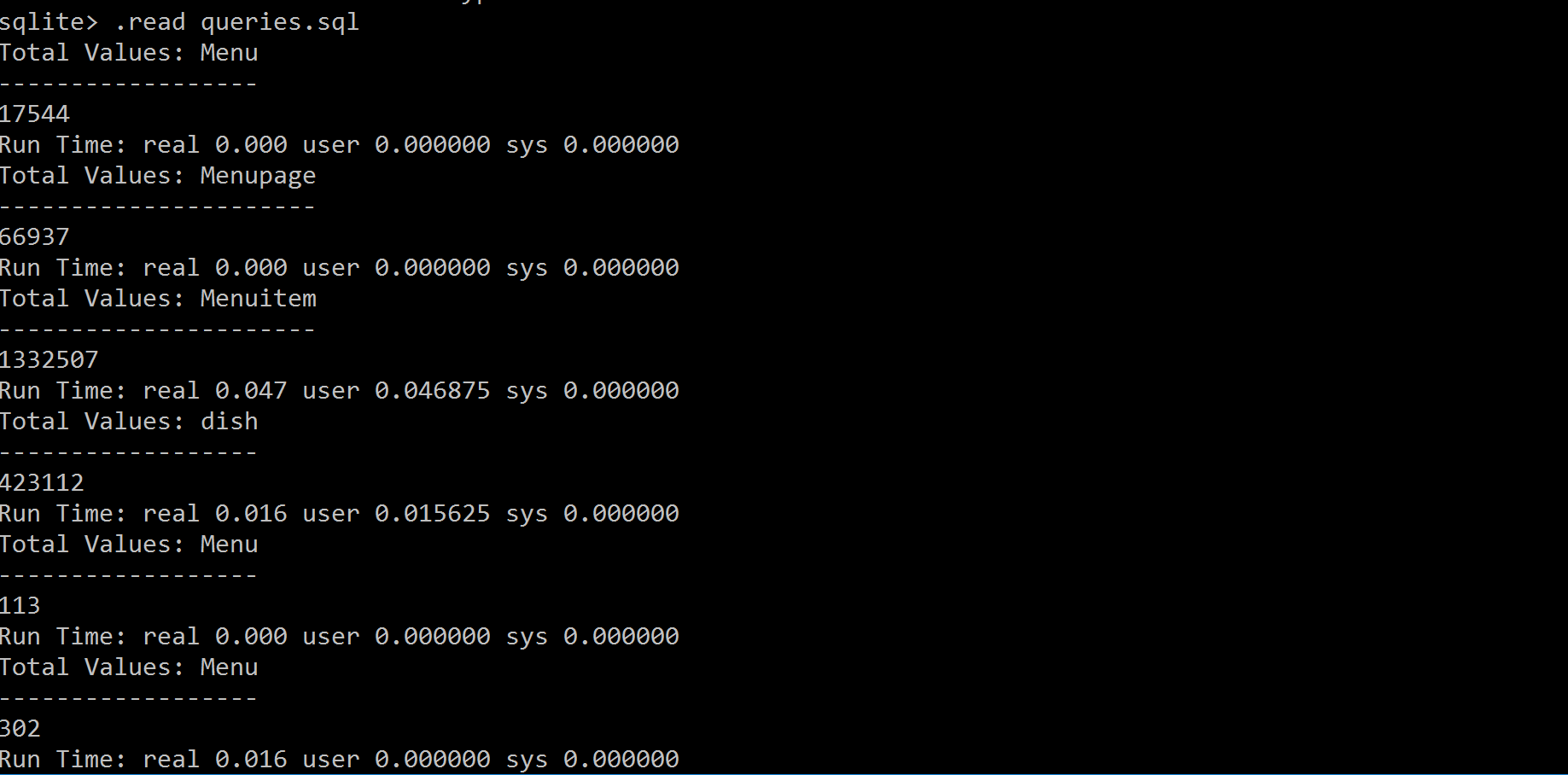
