

## Lab #5 — Introductions to SQL and Rust

Michael McThrow

CS 152 — Section 05

San José State University

Fall 2020

### SQL (50 points — 50% of lab grade)

NOTE: We will be using SQLite version 3 in this lab, which is available at [sqlite.org](http://sqlite.org). If you are using macOS, then the command `sqlite3` should be available through the terminal. All of your queries need to be written in such a way that conforms to what SQLite 3 expects.

Suppose our database, which is maintained by an e-commerce site, has the following tables with the following schemas:

```
create table customers(  
    c_id integer,  
    first_name varchar(30),  
    last_name varchar(30),  
    age integer,  
    street_address varchar(50),  
    city varchar(30),  
    state varchar(30),  
    zip integer  
);
```

```
create table purchases(  
    p_id integer,  
    c_id integer,  
    i_id integer,  
    quantity integer,  
    p_date date  
);
```

```
create table items(  
    i_id integer,  
    name varchar(50),  
    price real  
);
```

Please write the following queries in a file called `lab5.sql`:

1 (5 points): Retrieve all customers by first and last name who are from California. Retrieve just the first and last names.

2 (5 points): Retrieve all item names that are priced at or above \$10. Retrieve just the item names.

3 (10 points): Retrieve all purchases made on November 27, 2020. Retrieve the item names and the purchase dates.

4 (10 points): Find all customers who made purchases on November 27, 2020. Retrieve just the customers' first and last names.

5 (20 points): Find all customers who purchased more than \$100 of items on November 27, 2020. Retrieve just the customers' first and last names.

For resources regarding learning SQL, here are some helpful ones below:

<https://www.sqlitetutorial.net>

<https://www.w3schools.com/sql/>

I learned SQL from the textbook *A First Course in Database Systems* by Jeff Ullman and Jennifer Widom.

### **Rust (50 points — 50% of lab grade)**

For a great tutorial introduction to Rust, I recommend *Rust by Example*, which can be found at <https://doc.rust-lang.org/stable/rust-by-example/>. For a more complete description of Rust, I recommend the official book *The Rust Programming Language*, which can be found at <https://doc.rust-lang.org/book/>.

Please write the following programs:

1 (10 points): Write a program that generates the first 50 Fibonacci numbers. Place this program in `fib.rs`.

2 (40 points): Write a program that inputs a list of numbers from standard input, sorts them using the quicksort algorithm, and outputs the list to standard output. You must implement quicksort yourself; you cannot use a library implementation. Place this program in `sort.rs`.