Accessing MySQL from Python

DSCI 551

Wensheng Wu

Installing packages

- `pip3 install --upgrade 'sqlalchemy<2.0'`
- `pip3 install pymysql`

Using dataframe in Pandas

```
import sqlalchemy
import pymysql
pymysql.install_as_MySQLdb()
import pandas as pd

my_conn =
sqlalchemy.create_engine("mysql+mysqldb://root:Dsci-
551@localhost/dsci551")
beers = pd.read_sql('select * from Beers', my_conn)
beers
```

	name	manf
0	Bud	Anheuser-Busch
1	Bud Lite	Anheuser-Busch
2	Budweiser	Heineken
3	Michelob	Anheuser-Busch
4	Summerbrew	Pete's

1. `DATABASE_URI = 'mysql+pymysql://root:Dsci-551@localhost/CINEMA'`:
   - This line defines a database URI (Uniform Resource Identifier) string. It specifies the connection details required to connect to the MySQL database named "CINEMA" running on the local machine.
   - The URI format `'mysql+pymysql://username:password@host/database_name'` is used, where:
     - `username` is the username used to connect to the database (in this case, 'root').
     - `password` is the password for the specified username (in this case, 'Dsci-551').
     - `host` specifies the hostname of the MySQL server (in this case, 'localhost' means the database is running on the same machine).
     - `database_name` is the name of the MySQL database (in this case, 'CINEMA').

Therefore in HW3
we actually have to change the “database name” from dsci-551 to CINEMA

Using dataframe in Pandas

```
>>> beers.to_sql('mybeers', my_conn, index=False)
>>> # this will create a table called mybeers with content
in beers
```