Study Unit 8

Introduction to Business Intelligence

Introduction to Business Intelligence Outline

- Introduction to Business Intelligence
- Database Management System (DBMS)
- Structured Query Language (SQL)
- Data warehouse
- Online Analytical Processing (OLAP)
- Introduction to Data Mining
- Visualization and Dashboard
- Power BI

Study Unit Duration

This Study Unit requires a minimum of 4 hours' formal study time.

You may spend an additional 2-3 hours for revision

Introduction

In the past few decades, there has been a monumental transition in the creation, collection, and use of data. While great strides have been made in data management, the desire to extract business value hidden within the piles (or rather ''mountains'') of these data is the new trend. bits and bytes that today are not only stored within structured data systems. The Systemic processes involved in getting business insights from data is referred to as Business Intelligence.

This study unit will introduce you to Business Intelligence concepts and its importance. This unit is aimed to equip you (learners) with skills on how to mine data from a relational database, how to extract valuable information and create meaningful dashboards that can be used by business owners to make day to day decisions. In addition, you will get to know some of the Open-Source Business Intelligence software, how to quickly set them up and how to use them.

Learning Outcomes of Study Unit 8

Upon completion of this study unit, you should be able

8.1 Describe Business Intelligence, its technologies, benefits, platforms and tools

8.2 Explain the basic principles of DBMS, Data modelling using E-R and Crow's feet notation

8.3 Describe the concept SQL statements and build syntax

8.4 Explain Data mining concept and its different analytic models

8.5 Describe visualization, dashboards and chart methods

8.6 Use and Implement Power BI

8.1 Overview of Business Intelligence

8.1.1 Business Intelligence (BI) Definition

Business Intelligence (BI) refers to technologies, applications and practices for the collection, integration, analysis, and presentation of business information. The purpose of Business Intelligence is to support better business decision making. Basically, BI systems are data-driven Decision Support Systems (DSS). Other simpler definitions of BI are stated in Table 1. The different components of BI are shown by Figure 1. The descriptions of BI technologies are illustrated by Table 1.

Box 1: Definitions of BI

BI is a set of processes, architectures, and technologies that convert raw business data into meaningful information that drives profitable business actions.

It is a suite of software and services to transform data into actionable intelligence and knowledge.

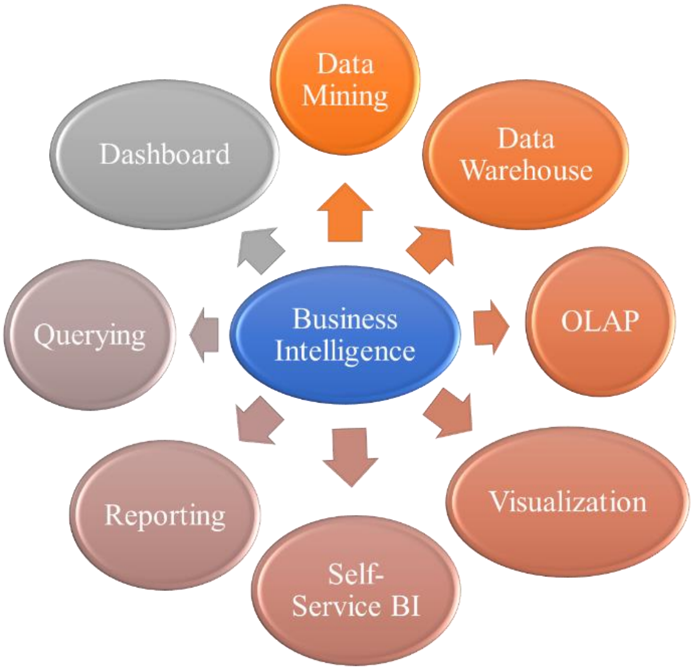


Figure 1: Components of BI

8.1.2 Importance of BI

a. Measurement: creating KPI (Key Performance Indicators) based on historic data

b. Identify and set benchmarks for varied processes.

c. With BI systems organizations can identify market trends and spot business problems that need to be addressed.

d. BI helps on data visualization that enhances the data quality and thereby the quality of decision making.

e. BI systems can be used not just by enterprises but SME (Small and Medium Enterprises)

