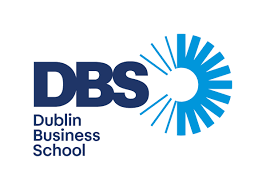
INVIDUAL CONTIBUTION REPORT

Basit halim

10558407



Contents

[Finding/cleaning a Dataset 1](file:///C:\Users\basit%20halim\Downloads\Individual%20report_Ajit%20(1).docx#_Toc325634774)

[Dimensional Modelling for Data warehousing 2](file:///C:\Users\basit%20halim\Downloads\Individual%20report_Ajit%20(1).docx#_Toc325634776)

[Implementation of Source Relational Schema 3](file:///C:\Users\basit%20halim\Downloads\Individual%20report_Ajit%20(1).docx#_Toc325634777)

[Implementaion of Data Warehouse Relational Schema 3](file:///C:\Users\basit%20halim\Downloads\Individual%20report_Ajit%20(1).docx#_Toc325634778)

[Report Generation with Visual Studio(SSRS) 4](file:///C:\Users\basit%20halim\Downloads\Individual%20report_Ajit%20(1).docx#_Toc325634780)

[Tableau 5](file:///C:\Users\basit%20halim\Downloads\Individual%20report_Ajit%20(1).docx#_Toc325634780)

[Neo4j vs SQL 5](file:///C:\Users\basit%20halim\Downloads\Individual%20report_Ajit%20(1).docx#_Toc325634780)

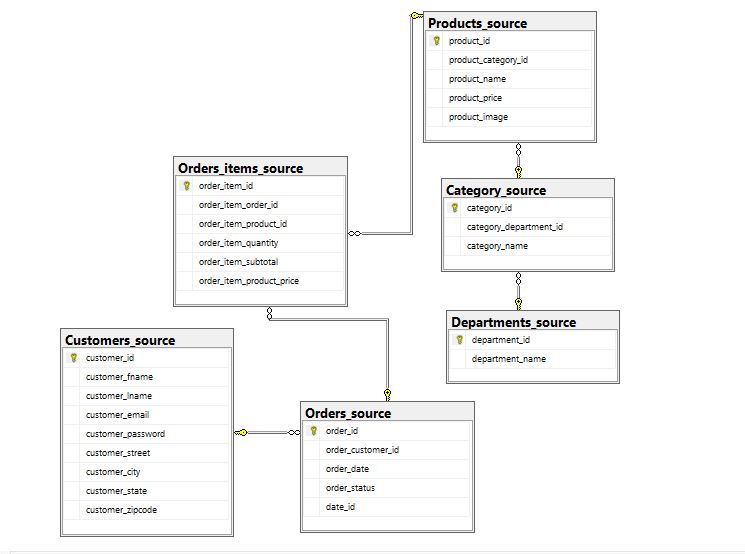
FINDING A DATASET

## Strategic Highlights

In order to pursue our desire to build and ETL project we first needed to find a suitable data set with enough relationships and yet not too many. Stumbling on the internet, we found the Amazon e-retail dataset consisting of various dimensions.

Namely categories, departments, customers, orders, order items and products represent different files of a source system with relations amongst one another which were then used to create a database schema.

Database Diagram of Source schema, diagram 1.1

diagram 1.1: Source Database diagram.

CLEANING THE AMAZON RETAIL DATASET

In the dataset originally, there were a few null values in the attributes that were to become primary keys. The null valued rows were removed resolving the error.

Dimensional Modelling for DATA WAREHOUSE

While performing data modelling, the first error I encountered was that the columns with relational attributes (ID’s) were type casted into different data types. Even after multiple trials, typecasting them into integers did not work. We then type casted them into float data type and it works perfectly.

## Dimension tables and its Attributes:

The next logical step was to realize the important attributes in the obtained tables and understand the relationship between them. After working our way through charades to multiple ideas we decided upon the present relational star schema as it efficiently helps portray the useful insights from data.

Diagram

Description automatically generated

Data warehouse diagram(star schema)

To convey the insight w.r.t. individual customers, areas, for orders, product in accordance with dates, we developed this schema. It obviously was a calculated trial and error process of deciding what to include. The main reason being every attribute gives some information or insight in a particular setting.

IMPLEMENTATION OF SOURCE RELATIONAL SCHEMA

## Creating TABLE:

Great hands-on learning in writing queries was obtained while writing for source tables. I had to repeat the process several times as the data types in source files were all of different data types (float and integer) and the tables had to be altered to make the necessary changes.

## Importing Data:

Used Import wizard to import the data into source tables.

Data was loaded into source tables in such a way that tables with foreign key reference were loaded after the referenced table is loaded, else error occurred. Data import should be done sequentially. First on the tables with no foreign key reference and then on the table with foreign key reference to the table already loaded and so on.

IMPLEMENTATION OF DATAWAREHOUSE RELATIONAL SCHEMA

## CREATING TABLES:

Tables are created as per the Dimension model with the fact referencing all the primary keys of dimension tables.

Customer\_key, Product\_key, Order\_key, calendar\_key are created as a composite key for a non-duplicate data. It is done with the thought that no two given customers can have the same order. Also the same customer can’t have the same product id un the same order and so on. In the end it was realized that a composite key consisting of Customer\_key, Product\_key, Order\_key was the optimal candidate key

GENERATING REPORTS

This part was rather simple yet a bit twisted. Starting with the report wizard, I first selected the data source. Then I built queries adding tables the attributes of which are to be included in the report. Also the reports were generated in Matrix form as they give a 2D representation as opposed to the 1D representation by tables. It was only upon generating some absurd reports I realized that that fact table attribute always goes into details and partition by (generate report w.r.t.) in page.

Graphical user interface

Description automatically generated

TABLEAU

First an idea for representation was dwelled upon. What to show! How to connect with client!

Tableau representation, more visual in its nature that the visual studio SSRS report but similar in nature were intended to help client visualize the figure in forms of charts and graphs. In the beginning, wrong attributes irrelevant to each other were added. But after making my share of irrelevant representation, useful insight were obtained which are discussed in detail in the group report.

NEO4J[NoSQL] and SQL

## Graph Database Neo4j:

Nodes, relations and properties are used to store and represent data. Separate nodes were created for each entity using ‘Ryan School’ dataset.

First, we tried to find a relational dataset but we could find only queries. Those queries were then run in sql server and the data generated was copied in excel files. Post this this data was cleaned for any missing values in identifier columns and removed.

Relations are built using adjoining keys among the datasets. No schema creation or data type considered. Queries and Insertion of data are done with commands called Cypher.

Properties and relations are represented in form of nodes.

Apart from connection issues, no other problem was encountered.

## Relational Database SQL:

Schema is taken into account before starting. Format should be defined prior to storing tables. Data types, relational attributes (primary & foreign keys) should be defined forehand.

Nested queries are used to join multiple tables for data to be inserted into warehouse which makes it more complex to use.