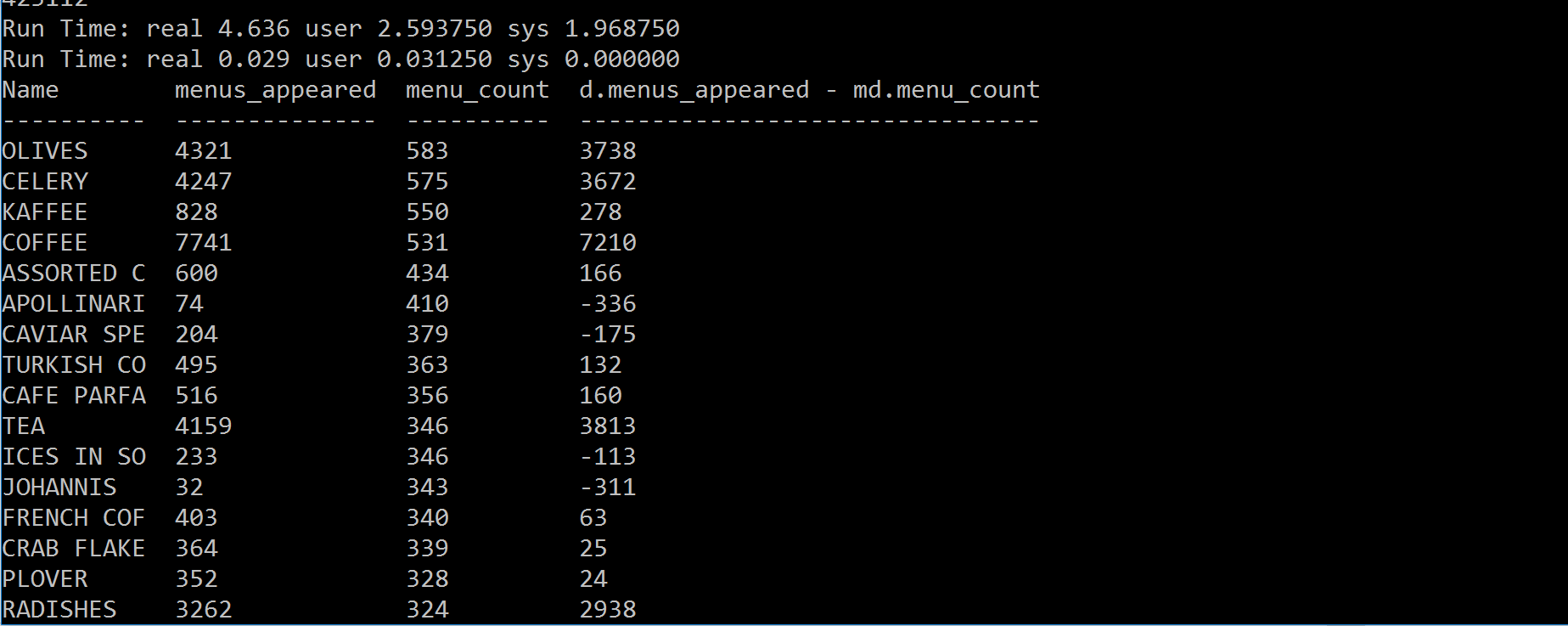
**SQL:**

