Further Study

Once you’ve created the required tables and entered data into them, this exercise is very open to self-exploration. Some things to try:

Creating Views

A curious property of relational databases is that our data is tabular, and when you extract different cross-sections of the data as above, the results are *also* tabular. Even though it’s not actually a table, you can use something called a “view” to pretend the results of a particular query are a table. You can then select and join on this virtual table as if it were a real one.

Try writing a query that joins together students, projects, and grades, and that keeps that around as the view ***report_card_view***.

```
CREATE VIEW report_card_view AS
SELECT students.first_name,
       students.last_name,
       projects.title,
       projects.max_grade,
       grades.grade
FROM students
  JOIN grades ON (students.github = grades.student_github)
  JOIN projects ON (projects.title = grades.project_title);
```

Now, you can use ***report_card_view*** as if it were a regular table:

```
SELECT *
FROM report_card_view;
```

Setting Other Primary Keys

Ideally, each table should have a primary key.

For the ***students*** table, you should be able to use a *natural key* – that is, instead of adding a new field just to become the primary key, you could use an existing field. No two students should share the same GitHub account, so you could mark that field as a primary key so that you can’t add two people with the same GitHub account name (or add the same student twice).

To alter tables in PostgreSQL you can use the ***ALTER TABLE*** command.

> **Note: Other Databases**
>
> Not all databases support the ***ALTER TABLE*** command. For example, if we were using SQLite and we wanted to add a primary key to a table that did not previously have one we would have to first dump the database as we did above, then manually change the SQL in the dump file (***hackbright.sql***) to add a primary key, and finaly remove and restore our database with the changed dump file.

To add a primary key constraint to the ***github*** column in the ***students*** table using ***ALTER TABLE***, type this into your psql console:

```
=# ALTER TABLE students ADD PRIMARY KEY (github);
```

After running this command, use the ***d*** command in ***psql*** to see how the ***students*** table has changed.

If you have time take a look at the PostgreSQL ALTER TABLE <http://www.postgresql.org/docs/9.1/static/sql-altertable.html> docs to see what else you can do with the command.