Table 1 Functions and descriptions of BI technologies

Functions of BI technologies	Description
Data mining	Using databases, statistics and machine learning to uncover trends in large datasets.
Reporting	Sharing data analysis to stakeholders so they can draw conclusions and make decisions.
Performance metrics and benchmarking	Comparing current performance data to historical data to track performance against goals, typically using customized dashboards.
Descriptive analytics	Using preliminary data analysis to find out what happened.
Querying	Asking the data specific questions, BI pulling the answers from the datasets
Statistical analysis	Taking the results from descriptive analytics and further exploring the data using statistics such as how this trend happened and why
Data visualization	Turning data analysis into visual representations such as charts, graphs, and histograms to more easily consume data.
Visual analysis	Exploring data through visual storytelling to communicate insights on the fly and stay in the flow of analysis.

Data preparation	Compiling multiple data sources, identifying the dimensions and measurements, preparing it for data analysis.
OnLine Analytical Processing (OLAP)	OLAP is a powerful technology for data discovery, including capabilities for limitless report viewing, complex analytical calculations, and predictive “what if” scenario (budget, forecast) planning.
Analytics	**It** is the systematic computational analysis of data or statistics. It is used for the discovery, interpretation, and communication of meaningful patterns in data. It also entails applying data patterns towards effective decision making.
Dashboard	It is a type of graphical user interface which often provides at-a-glance views of key performance indicators (KPIs) relevant to a particular objective or business process. In other usage, "dashboard" is another name for "progress report" or "report" and considered a form of data visualization.
Business performance management	Business process management (BPM) is a disciplined approach to identify, design, execute, document, measure, monitor, and control both automated and non-automated business processes to achieve consistent, targeted results aligned with an organization’s strategic goals.

8.1.3 Examples of BI use cases

Example 1:

A hotel owner uses BI analytical applications to gather statistical information regarding average room occupancy and room rate. It helps to find aggregate revenue generated per room.

It also collects statistics on market share and data from customer surveys from each hotel to decides its competitive position in various markets.

By analyzing these trends year by year, month by month and day by day helps management to offer discounts on room rentals.

Example 2:

A bank gives branch managers access to BI applications. It helps branch manager to determine who are the most profitable customers and which customers they should work on.

The use of BI tools frees information technology staff from the task of generating analytical reports for the departments. It also gives department personnel access to a richer data source.

Table 2 People involved with BI

BI Company users	Role description
Data Analyst:	He is a statistician who always needs to drill deep down into data. BI system helps them to get fresh insights to develop unique business strategies.
IT user	The IT user also plays a dominant role in maintaining the BI infrastructure.
Company Head	CEO or CXO can increase the profit of their business by improving operational efficiency in their business.
Business intelligence users	They are found across the organization. They are Casual BI user who uses dashboards to evaluate predefined sets of data and the power user who can work with complex data sets.

8.1.4 Merits of Business Intelligence

a. **Boost productivity:** With a BI program, it is possible for businesses to create reports with a single click thus saves lots of time and resources. It also allows employees to be more productive on their tasks.

b. **To improve visibility:** BI also helps to improve the visibility of these processes and make it possible to identify any areas which need attention.

c. **Fix Accountability:** BI system assigns accountability in the organization as there must be someone who should own accountability and ownership for the organization's performance against its set goals.

d. **It gives a bird's eye view:** BI system also helps organizations as decision makers get an overall bird's eye view through typical BI features like dashboards and scorecards.

e. **It streamlines business processes:** BI takes out all complexity associated with business processes. It also automates analytics by offering predictive analysis, computer modeling, benchmarking and other methodologies.

f. **It allows for easy analytics:** BI software has democratized its usage, allowing even nontechnical or non-analysts users to collect and process data quickly. This also allows putting the power of analytics from the hand's many people.

8.1.5 Demerits of BI System

a. **Cost:** Business intelligence can prove costly for small as well as for medium-sized enterprises. The use of such type of system may be expensive for routine business transactions.

b. **Complexity:** Another drawback of BI is its complexity in implementation of data warehouse. It can be so complex that it can make business techniques rigid to deal with.

c. **Limited use:** Like all improved technologies, BI was first established keeping in consideration the buying competence of rich firms. Therefore, BI system is yet not affordable for many small and medium size companies.

d. **Time Consuming Implementation:** It takes almost one and half year for data warehousing system to be completely implemented. Therefore, it is a time-consuming process.

