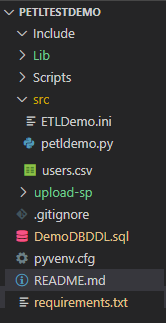
**ETL PROCESS USING PYTHON PETL LIBRARY**

**Objective:** Develop ETL process using Python opensource library to extract, transform and load from API call to SQLServer Database table.

**Requirements**:

1. **SQLServer Instance with Database and Table**
2. **VisualCode**
3. **Python VirtualEnvironment**

**Project structure:**

****

**Requriements.txt file for Python libraries installation**

* petl
* pymssql
* requests
* pandas
* mysqlclient

**Readme.md**

**1. Create virtual environment using below command**

**python -m venv petltestdemo**

**2. Once created the environment activate it.**

cd scripts

activate

**3. Go to root folder and open the folder in VisualStudio with below command**

code .

**4. Run below command to install the python lirbaries as needed for the process.**

pip install --trusted-host pypi.org --trusted-host pypi.python.org --trusted-host files.pythonhosted.org -r requirements.txt

**5. Create a database and table with user and permissions on SQLServer mentioned in DeoDBDDL.sql file**

CREATE DATABASE ETLDemo

GO

USE ETLDemo

DROP TABLE IF EXISTS users

CREATE TABLE [dbo].[users](

[id] [int] NOT NULL,

[Name] [nvarchar](50) NOT NULL,

[Username] [nvarchar](50) NOT NULL,

[Email] [nvarchar](50) NOT NULL,

[Phone] [nvarchar](50) NOT NULL,

[City] [nvarchar](50) NOT NULL,

[Street] [nvarchar](50) NOT NULL,

[Suite] [nvarchar](50) NOT NULL,

[Latitude] [float] NOT NULL,

[Longitude] [float] NOT NULL

) ON [PRIMARY]

GO

USE [ETLDemo]

GO

CREATE USER [petluser] FOR LOGIN [petluser] WITH DEFAULT\_SCHEMA=[dbo]

GO

USE [ETLDemo]

GO

ALTER ROLE [db\_owner] add MEMBER [petluser]

GO

**6. Update the ini file for the ETL process as needed.**

[CONFIG]

url = https://jsonplaceholder.typicode.com/users

server = xxxxxxxx

database = ETLDemo

user = petluser

password = XXXXX

**7. Check the API call**

https://jsonplaceholder.typicode.com/users

**8. Develop the application python code using petl lirbary for ETL process and run it to import data from api to SQLServer table created.**

**Fromdicts**: View a sequence of Python dict as a table.

**Unpackdict:** Unpack dictionary values into separate fields.

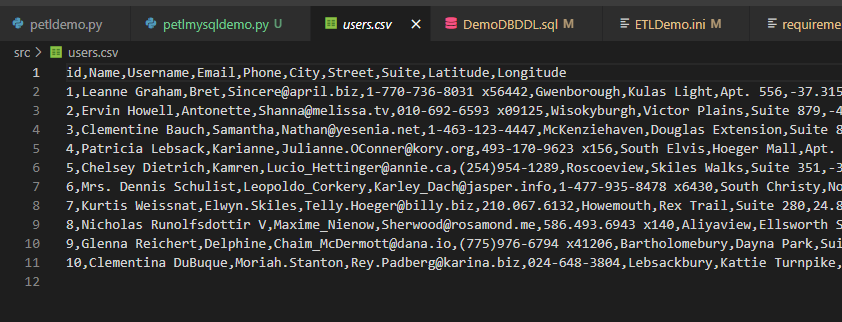
**Cut:** Choose and/or re-order columns.

