Report 2

# Repo:

<https://github.com/MicMe23/DatabaseDesign>

# Progress:

## Hash join function:

* A computer screen shot of a program code

  AI-generated content may be incorrect.
* The hash\_join\_inner function Is a handmade hash join function and consists of 2 major parts
  + Build Phase: The build phase first compares the two tables and picks the smaller one to create the hash table. For each row in the chosen table, find the value of the join key for that row. For example, customer\_id could be a hash value for a join key. Then store that key in a hash table that is grouped by that hash key. **It essentially takes a join key and creates a list of rows associated with that key.**
  + Probe Phase: Now we enter the large table being joined on. Read the rows, once the program finds a row with a matching key to the hash table then for every match output a new row that will be the new joined table.
* This can be a long process if the data is skewed or if memory is insufficient to build the hash table. In fact it can be catastrophically slow or just outright break.

## Sort Merge:

## Hybrid selector:

## Data used to test:

* NYC taxi data 2025. Normally joining yellow taxi data on green taxi data
  + <https://www.nyc.gov/site/tlc/about/tlc-trip-record-data.page>

# Results so far / hypothesis: