CSV to DB converter

A project report

submitted by

Mohd Zaid

(Class 12)

under the supervision of

Mrs. Swati Saxena

in completion of project
submitted as yearly project
For class 12.

to the

तत् त्वं पूषन् अपावृणु
केन्द्रीय विद्यालय संगठन

Kendriya Vidyalaya
Andrews ganj
November 2022

Kendriya Vidyalaya ANDREWS GANJ NEW DELHI

DECLARATION

I hereby declare that the work reported in the project titled **"CSV to DB converter"** submitted for the fulfillment of yearly project, is record of my work carried out under the supervision of *Mrs. Swati Saxsena*.

Mohd Zaid
(Class 12)
Kendriya Vidyalaya
Andrews Ganj ND
November 2022

KENDRIYA VIDYALAYA ANDREWS GANJ NEW DELHI

CERTIFICATE

It is certified that **Mohd Zaid** has submitted project under my supervision for fulfillment of Project on the topic **“CSV to DB converter”**. It is further certified that the above candidate has carried out the project work under my guidance during the academic session 2022-2022.

Mrs. Swati Saxsena
PGT Conputer Science
Kendriya Vidyalaya
New Delhi - 110025

ACKNOWLEDGEMENT

As a matter of first importance, I offer my genuine thanks to my supervisor Mrs. Swati Saxsena, PGT CS, Kendriya Vidyalaya. I appreciate his support and help during the project work. Own Ideas/Acknowledgement

Mohd Zaid
(Nov 2022**)

Contents

Chapter 1

Introduction

This is report of project named **CSV to DB converter** which is to be submitted as my yearly project for class 12.

1.1 Goals

Through this project i will be attempting to make a program which takes a **CSV**`(Comma separated values)` file as input and copies its content to `mySql Database` format.

1.2 Preliminaries

Some basic knowledge are required to build and understand this program.

1. **mySql** to edit tables and databases.
2. **Python** to code basic functionalities
3. **Csv** to read data from file.csv file.
4. **Loops** to iterate through rows.

Chapter 2

Header files and their uses

2.1 Frontend.stylers

It is a local package. Made with intention to provide Interface inside terminal.

It is imported using command

```
from Frontend.Stylers import Console, SQL
```

Both Console and SQL have calling functions which display respective informations.

2.2 Csv

This module is imported to read data from CSV file.

```
file data = csv.reader( csv file name.csv )
```

2.3 sqlite3

This module is imported to Create connection to on-disk file and add entries to it.

```
connection = sqlite3.connect( file name.db )

cursor = connection.cursor()

commands = cursor.execute( sqlite commands )
```

2.4 os

This module is imported to Check if file that has been provided by user exists or not. If not then shows error, continues otherwise.

```
        if not os.path.exists( csv file path.csv ) : ...
```

2.5 mySql

This module is imported to Create connections to localhost database and execute commands in there.

```
        try: import mysql except:ModuleNotFoundError:...
```

this is imported in try and except block as it may not be present on user's system or maybe not correct version.
If it is present then again

```
        connection=mysql.connector.connect(
                                            host='localhost',
                                            passwd=dbpasswd,
                                            dbname=dbname )


        cusrsor = connection.cursor()
```

All again in try and Except block because host may not be present or password may be wrong.

Chapter 3

Code

3.1 Doc-strings and file information.

Explains code inside file.

```
"""
    This programme converts csv entries into database entries.
        -   User can choose to use server database or file ‘.db‘.
        -   Also supports xls to db entry in future.
"""

# Info
__version__ = 1
__author__  = ’Zaid’
```

3.2 Imports and exception handling

```
# Imports from other files.
from Frontend.Stylers import Console, SQL

# Imports from inbulit library.
import csv
import os
import sqlite3

# Dependent Imports
# If mysql.conncetor (python) is not installed.
# Download - python pip install mysql-connector-python
try:
    import mysql.connector
except ModuleNotFoundError as MNFErr:
    ...
```

3.3 Menu

```
options_1 = [
'\nWhere do you want to make entry:',
'*   [f] -  To a "db" file using sqlite.',
'*   [d] -  To a existing "db" in system.\n']
DB_TYPE: str = Console.Input(*options_1, sep='\n')
```

3.4 Navigation and Inputs

```
# making connection and cursor
if ’f’ in DB_TYPE.lower():
    # Gathering informations.
    csv_file_path:str = Console.Input(...)
    db_file_name:str  = Console.Input(...)
    db_table_name:str = Console.Input(...)
    # Checking informations.
    if not os.path.exists(’\\Database’): os.makedirs(’\\Database’)
    # Processing to transfer data
    connection = sqlite3.connect(f’Database\\{db_file_name}.db’)
    cursor     = connection.cursor()
elif ’d’ in DB_TYPE.lower():
    # Gathering informations
    csv_file_path:str = Console.Input(...)
    db_passwd   :str  = Console.Input(...)
    db_name     :str  = Console.Input(...)
    db_table_name:str = Console.Input(...)
    # Creating cursor
    try:
        connection = mysql.connector.connect(...)
        cusrsor = connection.cursor()
    except NameError:
        Console.Error(...)
        Console.Log(’Using sqlite’)
        connection = sqlite3.connect(f’Database\\{db_name}.db’)
        cursor     = connection.cursor()
    except Exception as Exc:
        Console.Error(...)
        Console.Log(’Using sqlite’)
        connection = sqlite3.connect(f’Database\\{db_name}.db’)
        cursor     = connection.cursor()
```

3.5 Commands

```
# All commands
CREATE_TABLE = ’’’CREATE TABLE IF NOT EXISTS {} (
                       {});’’’

COLUMNS = ’VARCHAR,\n\t’

INSERT_INT = ’’’INSERT INTO {}
                       VALUES ("{}")’’’

VALUES = ’","’
```

3.6 Entry loop

```
# Convertor
with open( csv_file_path, ’r’ ) as csv_file:
    file_data = csv.reader( csv_file )

    for index, data in enumerate(file_data):
        try:
            if index == 0:
                cursor.execute( CREATE_TABLE.format( ..., ... ) )
                continue
            cursor.execute( INSERT_INT.format( ..., ... ))
        except Exception as Exc:
            Console.Error( Exc )

    connection.commit()
```

Chapter 4

Uses

This program can be used to transfer large tabular entries from csv files to school database.

Bibliography

[1] W3Schools Tutorial on SQL used to learn and implement basic SQL codes.

[2] W3Schools Tutorial on Python used to learn and implement basic Python codes.

[3] W3Schools Tutorial on Python file handling used to learn and implement basic file handling codes.