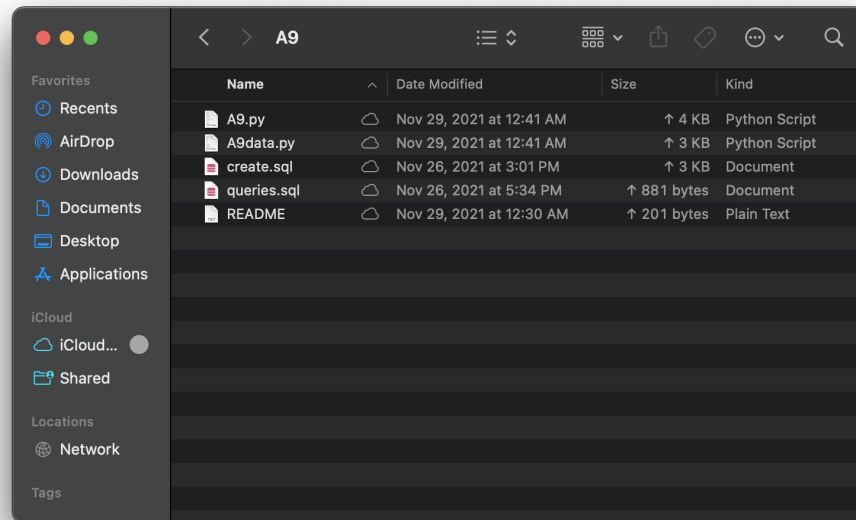


Installation instructions

1. Follow steps 1, 3, and 4 from the [Oracle Knowledge Base](#) to install all dependencies needed to run the database UI. Choose your operating system from the top tabs.
 - a. For step 1, follow the instructions for **“if you already have a database but it is on a remote computer”** in order to install Oracle Instant Client.
2. Extract A9.zip into the desired location. The zip should contain the following files:

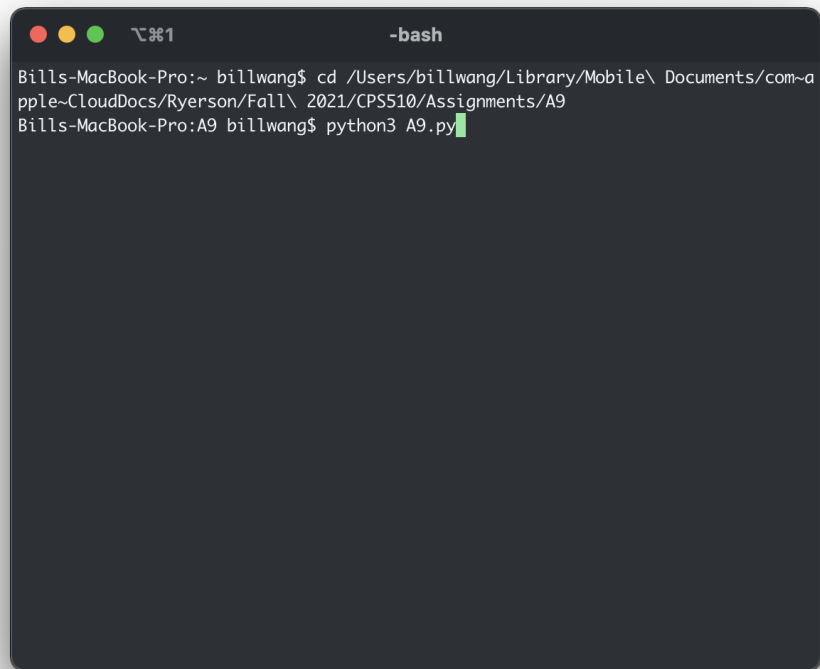


How to use

The following instructions are based on use on macOS.

