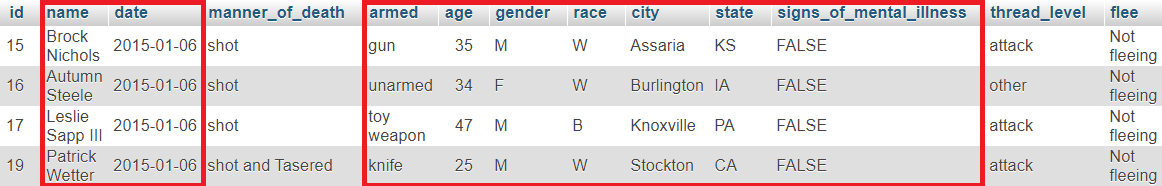
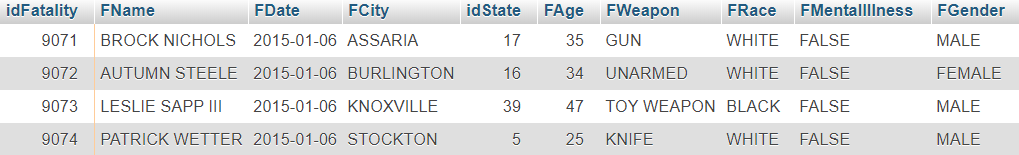
III. DATA PREPARATION

The first step of the data preparation phase is extracting the data from our initial data sources and importing it into 2 MySQL databases, one for each datasets. This is a straightforward process, as we had to write one LOAD DATA statement for each of the .csv files, with the correct field and line termination symbols. For each field, we assigned an appropriate data type, as well as an appropriate primary key for each table.

The main tables(containing information about the police fatalities) from our 2 datasets have a similar structure, with only a slight difference in the columns. We selected only the fields, which are relevant for our analysis and are available in both of the datasets. But despite all of this, the representation of some of the data is a little different. For example the values for race and gender are represented with their full names in one of the datasets, and with only the first letter in the other. Our strategy of dealing with this problem is deciding on one particular format for each field and storing the data in this unified way in the staging area. To avoid lower/upper case conflicts, we saved string values in upper case letters only. There are also some null values, which were automatically replaced by the LOAD DATA statement with an empty string for the VARCHAR type or 0 for numeric types. We stored the empty strings as ‘UNKNOWN’ in the staging area and kept the 0 as our value, which represents the unknown state, for the numeric types. Here is an overview of our solution strategy:





To harmonize the datasets we first stored the data for each one into a temporary table, applying the transformations described in the previous paragraph. However, some duplicate values existed in the bigger dataset, as well as matching values between both of them. The problem is that not all of the duplicates are exact matches. For example 2 records contain the same values for all but one column, in particular for the columns race, weapon and age. One record would contain a concrete value for the particular field and the other one an unknown value. Naturally, we kept the record that contains more information about the fatality. Mostly there we only 2 duplicate rows for one fatality, but there were a few cases with 3 duplicates, which we dealt with manually. After that, we deleted the records that match ones from the smaller dataset. In the end, the temporary tables were merged into one. To make the ids more consistent, we dropped the original id columns and created our own auto increment primary key. Later, while working on the location dimension, we noticed that there was a typo in one of the records with the city Albuquerque and fixed it manually.

Age Dimension

The Age Dimension consists of the tables Age and Age Group. We decided to group the ages into age groups. We used a mapping table for this. The age groups were defined by an interval with an appropriate lower and upper bound(age group ‘UNKNOWN’ for 0 values). The age groups from the mapping table, without the lower and upper bound, were copied into the CDWH Age Group table. For the Age table, all of the distinct age values were taken from the staging area, mapped to their appropriate age group and inserted into the Age table of the CDWH. The age values themselves are used as a primary key to the table.