BBM473 Database Laboratory (Fall 2018)

Exercise 0: Introduction

Let us warm up!

Note: DO NOT PANIC

- Don't worry if you get (a) a big red-highlighted warning or (b) a note that the extension has already been loaded! As long as your SQL commands work, it's loaded properly!
- If you don't understand all the SQL queries... very soon, you will!

In [ ]:
```
%load_ext sql
```

Next, we'll load an SQLite database stored as a file as follows:

NOTE: We load a file below (here, "dataset_1.db", which must be in the same directory as the notebook. You'll use this file throughout the next few lectures. Make sure to download it from the webpage!!

In [ ]:
```
%sql sqlite:///dataset_1.db
```

Lets look at one table from the US National Oceanic and Atmospheric Administration (NOAA) Rainfall dataset- `precipitation_full`- having the following schema:

- `state_code`
- `station_id`
- `year`
- `month`
- `day`
- `hour`
- `precipitation`
- `flag_1`
- `flag_2`

Each tuple in this table describes one hour of rainfall (`precipitation`- in hundredths of an inch) at one station (`station_id`) in one state (`state_code`). Note that tuples with `hour=25` record the total rainfall for that day, and that we can ignore the values of attributes `flag_1` and `flag_2` for now.

Let's take a look at some sample rows of the table via SQL queries.

Note that you can also load an empty in-memory database using:

`%sql sqlite://`

Now let's try out some queries! (Don't worry, if you're new to SQL, we'll go over all the syntax in more depth next lecture)

`%sql` is used for single line SQL commands:

In [ ]:
```
%%sql
SELECT * FROM precipitation_full  LIMIT 3;
```

In [ ]:
```
%%sql
SELECT COUNT(*) FROM precipitation_full;
```

And `%%sql` **is used for multi-line SQL commands:**

In [ ]:
```
%%sql
SELECT SUM(p.precipitation)
FROM precipitation_full p, states s
WHERE p.state_code = s.code AND s.abbrev = 'CA';
```

We can also work with the output of the queries we issue:

In [ ]:
```
result = %sql SELECT * FROM states;
```

In [ ]:
```
result.keys
```

In [ ]:
```
# Alternative python code
for k in result.keys:
    print(k)
```

Note that the first row is the header row with column names!

In [ ]:
```
result[0]
```

In [ ]:
```
result[1]
```

In [ ]:
```
result[1].name
```

Now feel free to have fun with the dataset- we'll see more of it later though!