8.1.6 BI Platforms

Business intelligence platforms enable people import, clean, and analyze data from databases, emails, videos, survey responses, and more. These data analyses provide mobile, desktop and real-time business intelligence so decision makers can act on insights to improve their organization. BI platforms allow users to customize dashboards, create stunning data visualizations, build scorecards, and compare them to key performance indicators (KPIs).

8.1.7 Why Organizations need BI platforms

Business intelligence platforms are more than business analytics software packages. They support your organization’s BI strategy by making it easier to access and analyze your data. Simple analytics platforms import, clean, analyze and forecast data—but business intelligence platforms are more robust and dynamic, and can adapt as you grow your BI strategy.

These are only some of the real-world applications of BI software and what they can do for your organization:

- Visualize supply changes over time
- Forecast sales and profits
- Analyze the results of marketing campaigns
- Automate reporting
- Automate report distribution to stakeholders
- Visualize sales and inventory with near real-time functionality
- Integrate with cloud-based, third party platforms like Amazon Web Services (AWS) and Microsoft Azure
- Embed dashboards into custom solutions for internal or external use

8.1.8 Common features of BI platforms

There are many commercial and open-source BI platforms available in the market, with different user experiences and functionalities. However, each platform will typically offer some combination of the features below.

- Customizable dashboards
- Data visualizations
- Report scheduling, with security specifications
- Data quality management and oversight for IT departments
- Natural language processing (NLP) to discover new insights from data like videos or social media platforms
- Faster data-mining capability

- Performance tracking against KPIs

8.1.9 Common BI tools

Knowi is a new name on the business intelligence radar, only recently starting to show up as a real competitor to the likes of Looker and Tableau. The Knowi platform takes a unique approach to business intelligence by focusing on three big differentiators that have not been a part of traditional intelligence tools. These are Data Virtualization, Search-Based Analytics, and native support for NoSQL data analytics.

- Data Virtualization
- Search-Based Analytics
- Native NoSQL Data Analytics
- White Label Embedded Analytics.

Knowi also supports all the standard features you would expect from a BI tool: data visualizations, interactive dashboards, reports, querying and analytics.

> **Noteworthy because:** Has some cutting-edge features that could have a big impact on the BI world.
>
> **Main downside:** Knowi is a startup, so they may be missing some of the polish of the big players.

Power BI is Microsoft’s big gun in the business intelligence game. If you’re in the Microsoft and **Azure ecosystem**, you are likely already using Power BI in some capacity. Power BI was designed specifically to be an **easy transition for people who love Microsoft Excel.** Here are a few of its features:

- Simple drag-and-drop interface.
- No-code platform. Allows users to do everything in the GUI interface.
- Can be used on desktop or in the cloud.
- Smooth integration into any Microsoft/Windows focused workflows.

- Bundled into Office 365

  Noteworthy because: Azure has been in a growth phase causing a lift to the Microsoft ecosystem as a whole and potentially boosting Power BI along with it.

  Main drawback: Power BI cannot connect to very many data sources because of it's limited library of pre-built connectors. It is also reported to have performance issues on large data sets.

Metabase is, in many ways, the exact opposite of Looker, as it is entirely open-source and fully free. Metabase was designed primarily with simpler use-cases in mind, which is why it's fairly lightweight. Here are a few stand out features:

Tableau is the old guard of business intelligence. If you've been in business analytics or data science for any length of time, you know the name and have probably used it at least once. Tableau is easily the **market leader in BI tools**. Here are a few features it is known for:

- Friendly low-code interface for making visualizations and dashboards.
- Some of the most attractive out-of-the-box visualizations in the industry.
- 150+ pre-built functions for running common data analytics operations.
- User access controls and sharability.
- A strong mobile experience… in case you're into doing data analytics on your phone.
- Generates reports.

Noteworthy because: It's still king of the mountain in 2020.

Main downside: Tableau has been around since 2003 and in most cases is still used as a program that you download to your local computer. The company has done a lot of work to port it into the cloud but it simply doesn't work as well in that space as do a lot of the newer business intelligence tools.

