**Extract, Transform, and Load Project 2 Technical Report**

**Group Members:** Lynne Freilich, Ronnie Phillips, Corban Doran

**Conceptualizing the Development of our Project:**

1. We needed to come up with a case study that took multiple sets of data with their sources and perform a ETL (Extract, Transform, and Load) on those respective sets.
2. For the datasets we would come to use we considered and determined essential the following:

a.) Find datasets with some commonality each containing at least 1000+ rows of data.

b.) Ensure those datasets were at least 2 different types of files (CSV, pdf, API, Spreadsheets, etc.)

c.) Determine which unique identifier we could use as ‘Primary Keys’ or would we use ‘Foreign Keys’ to assemble our data.

1. After some researching and in light of recent news events involving the school shooting in Uvalde, TX we decided to try to remain conscientiously topical and do our case study on gun violence in America.

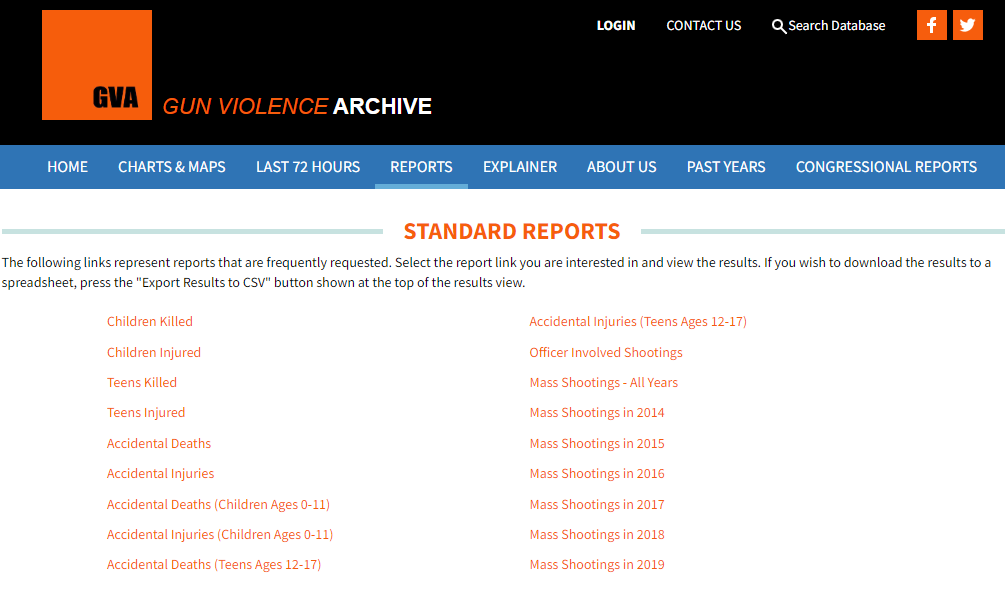
**Project Outline:**

Tragic mass shootings have plagued the United States of America for decades. With the absolute horrid recent school shooting at Robb Elementary School in Uvalde, Texas as well as the hundreds of other shootings that have happened at schools, work, at home, or just on the street we wanted to analyze gun violence in America. We want to find out if gun control laws should in fact be girded more tightly, as we will look at the FBI’s monthly NICs background check data to compare gun checks vs violence, and how these deliberate and malicious shootings stack up against accidental discharges and subsequent injuries or death.

**Extracting the Data:**

First, we read and verified formats of multiple sets of data and ensured each row of data contained useable data. Next, we used selected datasets taken from multiple websites including the FBIs National Instant Criminal Background Check Division which includes data from 1998-2022 for all background checks made in the United States. Additionally, we juxtaposed the FBIs information with multiple CSV documents from “Gun Violence Archive.Org”. From here we extracted an ‘Accidental Death’ dataset as a result from firearms, a ‘Accidental Injuries’ dataset as a result from accidental discharge or misuse of firearms, as well as a ‘Mass Shooting’ dataset that included information from every mass shooting that has happened in the United States since November 14th of 2018.

Since our data came from disparate sources (pdf, CSV, Spreadsheets, HTML tables) we used Pandas and Jupyter Notebook to extract the data and read in our CSV and converted pdf to CSV and put all in a Pandas dataframe to then prepare for the next stage of our project the transformation.