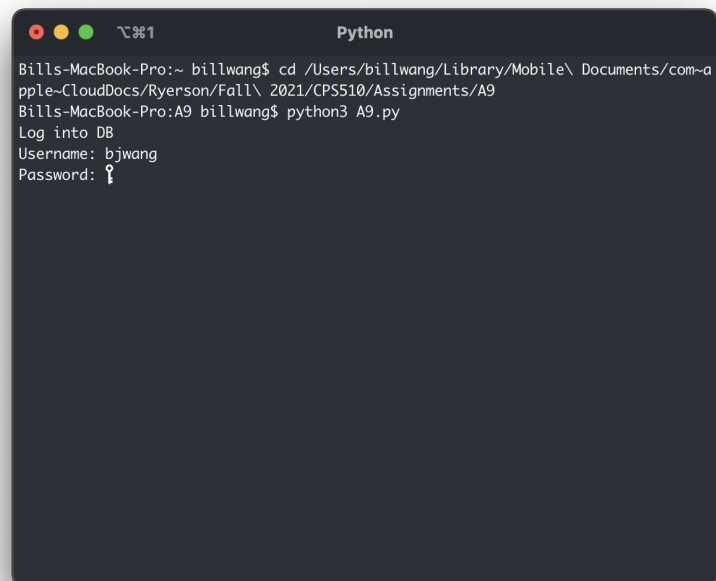
Before use, please ensure that you have access and are connected to the Ryerson DCS VPN network. More information on how to connect to the VPN can be found on the CPS510 course shell on D2L.

Once you have followed the installation instructions, ensure that you are situated in the directory where you have extracted the application files. Then, run the database application in a terminal window by typing in `python A9.py`. Depending on your operating system or Python installation, you may need to use `python3` instead of `python`.



```
Bills-MacBook-Pro:~ billwang$ cd /Users/billwang/Library/Mobile Documents/com~apple~CloudDocs/Ryerson/Fall\ 2021/CPS510/Assignments/A9
Bills-MacBook-Pro:A9 billwang$ python3 A9.py
```

On start, the application will prompt for your CS username and password in order to connect to the Ryerson Oracle 11g database:



```
Bills-MacBook-Pro:~ billwang$ cd /Users/billwang/Library/Mobile Documents/com~apple~CloudDocs/Ryerson/Fall\ 2021/CPS510/Assignments/A9
Bills-MacBook-Pro:A9 billwang$ python3 A9.py
Log into DB
Username: bjwang
Password: *
```

Once connected, the application will give options to interact with the database:

```
Python
=====
| Oracle All Inclusive Tool |
| Main Menu - Select Desired Operation(s): |
=====
1) Drop Tables
2) Create Tables
3) Populate Tables
4) Query Tables

E) End/Exit
Choose: █
```

Each option can be selected by typing the corresponding letter and pressing Enter. The options are described as follows:

1) Drop tables

Drops all tables from the current database.

```
Python
=====
| Oracle All Inclusive Tool |
| Main Menu - Select Desired Operation(s): |
=====
1) Drop Tables
2) Create Tables
3) Populate Tables
4) Query Tables

E) End/Exit
Choose: 1
Dropped 1 table
Dropped 1 table
Dropped 1 table
Dropped 1 table
Dropped 1 table
Dropped 1 table
Dropped 1 table
Dropped 1 table
Dropped 1 table
Dropped 1 table
Dropped 1 table
Dropped 1 table
Dropped 1 table
Dropped 1 table
Dropped 1 table
=====
| Oracle All Inclusive Tool |
```

2) Create tables

Creates all tables necessary for the functioning of the university online student registration system.

```
Python
=====
| Oracle All Inclusive Tool |
| Main Menu - Select Desired Operation(s): |
=====
1) Drop Tables
2) Create Tables
3) Populate Tables
4) Query Tables

E) End/Exit
Choose: 2
Created 1 table
Created 1 table
Created 1 table
Created 1 table
Created 1 table
Created 1 table
Created 1 table
Created 1 table
Created 1 table
Created 1 table
Created 1 table
Created 1 table
Created 1 table
Created 1 table
Created 1 table
Created 1 table
=====
| Oracle All Inclusive Tool |
```