Looker: Looker tends to appeal to a more tech-savvy audience than its main competitor Tableau. It puts a big emphasis on writing SQL queries but also has all the standard features for building dashboards and visualizations. Looker is used across verticals, with users in a wide range of industries. Here are a few standout features for Looker:

- Web-based interactive dashboards.
- Support for mobile devices.
- Can be installed on-premise or in the cloud
- Built-in machine learning/data modeling language called LookML.
- Git integration.
- Strong support for SQL queries.
- Built for data exploration–making it a good tool for data engineers and data scientists alike.

  Noteworthy because: Recently acquired by Google for $2.6 Billion.

  Main drawback: Steep learning curve. Uncertain future after Google Aquisition.

Google Data Studio appeared in 2016 as a new free Google offering. Admittedly, it's fairly limited in the number of data sources it can connect to. Some users have gotten around this issue by porting their data to Google Sheets and then connecting Data Studio to that Google Sheets file. But it seems that, like with PowerBI, Google Data Studio is only a good fit if most of what you do is already in the Google ecosystem. Here are a few features:

- It's completely free.
- Web-based data visualization and interactive dashboards.
- Easily connects to any Google-based data sources like Google Analytics, Google AdWords, YouTube, and Google Search Console.
- Clean minimalist user interface.
- Easy to use.

Noteworthy because: It doesn't often get recognition as a business intelligence tool even though it can work for a lot of use-cases–especially ones having to do with website data analytics.

Main drawback: Severe limitations on data sources

Sisense / Periscope Data The two business intelligence tools merged to provide a BI solution that collects and merges data from multiple sources while managing the collection, integration, and visualization steps itself. Here are a few noteworthy features:

- The combined Sisense / Periscope product provides an end-to-end analytics solution.
- Intelligence platform can be used on-premise or in the cloud.
- The tool has support for mobile devices.
- Integrates with numerous data sources.

Noteworthy because: The SiSense Periscope combined product changes their go-to-market and their value proposition as a business intelligence platform. It will be interesting to see how they position themselves in 2020.

Main drawback: Some users report a lengthy and expensive implementation before they are able to get the product working correctly.

8.2 Databases

A database is a collection of data, typically describing the activities of one or more related organizations. An example a university database with information about the following:
Entities such as students, faculty, product, customer, company, courses, and classrooms.
Relationships between entities, such as students' enrollment in courses, faculty teaching courses, and the use of rooms for courses.

8.2.1 Database Management System

A database management system, or DBMS, is software designed to assist in maintaining and utilizing large collections of data, and the need for such systems, as well as their use, is growing rapidly. Database Management System (DBMS) is software system that enables users to define, create, maintain and control access to the database.

8.2.2 Relational DBMS

A relational database is a digital database based on the relational model of data, as proposed by Egdar Codd in 1970. A software system used to maintain relational databases is a relational database management system (RDBMS). RDBMSs have been a common option for the storage of information in databases used for financial records, manufacturing and logistical information, personnel data, and other applications since the 1980s. Many relational database systems have an option of using the SQL (Structured Query Language) for querying and maintaining the database.

Relational model

This model organizes data into one or more tables or relations of columns and rows, with a unique key identifying each row. Rows are also called records or tuples. Columns are also called attributes. Generally, each table/relation represents one entity type (such as Student, Employee, Customer or Product). The rows represent instances of that type of entity (such as Fatuma or English) and the columns representing values attributed to that instance (such as Name or price). These terms are shown by Figure 3

For example, each row of a class table corresponds to a class, and a class corresponds to multiple students, so the relationship between the class table and the student table is one-to-many