****

*Sources For Extraction***:**

<https://www.fbi.gov/file-repository/nics_firearm_checks_-_month_year_by_state_type.pdf/view>

<https://www.gunviolencearchive.org/reports>

<https://www.gunviolencearchive.org/mass-shooting>

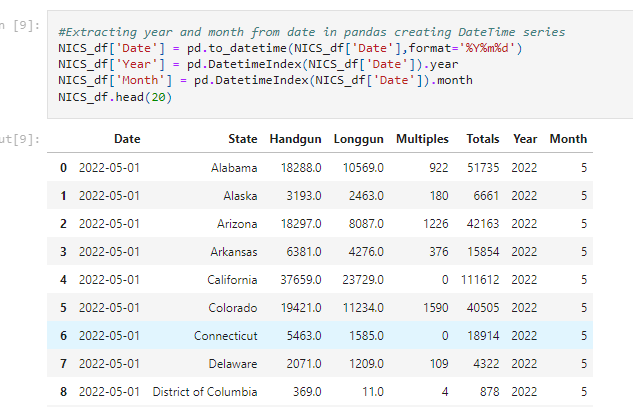
<https://www.gunviolencearchive.org/accidental-deaths>

**Transforming The Data:**

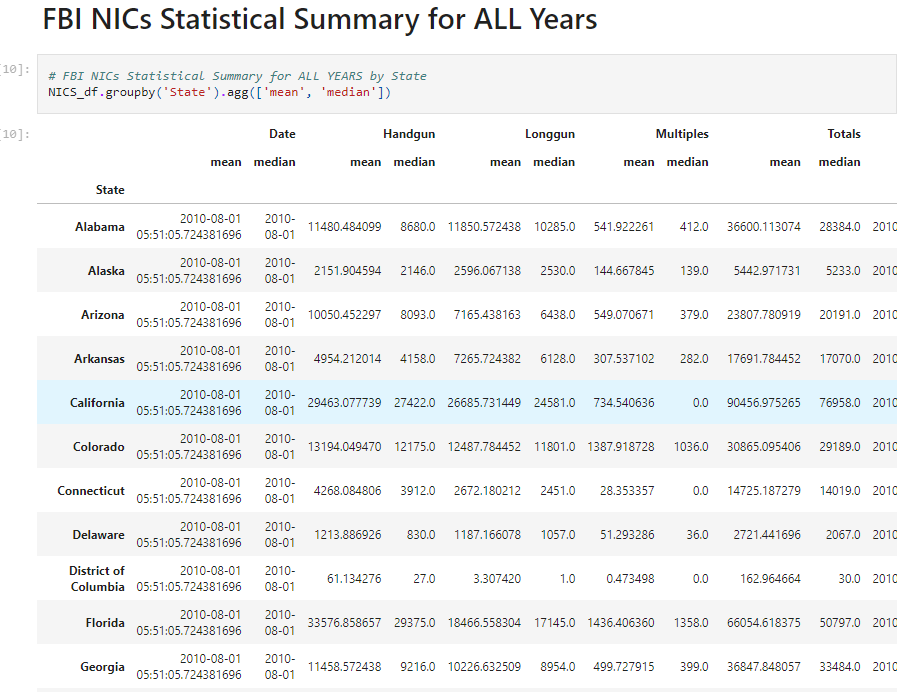
Using Pandas and Python to clean and structure our data we initially transformed the datasets by copying only the columns we wanted to see in each respective CSV. For example, in the FBI NICs CSV file we looked at only (State, Month, Handgun, Long gun, Multiples and Totals).

****

After selecting our preferred columns, we removed any null value columns, renamed all unique columns, selected and performed queries on exclusive years, states, totals, incidents, deaths, etc.… to do identify any duplicates across all 4 data sets and then perform a statistical summary on each. We imported and used datetime series to look and individual Months and Years. We performed data wrangling, cleaning, filtering, and aggregation on all CSV files.

****

We then sorted our data by year and by month so we could visually see which state had the highest number of accidental deaths, mass shootings, accidental injuries, and subsequently FBI NICs background checks ran during that time period. Lastly, we set the index to either a previously created ‘Primary Key’ or reset the index if no Primary Key was available.

****

**Loading The Data:**

After we pulled in the CSV files and loaded them into the data frames, we did an initial connection to the Postgres database using PG admin to store our original clean data sets. We used the quick database website to create the initial table schema that got loaded into the Postgres database that generated the first set of tables. After running the queries and created the new tables with only the relevant information we reconnected to the database and generated additional tables for the data frames.

****