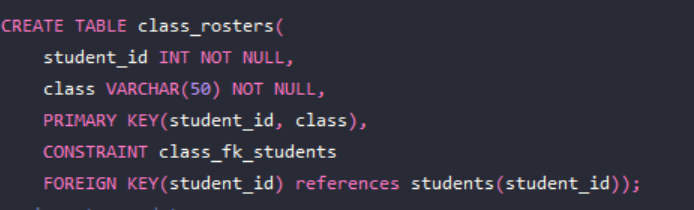
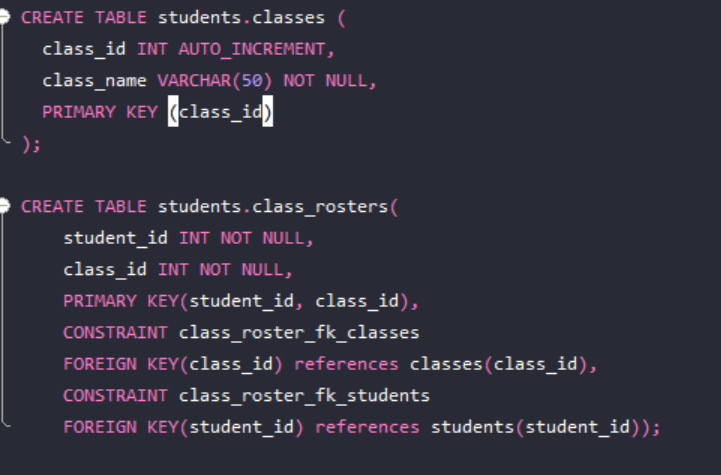


I started with our un-normalized table as presented by the homework assignment. Then to normalize it to the first form I made sure each row was unique by making our id the primary key. Next we have to deal with multivalued columns which include student\_name and classes. We can solve this by breaking student\_name into two columns first and last. The classes multivalued column requires we create a new table something like class roster which could have rows of student\_id and classes designed like



This way the data is no longer multi-valued and the data is stored in it’s own table. Now we have two tables of students and class roster however we have run into an issue when normalizing to the second form. The class column is dependant on the student\_id but the student\_id is only part of the primary key so we’ll have to create a classes table to serve as a parent table and create a foreign key relationship with a class\_id column so that the parent column holds the actual information about the classes while class\_rosters just contains a foreign key of class\_id. After normalizing to the second form our new tables look like



Splitting our table like this will also result in it being in the 3NF because all of our columns are entirely dependent on their respective primary keys.