**Rename:** Replace one or more fields in the table’s header row.

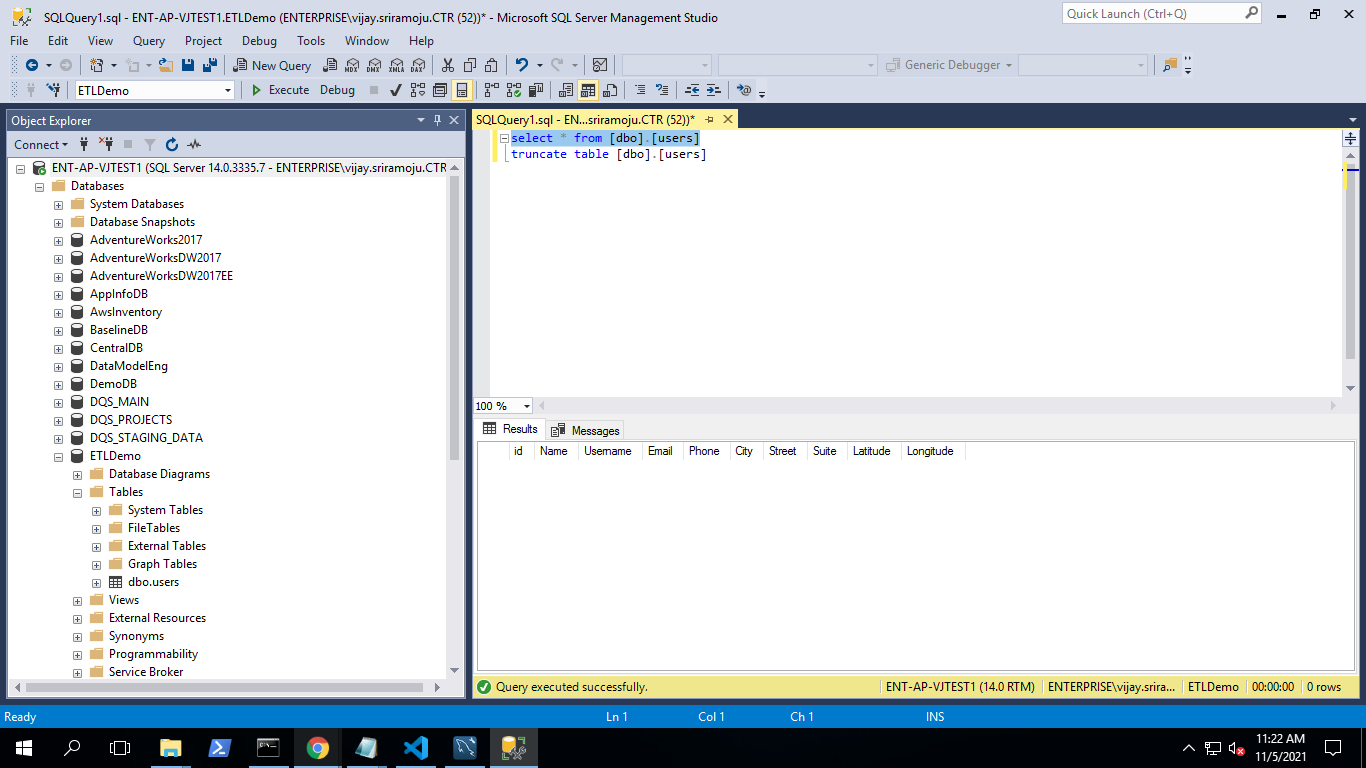
**petl.io.csv.fromcsv :** read CSV file.

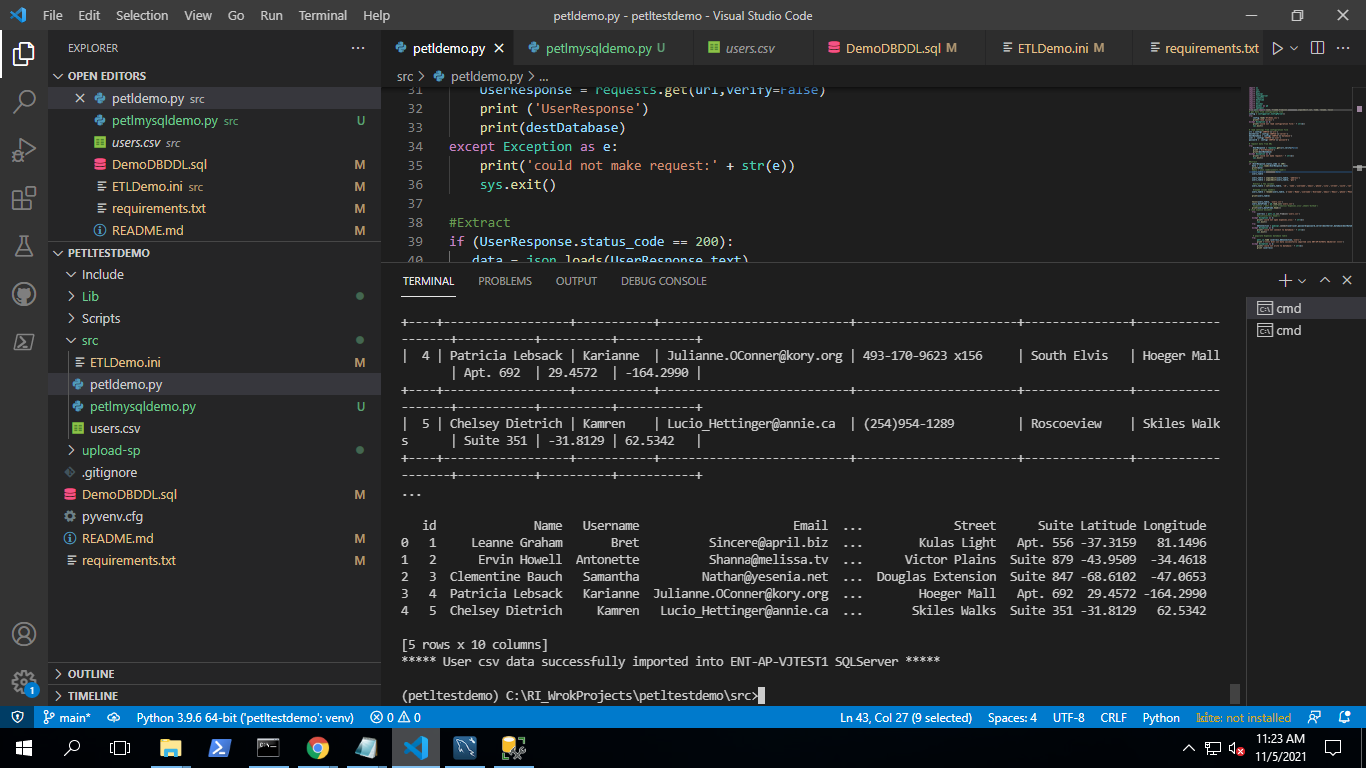
**petl.todb :** load data to database.