**Steps Performed:**

1. Created ER Diagram and identified the relationship between different dataset.
2. Identified the Integrity constraints and functional dependency between the data.
3. Created table structure with appropriate constraints identified.
4. Loaded the data into the database.
5. Executed queries for identifying the integrity constraint violations.
6. Included queries with some insight by creating sub-queries and used aggregate functions etc.

There are two sql files “create\_tbl\_load\_data.sql” for creation of the table and loading of the data, and the other one “queries.sql” for performing queries on the database.





**Datalog Provenance:**

We have created the facts by recording the steps performed by us, and also from the Openrefine history. We also created queries for the data provenance. We can identify which operation was performed on which dataset and on which column. These queries can be utilized for comparing the Yesworkflow data provenance as well.

**Below are the snippet of the manual operations and Openrefine history.**

**Manual Operation facts:**

operation(menu,id,trim).

operation(menu,id,ws\_collapse).

operation(menu,id,to\_number).

operation(menu,name,trim).

operation(menu,name,ws\_collapse).

operation(menu,name,replace\_char).

operation(menu,name,to\_uppercase).

operation(menu,name,transform\_col).

operation(menu,name,cluster\_keycollision).

operation(menu,name,cluster\_ngram).

**OpenRefine history facts:**

history\_refine(menu,id,trim).

history\_refine(menu,id,ws\_collapse).

history\_refine(menu,id,to\_number).

history\_refine(menu,name,trim).

history\_refine(menu,name,ws\_collapse).

history\_refine(menu,name,replace\_char).

history\_refine(menu,name,to\_uppercase).

**Queries for the ICV’s**

trim(C) :- operation(D,C,trim).

repl\_char(C):- operation(D,C,replace\_char).

ws\_rem(C):- operation(D,C,ws\_collapse).

to\_num(C):- operation(D,C,to\_number).

to\_upper(C):- operation(D,C,to\_uppercase).

trans\_col(C):-operation(D,C,transform\_col).

clus\_keycol(C):-operation(D,C,cluster\_keycollision).

clus\_ngram(C):-operation(D,C,cluster\_ngram).

upd\_val(C):-operation(D,C,update\_values).

**% Queries for manual operations recorded.**

% Which dataset and colums both clustering were applied.

clus\_ngram\_finger(D,C) :- operation(D,C,\_),clus\_keycol(C), clus\_ngram(C).

% Which dataset and colums both updates and clustering were performed .

clus\_ngram\_upd(D,C) :- operation(D,C,\_),upd\_val(C), clus\_ngram(C).

% Which dataset and colums both updates and transform were performed .

upd\_trans(D,C) :- operation(D,C,\_),trans\_col(C), upd\_val(C).

% Which dataset and colums had trim, uppercase,ws\_rem were performed .

trim\_upd\_ws\_collapse(D,C) :- operation(D,C,\_),trim(C), to\_upper(C),ws\_rem(C).

% Which dataset and colums had trim, uppercase,ws\_rem were performed .

trim\_upd\_ws\_collapse(D,C) :- operation(D,C,\_),trim(C), to\_upper(C),ws\_rem(C).

% Which dataset and colums had trim, uppercase,ws\_collapse and were performed .

trim\_upd\_ws\_collapse\_replace(D,C) :- operation(D,C,\_),trim(C), to\_upper(C),ws\_rem(C),repl\_char(C).

**% RULE Violation Queries**

% which of cols of datset didn't applied the trim operation.

icv\_trim(D,C) :- operation(D,C,\_),not trim(C).

% which of cols of datset didn't applied the whitesapces collapse operation.

icv\_ws\_coll(D,C):- operation(D,C,\_), not ws\_rem(C).

% which of cols of datset didn't applied the to\_number operation.

icv\_to\_num(D,C):- operation(D,C,\_), not to\_num(C).

% which of cols of datset didn't applied the to\_uppercase operation.

icv\_to\_upper(D,C):- operation(D,C,\_), not to\_upper(C).

% which of cols of datset didn't applied the transform operation.

icv\_tran\_col(D,C):- operation(D,C,\_), not trans\_col(C).

% which of cols of datset didn't applied the clustering keycol operation.

icv\_clust\_keycol(D,C):-operation(D,C,\_), not clus\_keycol(C).

% which of cols of datset didn't updated the values manually operation.

icv\_upd\_val(D,C):-operation(D,C,\_),not upd\_val(C).

% which of cols of datset didn't applied the replace of characters operation.

icv\_repl\_char(D,C) :- operation(D,C,\_),not repl\_char(C).

% which of cols of datset didn't applied the clustering ngram operation.

icv\_clust\_ngram(D,C) :- operation(D,C,\_),not clus\_ngram(C).

%% Common operations from OpenRefine history and manual erecorded steps.

common\_operation(D,C,X) :- operation(D,C,X), history\_refine(D,C,Y), X=Y.

%% OPerations additionaly recorded in MANUAL steps.

diff\_man\_refine(Y) :- operation(D,C,Y), not common\_operation(D,C,Y) .

%% OPerations additionaly recorded in refine history steps.

diff\_refine\_man(Y) :- history\_refine(D,C,Y), not common\_operation(D,C,Y) .

