AMProduction presents you the set of ETL tools that allow you to create a date dimension table and load the data from the different CSV source files located in different places. Through a set of config files, you could change the table structure and a location of a source file.

System requirements:	
Hardware	
RAM	Minimum: 1 GB RAM
	Recommended: 4 GB RAM
Processor Speed	Minimum: x64 Processor: 1.4 GHz
	Recommended: 2.0 GHz or faster
Software	
OS	Windows 10 TH1 1507 or greater
	Windows Server 2016 or greater
Microsoft SQL Server 2019	
Python 3.8 and newer	
pyodbc Python library	
pandas Python library	

Changelog

1. v.2.0 beta 2021/01/30
   1.1. SQL script to create a date dimension table:
      a) Added the new staging table to perform ETL procedures
      b) Added 8 new columns
      c) Added [RowId] column as a Surrogate Key
   1.2. ETL_Tools.py script:
      a) Deploying data from the source CSV file to the new staging table to perform ETL procedures
      b) Added a simple ETL procedure
      c) Transferring data from the staging table to the main table
2. v.1.0 beta 2021/01/20
   2.1. SQL script to create a simple Database
   2.2. SQL script to create a date dimension table
   2.3. Database_Tools.py script to create empty the DATE_DIM table with error handling
   2.4. ETL_Tools.py script to deploy data from the source CSV file right to the DATE_DIM table with error handling

Manual

Notice. You should get access and rights to change the Database before (!) use the tools.

Steps before. Change the config files (if need).

Notice. Do not change the names of files.

Step 1. Create DB (If need). We pleasant present you the SQL script which allows you to create a simple database.

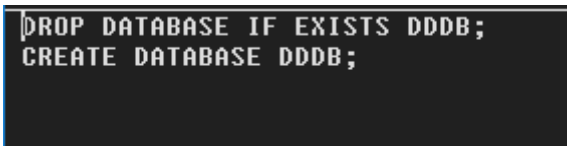

Figure 1: CREATE_DATABASE.sql

Step 2. Change the database credentials

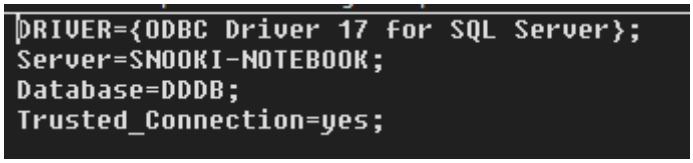

Figure 2: DB_config.cfg

Step 3. Change the table columns (if need)

```
USE DDDB;
DROP TABLE IF EXISTS DATE_DIM;
CREATE TABLE DATE_DIM (
   [DateNum] INT NOT NULL PRIMARY KEY,
   [Date] DATE NOT NULL,
   [YearMonthNum] INT NOT NULL,
   [Calendar_Quarter] VARCHAR(8) NOT NULL,
   [MonthNum] TINYINT NOT NULL,
   [MonthName] VARCHAR(10) NOT NULL,
   [MonthShortName] CHAR(3) NOT NULL,
   [WeekNum] TINYINT NOT NULL,
   [DayNumOfYear] SMALLINT NOT NULL,
   [DayNumOfMonth] TINYINT NOT NULL,
   [DayNumOfWeek] TINYINT NOT NULL,
   [DayName] VARCHAR(10) NOT NULL,
   [DayShortName] CHAR(3) NOT NULL,
   [Quarter] TINYINT NOT NULL,
   [YearQuarterNum] INT NOT NULL,
   [DayNumOfQuarter] TINYINT NOT NULL,
)
```

Figure 3: CREATE_TABLE.sql

Step 4. Set the location of data source CSV-file

```
DimDates2020.csv
```

Figure 4: Data_File_config.cfg

Now you are ready to launch!

Using

Step 1. Run the Database_Tools.py script to create the DATE_DIM table.

Notice. If the table exists it will be dropped. Attention! All data will be deleted!

