**How to Bulk Load Data with ODBC and Python**

I think Hello World of Data Engineering to make an one-to-one copy of a table from the source to the target database by bulk-loading data. The fastest way to achieve this is exporting a table into a CSV file from the source database and importing a CSV file to a table in the target database. With any database, importing data from a flat file is faster than using insert or update statements.

To connect ODBC data source with Python, you first need to install the [pyodbc](http://mkleehammer.github.io/pyodbc/) module. Obviously, you need to install and configure ODBC for the database you are trying to connect.

Let’s load the required modules for this exercise.

[cc lang="python" tab\_size="4" lines="-1"]

[/cc]

**Exporting table to CSV**

The function below takes a select query, file path for exported file and connection details. The best practice is to turn on autocommit. Some ODBC will give you an error if this parameter is not there.

[cc lang="python" tab\_size="4" lines="-1"]

[/cc]

The execution example is exporting the city table to a csv file from MySQL. ODBC is set up with MySQL\_ODBC as DSN.

[cc lang="python" tab\_size="4" lines="-1"]

[/cc]

**Load Table from CSV**

The function below takes a csv upload query and connection details to import CSV to a table. Autocommit should be turned on. The local\_infile parameter helps MySQL’s LOAD DATA INFILE commands. It may not relevant for other databases. If the parameter is not relevant in the connection for the specific database, it will ignore it. So, you can keep the local\_infile parameter for other databases.

[cc lang="python" tab\_size="4" lines="-1"]

[/cc]

Let’s load the data exported with the first function into both MySQL and PostgreSQL databases. Each database has SQL syntax for this and you need to pass the statement to the function. MySQL uses the [LOAD DATA INFILE](https://dev.mysql.com/doc/refman/5.7/en/load-data.html) command while Postgres uses the [copy command](https://www.postgresql.org/docs/10/static/sql-copy.html).

[cc lang="python" tab\_size="4" lines="-1"]

[/cc]