Notes: 2/12

- Dataset ideas
    - https://www.kaggle.com/datasets/shiivvvaam/most-watched-movies-and-tv-shows
        - Remove the rank
        - 0 and 1 for type. Tv sho and movie (Ohe hot encode)
        - Genre we can number it up
        - Organized the year in to different group
        - Cleaning up duplicate
        - 181 genre blank - name it unknown or n/a
            - Or delete this data?
        - Join genre + type
        - Chose one of watch time or watchtime in M /
            - Take out the M for watching time in M
        - From a larger website: https://flixpatrol.com/most-watched/.
            - If needed, we can scrape for additional data
- SQL:
    - Doing 2 sql
- NoSQL
    - Firebase: convert from XLSX to JSON (can convert it)
    - **One no sql**
- Tasks:
    - Set up MySQL and PostgreSQL
    - Clean dataset (Riya) (02/ 24)
    - Set up a NoSQL database (Kevin)

Group Proposal:

- [Page 1] **Title**
- **[**Page 1**] Team Details**
- **Team members background and skills [short paragraph for each member]**
    - Riya: undergrad, internships, skills/languages?
        - Undergraduate Data Science
    - Quynh Tran: Background in Business,
        - Focusing on SQL/Python, using backend to solve the problem
        - My name is Quynh Tran and my undergraduate is Business Administration. I have experience in managing and making business decisions based on statistics. I just transferred to Data Science and am doing my master's in person at USC.
    - Kevin Bui:
        - Focusing on Database, SQL, Java, Undergrad in computer science

        - Full-time Remote job as an Application Support Specialist; working in MSSQL, SQL Server, React, Javascript.
        - Have experience in the following languages: Java, Python, SQL, CSS, PHP.
        - My name is Kevin Bui, and I am currently a DEN Student majoring in Computer Science Data Science for my Master of Science. I work full-time as an Application Support Specialist at Center for Creative Leadership. This is a fully remote position, and I work heavily within SQL Server to maintain live data in many different applications and API's. This involves performing data fixes, creating/executing stored procedures, and creating reports for data analysis. I studied at Azusa Pacific University with a Bachelor of Science in Computer Science. I am comfortable with Python, java, and SQL.
- **Project requirements** [Write your understanding of requirements in your own words] **[DO NOT** copy and paste from guidelines]
    - (Kevin and Riya)
    - Kevin:
        - In this project, our task is to create a NLI (natural language interface) to both SQL and a NoSQL database system that will perform CRUD operations by using natural language. This means the NLI can query our database to find what tables exists, show what attributes there are, and provide data from those tables. This should also be able to accept a natural language query from a user and translate that into an SQL/NoSQL query to obtain information. It should also know how to join multiple tables. Our NLI should also be able to create, update, and delete data depending on the user's query. For this project, we must make a proposal, submit progress reports, and demo our project to the rest of the class.
- **Planned Implementation** [Brainstorming][Your planned implementation so far] [Not to be too details whatever idea you have on implementation] (Rachel)
    - SQL
        - Use github to share code
        - MySQL
    - NoSQL:
        - Mongo DB
        - Amazon DynamoDB (integrates with EC2, S3)
        - Cassandra (Apache)
        - Could we also use Firebase and Rest API?
    - LLM API:
        - Mistral API (HuggingFace)
    - Datasets: something in entertainment/media
        - Kaggle may have some large, easy-to-access datasets
        - Yelp/movie reviews
        - Concerts: scrape from LiveNation, TicketMaster

        - https://datasetsearch.research.google.com/ (download CSVs, Excel Sheets)
    - Dataset preparation
        - Scraping and cleaning the dataset
    - NLI interface working (web app)
        - Figure out which type of LLM API we're using
        - Front end for inputting responses and outputting results (need to have a beta NLI interface)
        - 1) Exploration of database
        - First get NLI working on SQL database, then the NoSQL database

Project Breakdown & Task Assignment

1. Project Setup (Everyone)

2. Dataset Preparation

3. SQL Database Implementation

4. NoSQL Database ImplementationNo

5. Natural Language Processing (LLM API Integration)

6. Backend Development

7. Frontend UI Development

8. Testing and Debugging

9. Documentation & Final Report

Responsible Team Member: (Assign a person)

- Write detailed documentation:
    - How the system works.
    - How SQL and NoSQL queries are handled.
    - Challenges and solutions.
- Prepare the **final project report**.
- Upload **code and documentation** to **Google Drive** before submission.
- **Team responsibilities:** How is the work distributed among the team members. [Eg, frontend, distributed database management, backend etc]
    - Rachel: planning the timeline and project management
        - Reminder of deadlines

        - Writing the reports/presentations
        - Back-end
    - Kevin:
        -
- **Timeline** [Planned timeline in a tabular manner with some milestones and checkpoints] [Just a initial draft you can modify as you go] (Rachel and Riya)
    - Weekly Meetings on every Wed before class time (1-2)
    - Next week (2/12):
        - **Get a dataset solidified (everyone)**
        - Research plan on implementing ChatDB
        - Set up Git
        - Set up all the tools we need and do research how we are going to connect and use them
        - Dataset preparation (Riya)
            - Use pandas for cleaning, export to CSV
            - Find a way to make data structured for SQL and NoSQL during cleaning
            - Convert to JSON for NoSQL
    - Week of 2/17
        - MySQL server set up (Riya and Rachel)
            - NoSQL server (Kevin)
    - Week 2/24
        - Set up the LLM API
        - Also get the Web Server started
        - Brainstorming about the back-end work
    - Week 3/3
        - Back-end
        - Front-end implementation
        - Write up the project report
    - Midterm progress report (3/7)