Data Storage and Retrieval

Fall 2017

Assignment 5

September 28, 2016

Show all your work!

This is a pair assignment.

Work together on as much as possible.

1. Normalize the jail database:
   1. Draw an ER diagram with what you think are important entities. You can using the Drawing tool of Google Docs, starting with the ER Diagram Pieces doc.
   2. Make a new database, jail\_norm\_<name>.db (readable by Noah and me), and create the tables for a jail db in 3rd normal form. Use all appropriate integrity constraints.
   3. Populate the new db. Show your work. Here’s how to work with two db’s in one sqlite3 session:
      1. Open jail.db.
      2. Attach your new db, giving it a schema name of jailnew.
      3. Now you can use table foo in the new db, as jailnew.foo.

CREATE TABLE bookings(bookingNumber int PRIMARY KEY, Agency varchar, ABN int, Court varchar, releaseDate timestamp, releaseCode varchar,SOID int, foreign key(SOID) references people(SOID));

CREATE TABLE charges(charge\_id integer primary key autoincrement, charge\_name varchar unique);

CREATE TABLE Addresses(Address\_id integer primary key autoincrement, address varchar unique);

CREATE TABLE people(SOID int primary key, name varchar, DOB timestamp, POB varchar, Race char(1) check (Race in ("A","B","I","U","W","a","b","w")), e char(1) check (e in ("H","N","U","h","n","u")));

CREATE TABLE BookingsCharges(bookingNumber int foriegn key references bo okings(bookingNumber), chargeType varchar, charge\_id foriegn key references char ges(charge\_id), primary key (bookingNumber,charge\_id));

insert into addresses values(address) select distinct address address from jailnew.bookingsB;

insert into bookings select bookingNumber, Agency, ABN, Court, releaseDate, releaseCode, SOID from jailnew.bookingsB;

insert into people (SOID, name, DOB, POB, Race, e) select \* from (select SOID, name, DOB, POB, race, e from jailnew.bookingsB group by SOID);

insert into BookingsCharges(bookingNumber, chargeType, charge\_id) select bookingNumber, chargeType, charge\_id from (select a.bookingNumber, a.chargeType, b.charge\_id from jailnew.**booking\_addl\_charge** a left join charges b on a.charge = b.charge\_name);

insert into BookingsCharges(bookingNumber, chargeType, charge\_id) select bookingNumber, chargeType, charge\_id from (select a.bookingNumber, a.chargeType, b.charge\_id from jailnew.**bookingsB** a left join charges b on a.charge = b.charge\_name);

1. Put fanfiction.stories into 2nd normal form (which implies 1st normal form). To keep url as primary key you’ll need to create a new table. starringchars isn’t the only problem. You’ll probably want to read the Postgres docs on string functions.
2. Seven Databases, Chapter 2, Days 1 and 2 homework.
   1. Section 2.2, Find problem 3.

Match Full is a restraint on foreign keys consisting of more than one column, establishing that either both columns must have non-null values, or both must have null values.

* 1. Section 2.2, Do problems 1, 3.
     1. 1. select relname from pg\_class where relname in ('countries','cities','venues','events');
     2. 2. select c.country\_name from venues v join countries c on v.country\_code=c.country\_code join events e on v.venue\_id = e.venue\_id;
     3. 3. alter table venues add active boolean default TRUE;
  2. Section 2.3, Do problem 2.

SELECT \* FROM crosstab( 'SELECT extract(year from starts) as year, extract(month from starts) as month, count(\*) FROM events GROUP BY year, month', 'SELECT \* FROM generate\_series(1,12)' ) AS ( year int, jan int, feb int, mar int, apr int, may int, jun int, jul int, aug int, sep int, oct int, nov int, dec int ) ORDER BY YEAR;