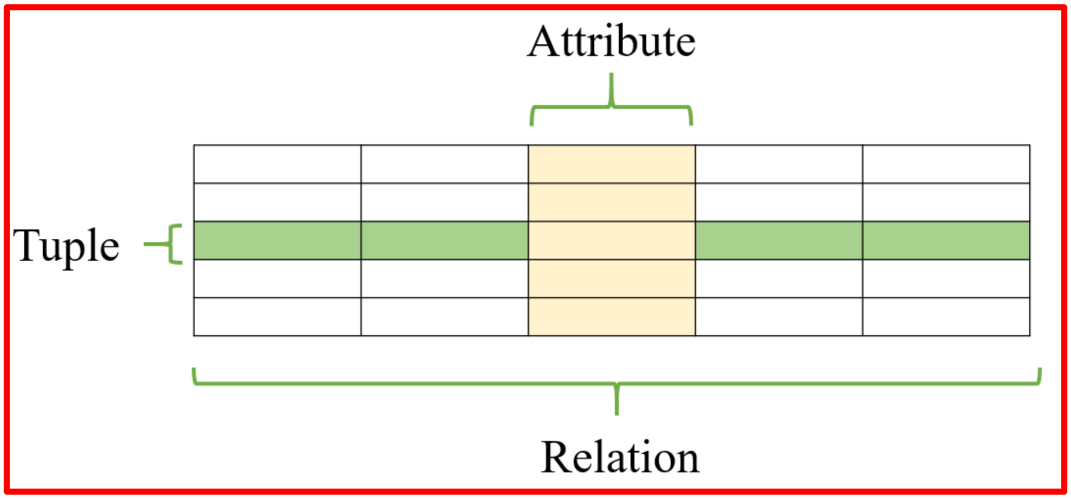


Figure 3 Terms in RDBS

Keys

Each row in a table has its own unique key named **Primary Key** (**PK**). Rows in a table can be linked to rows in other tables by adding a column for the unique key of the linked row (such columns are known as **Foreign Key (FK**).

The primary keys within a database are used to define the relationships among the tables. When a PK migrates to another table, it becomes a FK in the other table. When each cell can contain only one value and the PK migrates into a regular entity table, this design pattern can represent either a one-to-one or one-to-many relationship. Most relational database designs resolve many-to-many relationships by creating an additional table that contains the PKs from both of the other entity tables – the relationship becomes an entity; the resolution table is then named appropriately and the two FKs are combined to form a PK. The migration of PKs to other tables is the second major reason why system-assigned integers are used normally as PKs; there is usually neither efficiency nor clarity in migrating a bunch of other types of columns.

Relationships

Relationships are a logical connection between different tables, established on the basis of interaction among these tables.

Relational Database Terminologies

A relational database has become the major type of database. Other models besides the *relational model* include the hierarchical database model and the network model. Table 4 shows major terms used in relational database

Table 4 Relational database terms

SQL term	Relational database term	Description
Row	Tuple or record	A data set representing a single item
Column	Attribute or field	A labeled element of a tuple, e.g., Name, Course, Order, Customer_ID, Company
Table	Relation or Base relvar	A set of tuples sharing the same attributes; a set of columns and rows

View or result set	Derived relvar	Any set of tuples; a data report from the RDBMS in response to a query

Relations or tables

A *relation* is defined as a set of tuples that have the same attributes. A tuple usually represents an object and information about that object. Objects are typically physical objects or concepts. A relation is usually described as a table, which is organized into rows and columns. All the data referenced by an attribute are in the same domain and conform to the same constraints.

Tuples by definition are unique. If the tuple contains a candidate or primary key then obviously it is unique; however, a primary key need not be defined for a row or record to be a tuple. The definition of a tuple requires that it be unique, but does not require a primary key to be defined. Because a tuple is unique, its attributes by definition constitute a superkey.

Base and derived relations

In a relational database, all data are stored and accessed through relations. Relations that store data are called "base relations", and in implementations are called tables. Other relations that do not store data, but are computed by applying relational operations to other relations are called derived relations. In implementations these relations are called views or queries. Derived relations are convenient in that they act as a single relation, even though they may take information from several relations.

Domain

A domain describes the set of possible values for a given attribute, and can be considered a constraint on the value of the attribute. Mathematically, attaching a domain to an attribute means that any value for the attribute must be an element of the specified set. The string ‘NUFFIC’ belongs to the character datatype domain while value 789 belongs to the integer datatype domain. Common domain data types, their syntax and descriptions used in database are shown by Table 5.

