* **Intro**
  + **Project Title:** America’s Next Top Baby Name
  + **Group Members**
    - Jahin Fayyaz
    - Seth Pruitt
    - Ceci Tipton
    - Amanda Fitzpatrick
    - Greta Pfundt
    - Curt Gifford
    - Xing Shen
  + **Summary**
    - Questions/Problem
      * Is there a correlation between the characters from popular TV shows and movies and the use of baby names?
    - Hypothesis
      * There will be a correlation between Movies/TV shows making a cultural impact with the usage of baby names.
  + **Technologies and Resource Contributions**
    - All team members worked on everything collectively.
    - Cross team collaboration is our key to success.
* **Data Import/Extract Sources and Method**

**A screenshot of a cell phone

Description automatically generated**

* + **Movies Data Set Resource: The Movies Database API**
    - <https://developers.themoviedb.org/3/movies/get-popular-movies>
    - Ranked movies by their popularity for a particular year.
    - Saved data into a Pandas DataFrame and organized by year and popularity.
    - Dropped unnecessary data from table:
      * ‘Adult’, ‘backdrop\_path’, ‘genre\_ids’, ‘overview’, ‘poster\_path’, ‘video’
    - **To get the characters in each movie for analysis**
      * Only retrieved data between 2012 – 2017.
      * Data retrieval methods:
        + Initial calls included top 20 movies, limited it to top 10 movies as they would most likely have the most cultural impact.
        + Called OMDB API to attain unique ID for each movie.

<http://www.omdbapi.com/>

* + - * + Used unique ID to build useable URLs for the IMDB website.
        + Utilized BeautifulSoup and unique IDs to web scrape character lists for each movie.

The ‘tconst’ (IMDB unique identifier) allowed us to access the IMDB page/URL for any movie we wanted to access.

* + - * + Movies without character lists were removed from final data set.
  + **TV Shows Data Set Resource: The Movies Database API**
    - <https://developers.themoviedb.org/3/tv/get-popular-tv-shows>
    - Repeated same exact process for the top TV shows.
    - **To get the characters in each TV Shows for analysis:**
      * Only retrieved data between 2012 – 2017.
      * Data retrieval methods
        + Initial calls included top 20 TV shows, limited it to top 10 TV shows as they would most likely have the most cultural impact
        + OMDB API to attain unique ID for each TV show

<http://www.omdbapi.com/>

* + - * + Used unique ID to build useable URLs for the IMDB website
        + Utilized BeautifulSoup and unique IDs to web scrape character lists for each TV shows.

The ‘tconst’ (IMDB unique identifier) allowed us to access the IMDB page/URL for any TV show we wanted to access.

* + - * + TV shows without character lists were removed from final data set.
  + **Baby Names Resource: Data World CSV**
    - <https://data.world/chhs/4a8cb74f-c4fa-458a-8ab1-5f2c0b2e22e3>
    - Found most popular baby names from 2005 – 2017.
* **Data Acquisition**
  + Data is static and does not need updates as we are looking at set a timeframe of 5 years.
* **Data Transform**
  + Movie Database API gave us the most popular movies/TV shows for 30-40 pages due to API call limitations.
    - Used a time delay to get as many API calls as possible.
  + Movies Data Transformation:
    - Returned 8000 results
    - Grouped results by release date
    - Dropped all movies that weren’t in English
    - Sorted by popularity
    - Only utilized top 10 movies
    - Saved data into a Pandas DataFrame and organized by year and popularity
    - Dropped unnecessary data from table:
      * ‘Adult’, ‘backdrop\_path’, ‘genre\_ids’, ‘overview’, ‘poster\_path’, ‘video’
    - Appended character list from OMDB API call/BeautifulSoup web scraping to pandas data frame.
  + TV Shows Data Transformation:
    - Returned 9000 results
    - Grouped by first air date
    - Dropped all shows that weren’t in English
    - Sorted by popularity
    - Only utilized top 10 TV shows
    - Saved data into a Pandas DataFrame and organized by year and popularity
    - Dropped unnecessary data from table:
      * ‘backdrop\_path’, ‘genre\_ids’, ‘overview, ‘poster\_path’
    - Appended character list from OMDB API call/BeautifulSoup web scraping to Pandas data frame.
  + Had to use .explode method for each character name to be on its own row.
  + Dropped unnecessary columns.
  + Renamed columns to make them consistent.
  + Appended both TV shows and Movies DataFrame together.
  + Merged both the TV Shows and Movies DataFrame with popular baby names csv.
* **Visualizations**
  + Used baby names CSV and dropped columns to only have name, count, and year.
  + Used .groupby and .getgroup to group by each unique name and retrieve a group of years and counts that the name appears.
  + Created unique data frames and exported into excel.
  + Uploaded into chart\_studio.plotly to create visualizations.
  + 3D Models:
    - <https://www.youtube.com/watch?v=SHYuB3M5l6M&feature=youtu.be>
  + Interactive Line Graph:
    - <https://cecind123.github.io/class_activities/Names_Plot.html>

A screen shot of a computer

Description automatically generated

* **Data Integrity**
  + Movies/TV shows without character lists were removed.
  + IMDB names did not always include all character names.
    - If one voice actor voices more than one character, not all characters were listed.
  + Some Movies/TV shows didn’t have any characters listed and thus we were unable to pull any data.
  + Movies/TV shows that were not in English were removed.
  + Removed duplicate character names.
  + Took all names in character list and compared them to our list of popular baby names. All names that were not in both lists, were not used going forward.
* **Data Refresh Frequency**
  + Our project contains static data. We are only looking at the years 2012-2017 and thus data will not need refreshing/updating.
* **Data Security**
  + Data anonymity and data security requirements are not needed as we are trying to identify a public cultural trend.
* **Data Loading and Availability**
  + Loaded final data into MongoDB for user access.
    - Each entry included name, count of name, all the movies/tv shows that this name is associated with, and year of movie/tv show.
    - Exporting data into CSV from mongoDB resulted in an unorganized/messy CSV.
* **Discussion**
  + We should look for other places to look for character names, however one would think that IMDB would be one of the prime sources for this data.
* **Final Conclusion**
  + There is no defined relationsip between movie character names and baby name usage as a general trend within our own data set. We found six names that had a change in slope which may be worth investigating further.