**PostgreSQL and Python Lab**

import psycopg2

# Connect to an existing database

conn = psycopg2.connect(host="whatever", dbname="test", user="postgres", password="")

# Open a cursor to perform database operations

cur = conn.cursor()

# Execute a command: this creates a new table

cur.execute("CREATE TABLE test (id serial PRIMARY KEY, num integer, data varchar);")

# Pass data to fill a query placeholders and let psycopg2 perform

# the correct conversion (no more SQL injections!)

cur.execute("INSERT INTO test (num, data) VALUES (%s, %s)", (100, "abc'def"))

# Query the database and obtain data as Python objects

cur.execute("SELECT \* FROM test;")

cur.fetchone() # returns (1, 100, "abc'def")

# Make the changes to the database persistent

conn.commit()

# Close communication with the database

cur.close()

conn.close()

1. Clone the repl link from the class webpage to your own repl.   
   1. Click the green "run" button at the top to get started (we won't need to do this again).
   2. Once you see "no such file main.py" you're good to go!
   3. Click "Shell"
   4. You can now run python programs manually by typing "python *filename*.py"
2. Edit schema.sql
   1. Run create-tables.py (you will need to use your own username/password)
3. Edit test-users.py
   1. Examine dump\_users()
   2. Examine add\_user()
   3. Call add\_user a few times from main()
   4. Use print/input statements to let the user add users to the database.
   5. Examine add\_users\_from\_csv() and call from main().
4. Duplicate the functionality of test-users in test-classes.py, through importing from CSV.
5. Add a new table to let students enroll in classes. Edit schema.sql to do this. Rerun create-tables.py to make the new table. (You will have to re-run test-users.py and test-classes.py to re-add the content to the tables.)
6. Write a new Python program to let the user (from the keyboard) choose a user and a class and add the user into the class. Display a list of users and a list of classes and let the user pick which class they want. Make sure they can't add a class at the same time as a class they already have. If time, make it so the list of classes displayed won't show classes that conflict in time with their current schedule.