Table 5 Common domain data types

Data Type	Description
CHARACTER(n)	Character string of fixed length n.
VARCHAR(n)	Variable length character string, maximum length n.
BINARY(n)	Fixed-length binary string, maximum length n.
BOOLEAN	Stores truth values - either TRUE or FALSE.
VARBINARY(n)	Variable length binary string, maximum length n.
INTEGER(p)	Integer numerical, precision p.
SMALLINT	Integer numerical precision 5.
INTEGER	Integer numerical, precision 10.
BIGINT	Integer numerical, precision 19.
DECIMAL (p, s)	Exact numerical, precision p, scale s.
NUMERIC (p, s)	Exact numerical, precision p, scale s.
FLOAT(p)	Approximate numerical, mantissa precision p.
REAL	Approximate numerical mantissa precision 7.
FLOAT	Approximate numerical mantissa precision 16.
DOUBLE PRECISION	Approximate numerical mantissa precision 16.
DATE TIME TIMESTAMP	Composed of a number of integer fields, representing an absolute point in time, depending on sub-type.
INTERVAL	Composed of a number of integer fields, representing a period of time, depending on the type of interval.
COLLECTION (ARRAY, MULTISET)	ARRAY (offered in SQL99) is a set-length and ordered the collection of elements.
XML	Stores XML data. It can be used wherever a SQL data type is allowed, such as a column of a table.

Constraints

Constraints make it possible to further restrict the domain of an attribute. For instance, a constraint can restrict a given integer attribute to values between 1 and n as illustrated by Table 5 above. Constraints provide one method of implementing business rules in the database and support subsequent data use within the application layer. SQL implements constraint functionality in the form of check constraints. Constraints restrict the data that can be stored in relations. These are usually defined using expressions that result in a Boolean value, indicating whether or not the data satisfies the constraint. Constraints can apply to single attributes, to a tuple (restricting combinations of attributes) or to an entire relation. Since every attribute has an associated domain, there are constraints (**domain constraints**). The two principal rules for the relational model are known as **entity integrity** and **referential integrity**.

Primary Key

Each relation/table has a primary key, this being a consequence of a relation being a set. A primary key uniquely specifies a tuple within a table. Figure 4 shows all possible keys with DBMS. Figures 5 and 6 shows examples of theses keys. While natural attributes (attributes used to describe the data being entered) are sometimes good primary keys, surrogate keys are often used instead. A surrogate key is an artificial attribute assigned to an object which uniquely identifies it (for instance, in a table of information about students at a school they might all be assigned a student ID in order to differentiate them). The surrogate key has no intrinsic (inherent) meaning, but rather is useful through its ability to uniquely identify a tuple. Another common occurrence, especially in regard to N:M cardinality is the composite key. A composite key is a key made up of two or more attributes within a table that (together) uniquely identify a record

