Lesson 9

* To design a database, we need to think about the data we are working with. Let’s pretend we are creating a database to store customer data and track orders and products.
  + To start, we need to define the data that we will need to represent each of these entities.
  + Customer
    - Id, first\_name, last\_name, city, street, state, zip, phone1, phone2, phone3
  + Products
    - Id, name, description, price
  + Orders
    - Id, customer\_id, date\_ordered, date\_delivered, total
  + Product\_orders
    - Product\_id, order\_id, qty
  + This is a decent start to creating a database schema. However, notice how we have phone1, 2, and 3 in our customer table. Any time we start to have repeating rows, that is a sign that our data is not normalized.
  + Database normalization is the process of restructuring a relational database in order to reduce redundancy and improve data integrity. Or, in other words, we normalize our tables to make sure we aren’t repeating data and we don’t have conflicting data.
  + To normalize the customer table, we should move phone numbers into their own table and create a one to many relationship between the customer and phone tables respectively.
  + Phone
    - Id, customer\_id, phone\_number
  + To model our database we will create something called an Entity Relationship Diagram, or ERD. To do this, we can use draw.io.
  + //show how to create diagram on draw.io
* Now that we’ve seen how to model a database, let’s take a look at actually creating the database. To start this, open your text editor of choice, I will just use notepad for simplicity; we are going to write some SQL that will create our database, but we are going to save it in a script that will run all of our commands at once when we are ready as opposed to running each command, one at a time, in the console.
  + Create database if not exists customers;
  + Drop table if exists phones
    - Drop before customers table because of its dependency
  + Drop table if exists customers;
  + Drop table if exists product\_orders;
    - Drop this table first because of its dependencies on products and orders tables
  + Drop table if exists products;
  + Drop table if exists orders;
  + Now we can start actually defining and creating our tables. We can refer to our ERD from the previous lesson to create our database with the same schema we already designed.
  + CREATE TABLE tablename () each column name and data type separated by a comma
    - You can look up the different datatypes. For this database, however, we will only be using int, varchar, datetime, and decimal
  + //define all tables
    - Not null auto\_increment
    - PRIMARY KEY (id)
    - FOREIGN KEY fk\_columnName(column) REFERENCES parent\_table(column)
  + Let’s test the database by entering some data in each table and then querying it.