BDA MINI PROJECT: ECOMMERCE

INDEX

1. Aim……2

2. Students and their contribution……3

3. Tool used in the project……4

AIM:

The aim of this project is to identify the unique sellers, unique customers, their delivery location and their payment options using the dataset containing information about customers and their location.

STUDENTS AND THEIR CONTRIBUTION

AKSHAT SHAH 19162121036

- Performed tasks 1-3

ASHIL SHAH 19162121037

- Performed tasks 4-6

DISHWA SHAH 19162121038

- Performed tasks 7-9

TOOL USED IN THE PROJECT: HIVE

- **Hive** was developed by **Facebook** and is designed for **OLAP**.
- Hive is built on top of Apache Hadoop which is an open-source framework used to efficiently store and process large datasets.
- It was created for **non-programmers familiar with SQL** to work with petabytes of data, using an **SQL-like interface called HiveQL** by allowing them to read, write and manage the data.
- Hive uses **batch processing** so that it works quickly across a very large distributed database.
- Hive transforms HiveQL queries into **MapReduce or Tez** jobs that run on YARN.
- It queries the data stored in a distributed storage solution.
- Hive stores its database and table metadata in a **metastore** to abstract and discover data easily.
- Hive is **very easy to distribute and scale** based on your needs.
- Hive **supports multiple file formats** and **supports structured and unstructure**d data.
- Hive operates on a **server side of a cluster**.
- It is mainly **used by data analysts** rather than programmers and researchers.
- Hive **supports all extensions**.
- Hive **supports partitioning** unlike Pig.

Components of Hive:

- **Metastore**: stores system catalog
- **Driver**: manages life cycle of HiveQL query as it moves through HIVE; also manages session handle and session statistics
- **Query compiler**: Compiles HiveQL into a directed acyclic graph of map/reduce tasks
- **Execution engines**: The component executes the tasks in proper dependency order; interacts with Hadoop
- **HiveServer**: provides Thrift interface and JDBC/ODBC for integrating other applications.
- **Client components**: CLI, web interface, jdbc/odbc inteface
- **Extensibility interface** include SerDe, User Defined Functions and User Defined Aggregate Function.

We, as a group, agreed to use Hive to execute our Mini Project. The main reasons for us to do so are:

- The main reason behind this decision is that Hive uses **Hive Query Language**, which is very **similar to SQL** which we are all **quite familiar** to.
- Hive supports structured data which works perfectly for this situation, as our **dataset is structured**.
- Hive structures data into well-understood database concepts such as **tables, rows, columns and partitions.**
- Hive supports **schema for data insertion into the tables**.
- It **supports primitive types** like integers, floats, doubles, strings, arrays, maps, lists and structures.
- It is **fast, familiar scalable and extensible**.
- It is mainly used for **creating reports** and **analyzing the data** which is exactly what we are doing here.