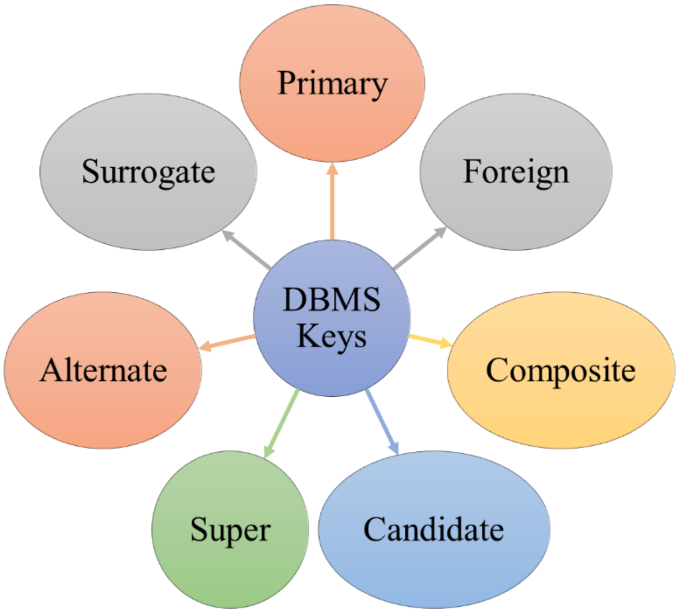
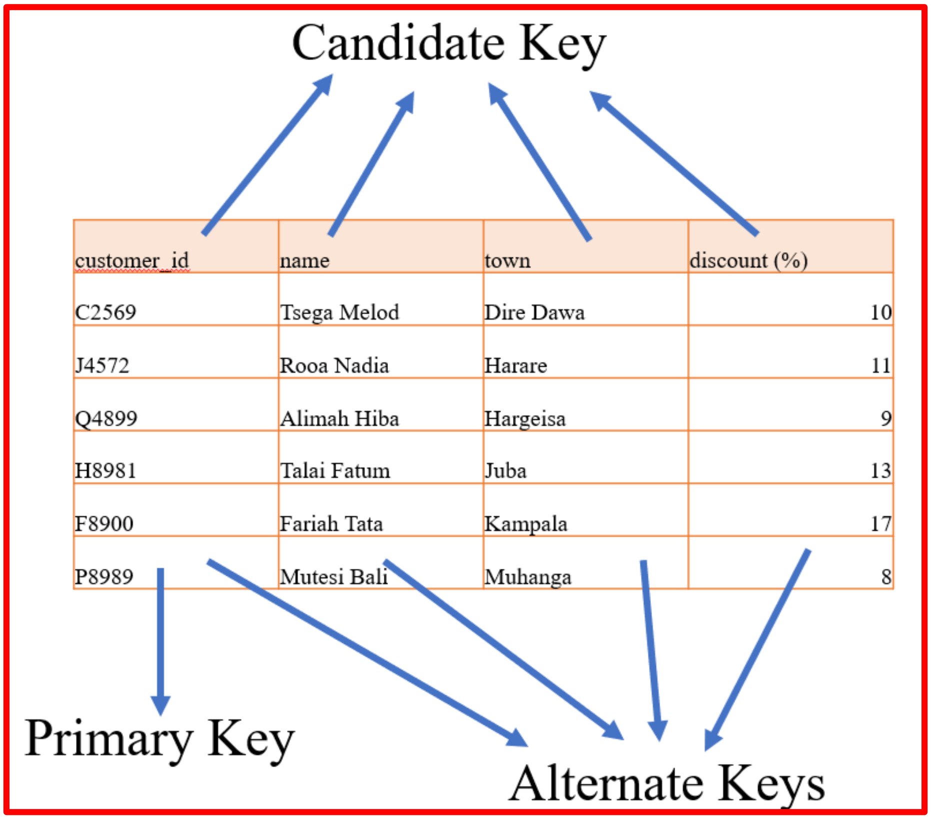
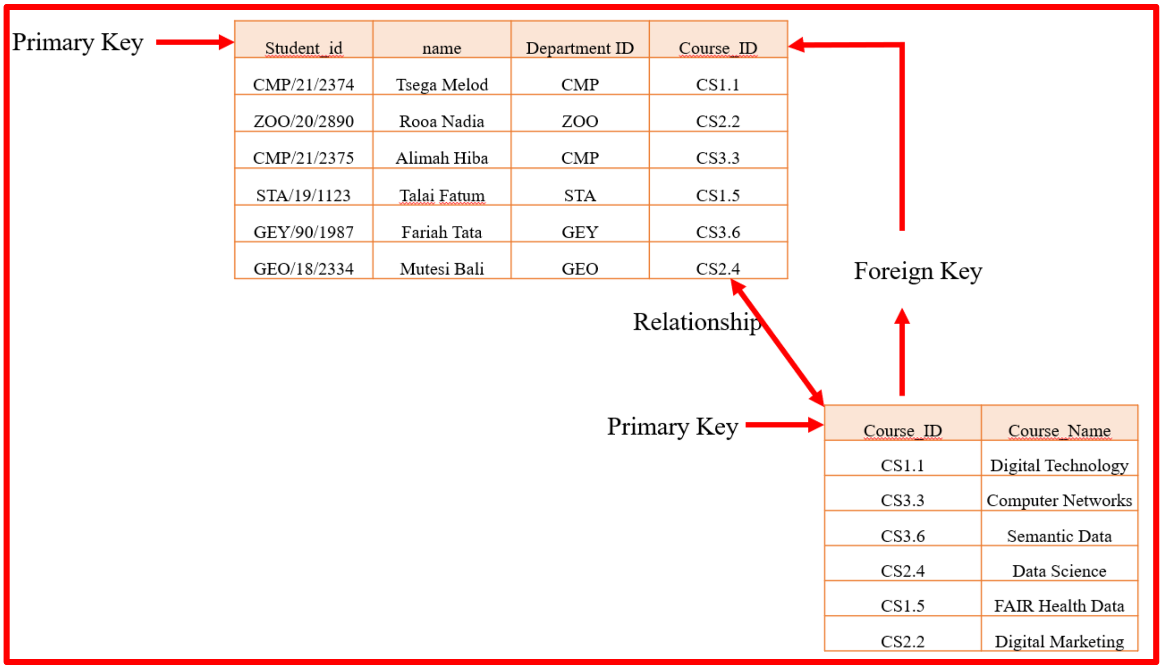