3) Populate tables

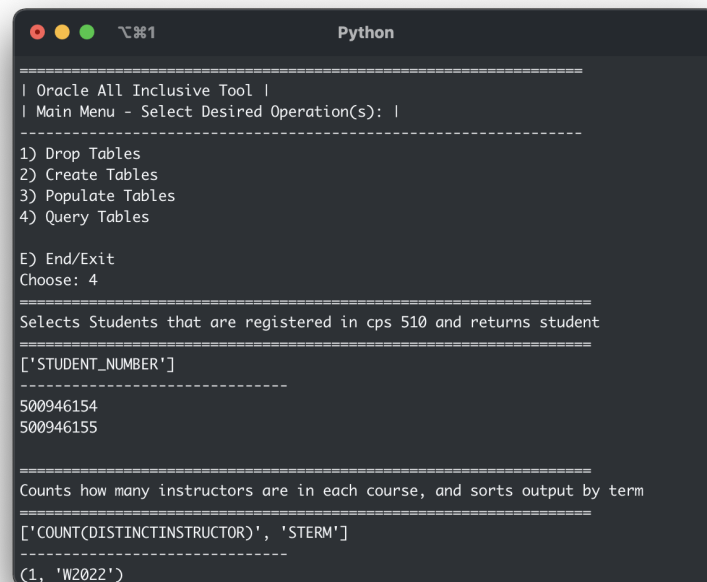
Populates all tables generated from 2) with dummy data.

```
Python
=====
| Oracle All Inclusive Tool |
| Main Menu - Select Desired Operation(s): |
=====
1) Drop Tables
2) Create Tables
3) Populate Tables
4) Query Tables

E) End/Exit
Choose: 3
5 rows inserted into Student table
2 rows inserted into Transactions table
3 rows inserted into Addresses table
5 rows inserted into Course table
15 rows inserted into Section table
15 rows inserted into sSection table
2 rows inserted into Department table
4 rows inserted into Registered table
3 rows inserted into Undergrad table
1 rows inserted into Documents table
2 rows inserted into Graduate table
2 rows inserted into sDegree table
=====
| Oracle All Inclusive Tool |
```

4) Query tables

Runs queries with specific applications using the tables with data generated from 3).



```
Python
=====
| Oracle All Inclusive Tool |
| Main Menu - Select Desired Operation(s): |
=====
1) Drop Tables
2) Create Tables
3) Populate Tables
4) Query Tables

E) End/Exit
Choose: 4

=====
Selects Students that are registered in cps 510 and returns student
=====
['STUDENT_NUMBER']
-----
500946154
500946155

=====
Counts how many instructors are in each course, and sorts output by term
=====
['COUNT(DISTINCTINSTRUCTOR)', 'TERM']
-----
(1, 'W2022')
```

e) End/exit

Ends the program and database connection.

Queries in relational algebra

1) List student numbers of students who are enrolled in CPS 510

$$\Pi_{\text{Student_number}} (\sigma_{\text{Course_number} = '510'} (\text{Student} \bowtie \text{Registered}))$$

2) Count number of instructors in each course, sorted by term

$$\text{result}(\text{Sterm}, \text{Instructor}) \leftarrow \text{sterm } F_{\text{COUNT Instructor}}(\text{sSection})$$

3) Count students with each citizenship and list the citizenships with more than 3 students

$$\text{cStudents}(\text{Citizenship}, \text{Student_count}) \leftarrow \text{citizenship } F_{\text{COUNT Student_number}}(\text{Student})$$
$$\text{result} \leftarrow \sigma_{\text{Student_count} > 2}(\text{cStudents})$$

4) List of undergrad or graduate students in Ryerson currently studying Computer Science

$$\text{CSgrad} \leftarrow \Pi_{\text{Student_number}} (\sigma_{\text{GDegree} = 'COMPUTER SCIENCE'} (\text{Graduate}))$$

```
CSundergrad  $\leftarrow$   $\pi_{\text{Student\_number}}(\sigma_{\text{Degree} = \text{'COMPUTER SCIENCE'}}(\text{Undergraduate}))$   
result  $\leftarrow$  CSgrad  $\cup$  CSundergrad
```