**9. Check the csv file imported from api.**

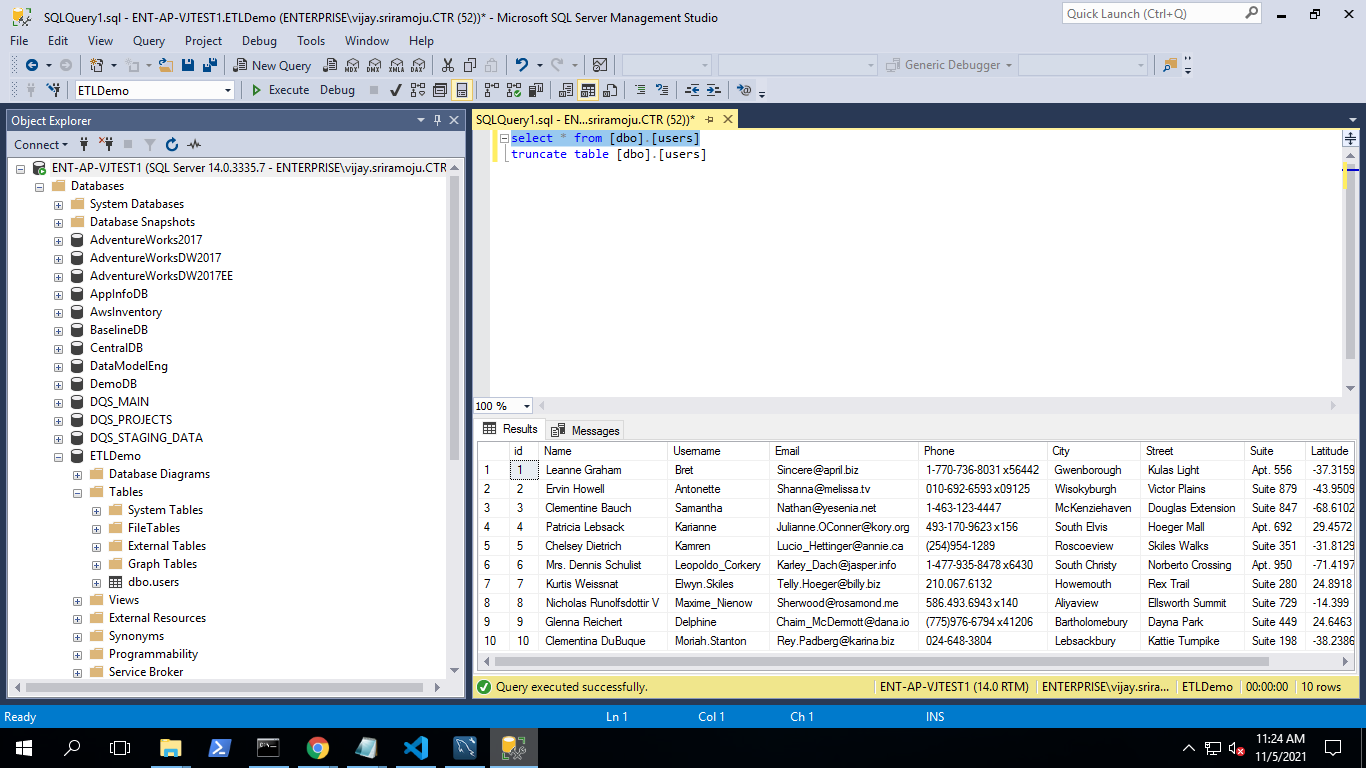
****

**10. check the exported data from csv to SQLServer.**



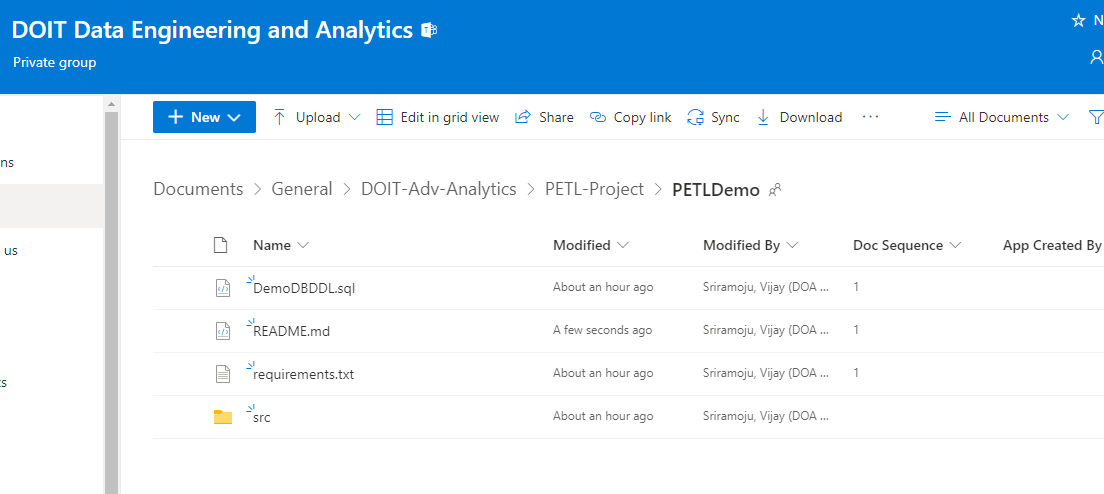


**Exported data into SQLServer:**



**11. Uploaded files to the SharePoint.**

[**https://rigov.sharepoint.com/:f:/r/sites/ETSS-DataEngineering-Analytics-Team/Shared%20Documents/General/DOIT-Adv-Analytics/PETL-Project/PETLDemo?csf=1&web=1&e=OdFo3P**](https://rigov.sharepoint.com/:f:/r/sites/ETSS-DataEngineering-Analytics-Team/Shared%20Documents/General/DOIT-Adv-Analytics/PETL-Project/PETLDemo?csf=1&web=1&e=OdFo3P)

****