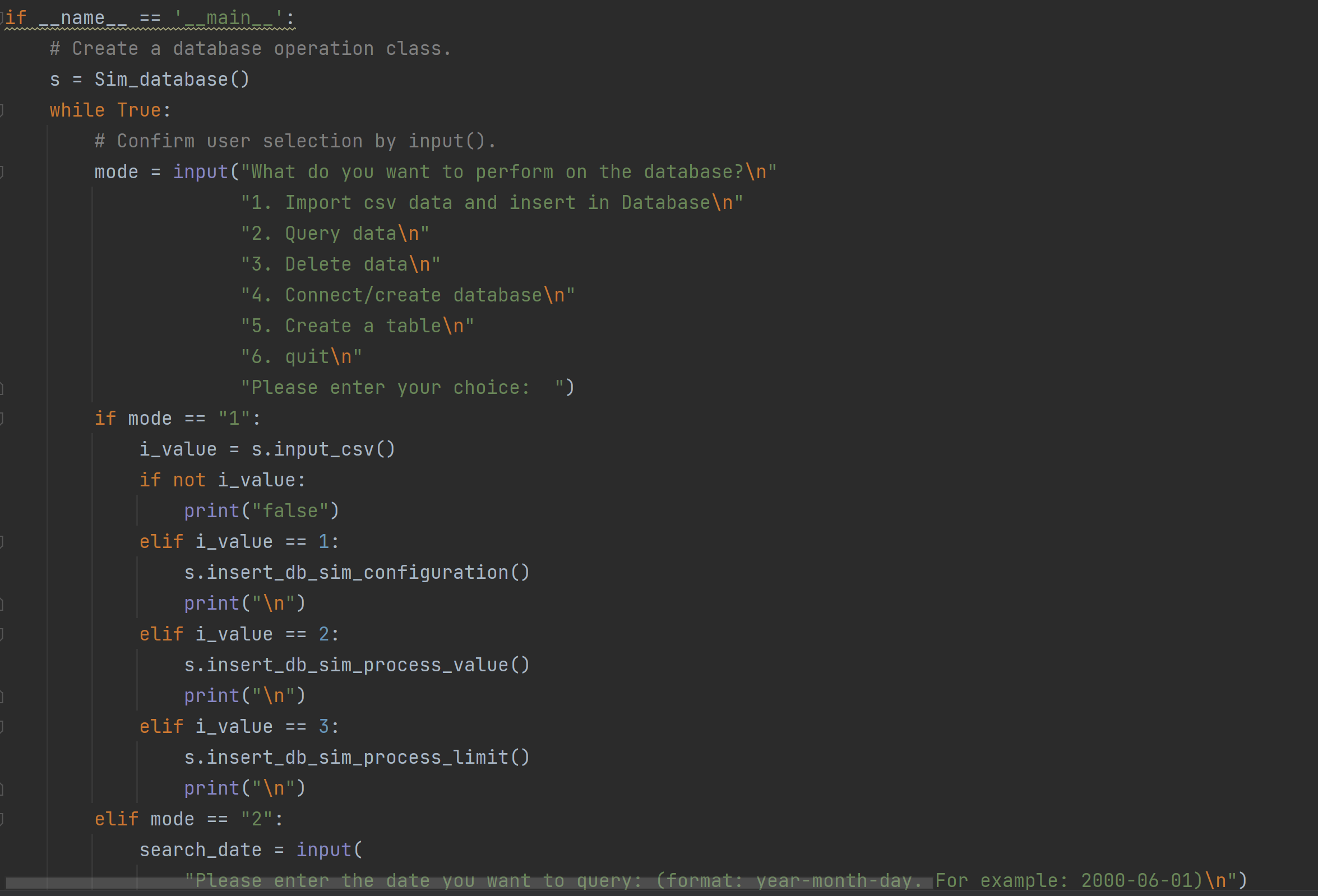


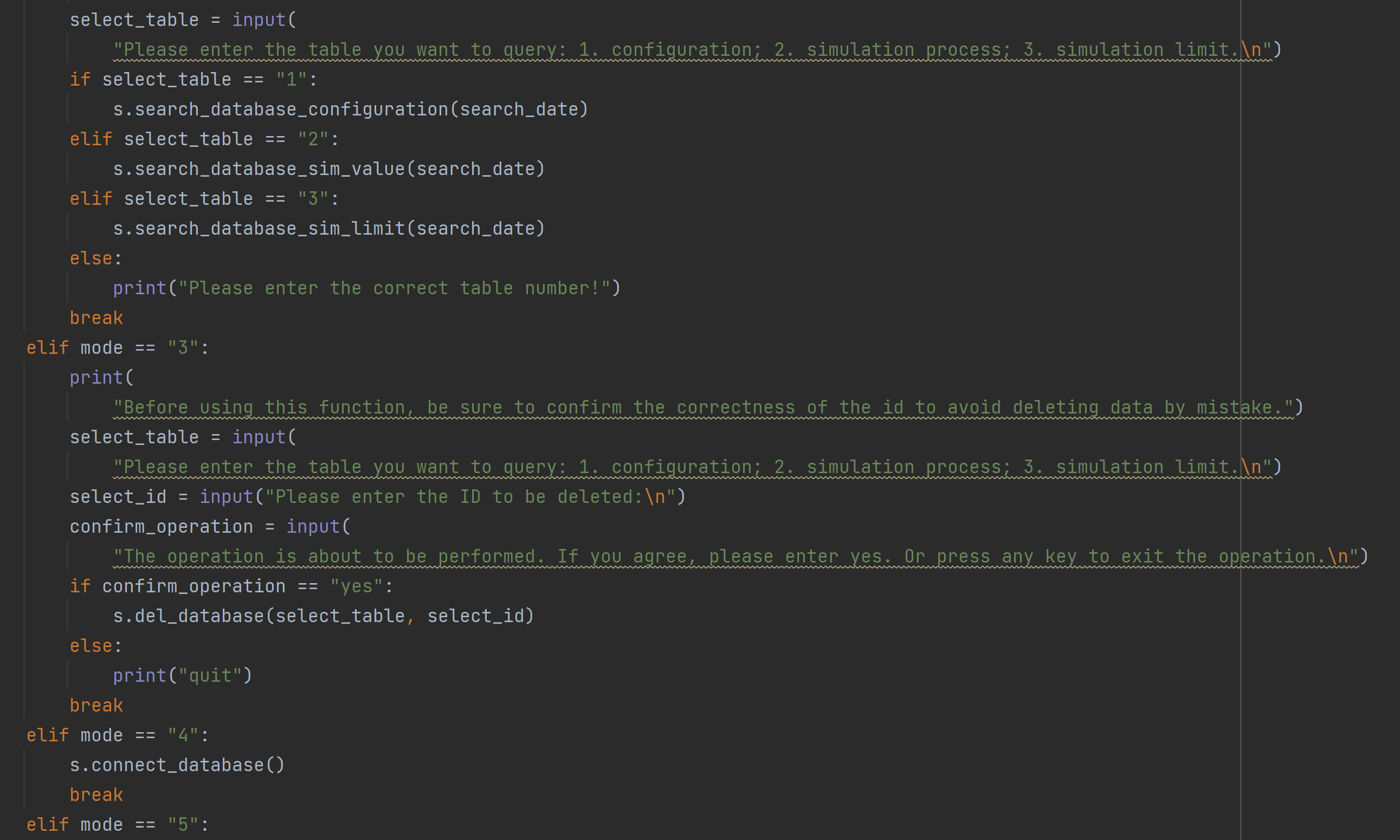


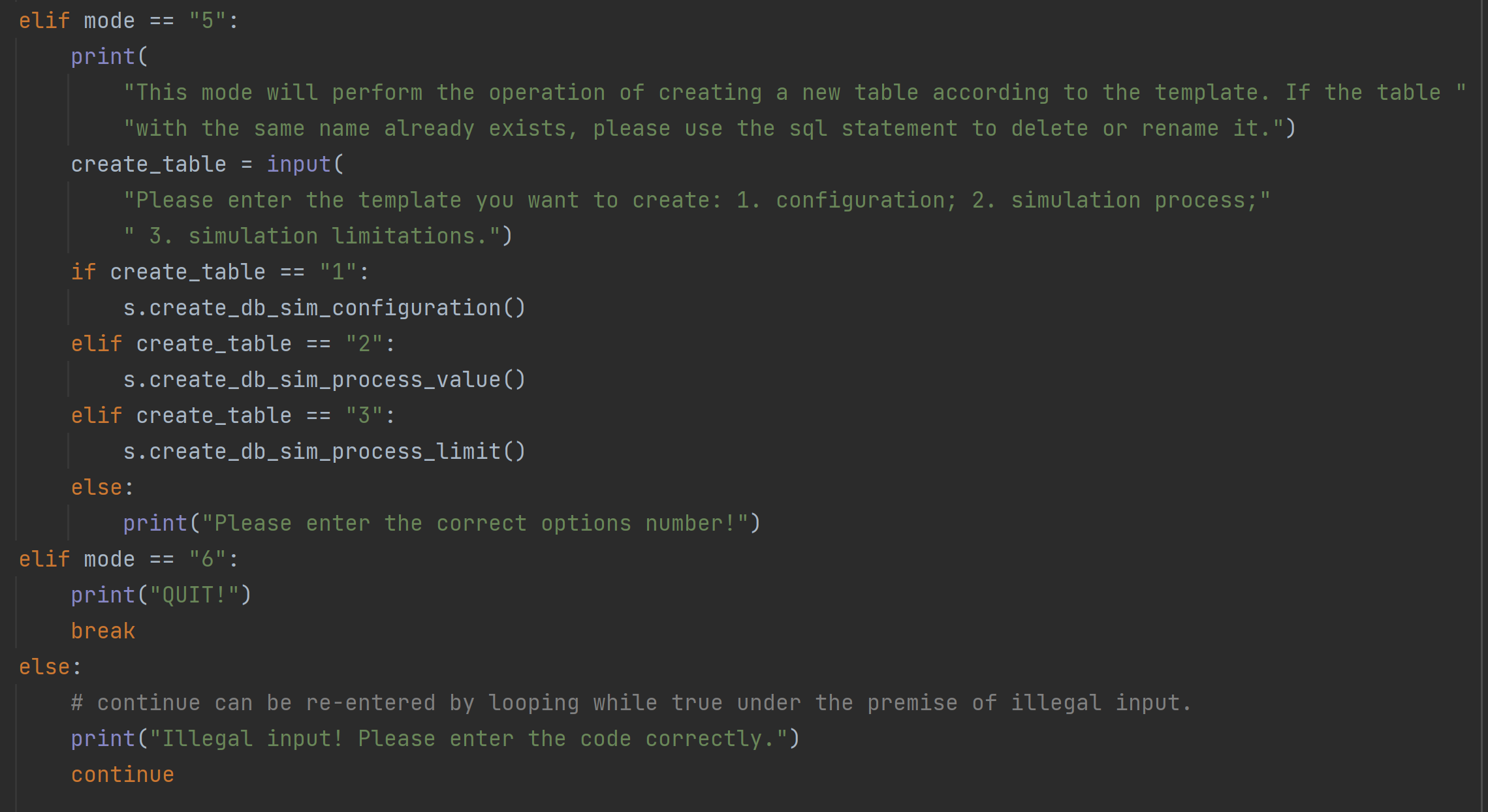
Where and like statements are used in the search method, the corresponding records in different tables are searched through the "year-month-day" entered by the user. The "value%" format simplifies user input. The user can ignore the minute and second record and then query it.



Through the id provided by the user, use DELETE to delete the data corresponding to the specific ID in the table.







Use input() to get user input. Guide users to perform simple database operations through text reminders and if branches.