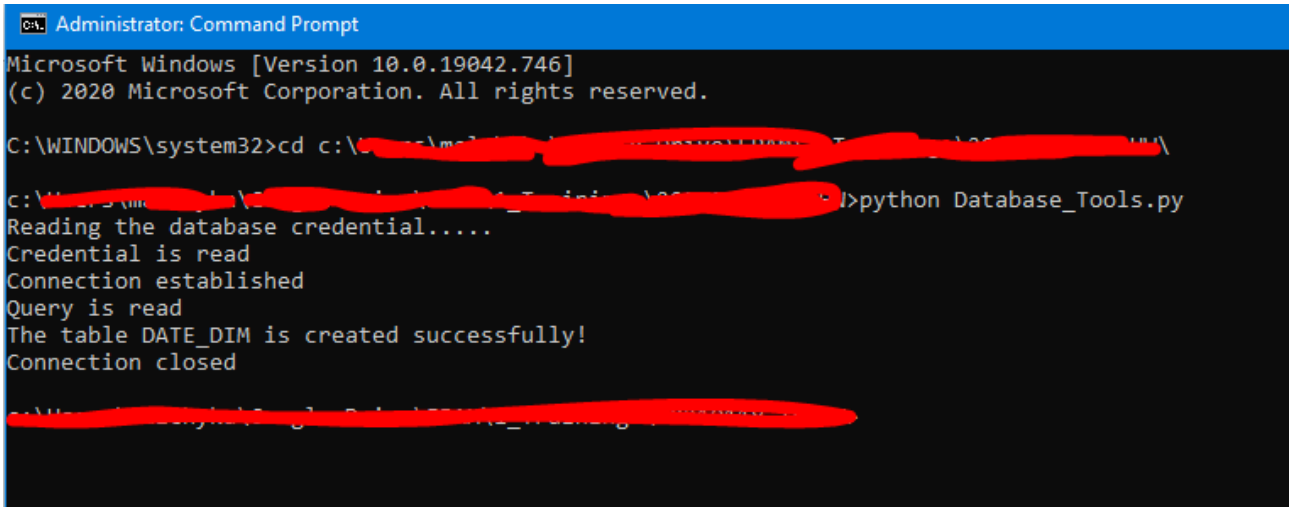

Figure 5: Run Database_Tools.py

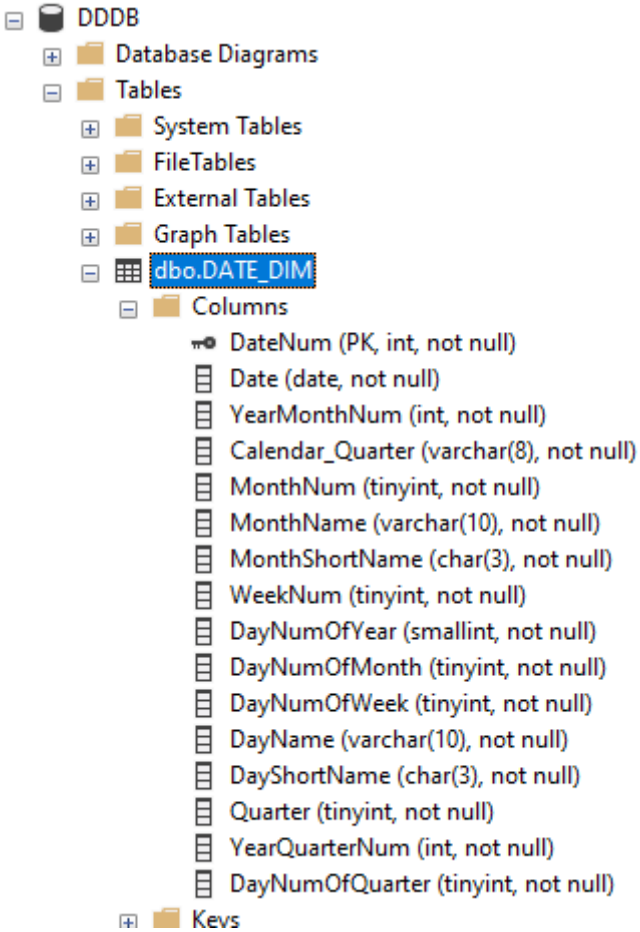

Figure 6: DATA_DIM table

Step 2. Run the ETL_Tools.py script to deploy data from the source to the table.

Notice. You should be sure the CSV file has the same structure as the table.

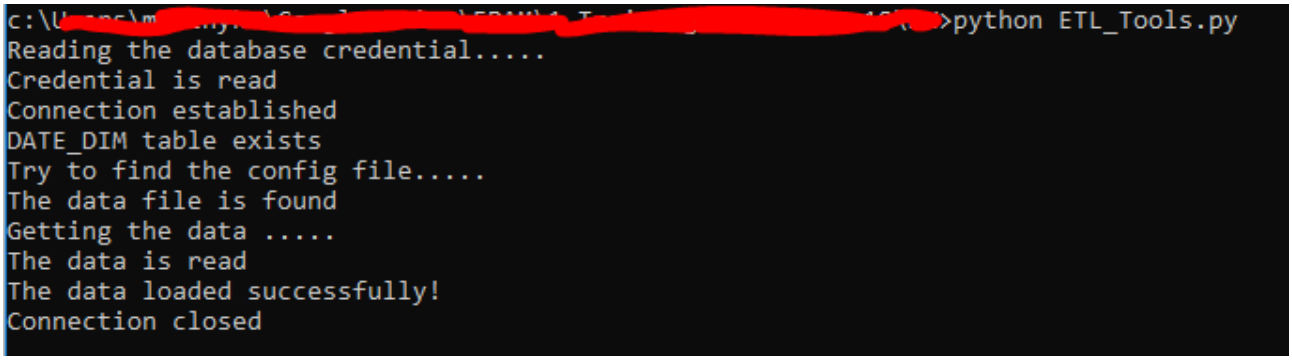

Figure 7: Run ETL_Tools.py