# HW 1 – Apache Cassandra - Executive Summary

**1. Dataset Description and Selection:**

* **Dataset**: Book-Crossing dataset was selected for its real-world application and large size - over 1.14M rows for the largest table - ideal for testing Cassandra's capabilities.
* **Contents**: Three CSV files:
  + Users: provides data on users such as age, location and keyed by a unique user\_id.
  + Books: includes data on books such as publisher and title, keyed by ISBN number (unique).
  + Ratings: provides data on interactions where a user rated a specific book identified by ISBN.

**2. Database Schema Design**:

* Design was tailored for specific queries like book ratings, author ratings, and user activity. Tables include:
  + **Books**: data for all the books, keyed by ISBN. This allows extraction of raw data about books.
  + **Users**:user data keyed by User-ID. This allows extraction of full data per user.
  + **books\_rated\_by\_user:** for each interaction (book ranking) the relevant user data is recorded as well as book information and the rank given by the user. Keyed by (ISBN, location, user-ID).  
    This allows queries about book preferences to be aggregated by the book info.
  + **users\_by\_rated\_books**: here the data is the same as in books\_rated\_by\_user and contains full information for each interaction. However, this table is keyed by (user-ID, book\_author, ISBN).  
    Enables queries based on aggregation by user.

**3. Key Insights from Data:**

* **Book ratings**:
  + While attempting to find the highest rated books, we discovered that the books with the highest rating often had very few users rating them (e.g. rating of 10 by 1 user).
  + This led us to a more refined query: average rating X number of ratings, AND avg. rating > 6.
  + Using this score, we found the top 10 rated books by the user audience.
* **Rating activity:** (see fig. 1)
  + We found that when a book has a lower rating, he is rated more times.
  + Furthermore, the distribution of ratings is heavily left – skewed.
  + This indicates users are more likely to rate low rated books.
* **Popular Authors:**
  + The author with the most ranking activity is Stephen King, followed by Nora Roberts.
  + To identify the top-rated authors, we ratings of authors who were rated at lest 1000 times.
  + The top-rated authors are J. K. Rowling, Barbara Kingsolver, and Dan Brown.
* **Age Groups:**
* Top Books and Authors: Analysis of books and authors with the highest average ratings.
* Queries Example: 'Find average rating of books by each author'

**4. Challenges and Solutions:**

* For easier readability and interfaces we decided to integrate all code to one Jupyter notebook. CQL commands were given to the cluster using Cassandra Driver, a python-Cassandra API.
* During data ingestion, our original script has loaded one row at the time in a for loop. As we have a very large number of rows in our dataset (1.14M in the largest table) runtime was very long. The solution was to use batch operations during loading and using concurrency when executing the calls to the Cassandra cluster.
* Since Cassandra does not allow JOIN between tables and GROUP BY on each column, we had to design a table for almost each query we had to run. We selected the partition key to have high cardinality while being a variable we would like to aggregate \ GROUP BY in the query.
* We found it difficult to analyze user data by age, since age is a continuous variable (int). We chose to sort the users into bins by age. Due to limitations in Cassandra this was done in memory after the query.

**5. Visualizations:**

* **Fig 1** from left to right: each book represented by the avg. ranking and number of rankers;  
  the ranking distribution of all books; and the avg. ranking vs. the number of rankers for all books.

**A graph with blue bars

Description automatically generated**

From this visualization we conclude that users are more likely to rank lower books, and the higher (= better) a book is, the less user ranking activity it actually has. Also, there seems to be an outlier (single book point) ranked 1 many times; and also, abnormally large amounts of zero rankings.

* A graph with a red line and a graph with text

  Description automatically generated with medium confidence**Fig 2** shows the top-rated authors for authors with at least 1000 ratings.
* df