Figure 4 DBMS keys



customer_id	name	town	discount (%)
C2569	Tsega Melod	Dire Dawa	10
J4572	Rooa Nadia	Harare	11
Q4899	Alimah Hiba	Hargeisa	9
H8981	Talai Fatum	Juba	13
F8900	Fariah Tata	Kampala	17
P8989	Mutesi Bali	Muhanga	8

Figure 5 Examples of DBMS keys



Student_id	name	Department ID	Course_ID
CMP/21/2374	Tsega Melod	CMP	CS1.1
ZOO/20/2890	Rooa Nadia	ZOO	CS2.2
CMP/21/2375	Alimah Hiba	CMP	CS3.3
STA/19/1123	Talai Fatum	STA	CS1.5
GEY/90/1987	Fariah Tata	GEY	CS3.6
GEO/18/2334	Mutesi Bali	GEO	CS2.4

Course_ID	Course_Name
CS1.1	Digital Technology
CS3.3	Computer Networks
CS3.6	Semantic Data
CS2.4	Data Science
CS1.5	FAIR Health Data
CS2.2	Digital Marketing

Figure 6 Relationship between Primary and Foreign keys

Foreign key

A foreign key is a field in a relational table that matches the primary key column of another table. It relates the two keys. Foreign keys need not have unique values in the referencing relation. A foreign key can be used to cross-reference tables, and it effectively uses the values of attributes in the referenced relation to restrict the domain of one or more attributes in the referencing relation. The concept is described formally as: "For all tuples in the referencing relation projected over the referencing attributes, there must exist a tuple in the referenced relation projected over those same attributes such that the values in each of the referencing attributes match the corresponding values in the referenced attributes."

8.2.3 Multidimensional DBMS

Data modeling is the analysis of data objects and their relationships to other data objects. Data modeling is often the first step in database design and object-oriented programming as the designers first create a conceptual model of how data items relate to each other. Data modeling involves a progression from conceptual model to logical model to physical schema.

8.2.4 Data Modelling

This is the process of creating a data model for the data to be stored in a database. This data model is a conceptual representation of data items, the relationship between these different data items, and the rules. Data modeling helps in the visual representation of data and enforces business rules, regulatory compliances, and government policies on the data. Data Models ensure consistency in naming conventions, default values, semantics, security while ensuring quality of the data.

Data Model

The **Data Model** is defined as an abstract model that organizes data description, data semantics, and consistency constraints of data. The data model emphasizes on what data is needed and how it should be organized instead of what operations will be performed on data. Data Model is like an architect's building plan, which helps to build conceptual models and set a relationship between data items.

The major two types of Data Modeling Techniques are

1. Entity Relationship (E-R) Model
2. UML (Unified Modelling Language)

Uses of Data Model

The primary goal of using data model is:

a. Ensures that all data items needed by the database are accurately represented. Omission of data will lead to creation of faulty reports and produce incorrect results.

b. A data model helps design the database at the conceptual, physical and logical levels.

c. Data Model structure helps to define the relational tables, primary and foreign keys and stored procedures.

d. It provides a clear picture of the base data and can be used by database developers to create a physical database.

e. It is also helpful to identify missing and redundant data.

f. Though the initial creation of data model is labor and time consuming, in the long run, it makes your IT infrastructure upgrade and maintenance cheaper and faster.

Data Abstraction

There are mainly five levels of abstraction or details of data in data models but we will talk about three in this lesson which are shown by Figure 7: conceptual data models, logical data models, and physical data models. The data models are used to represent the data and how it is stored in the database and to set the relationship between data items.

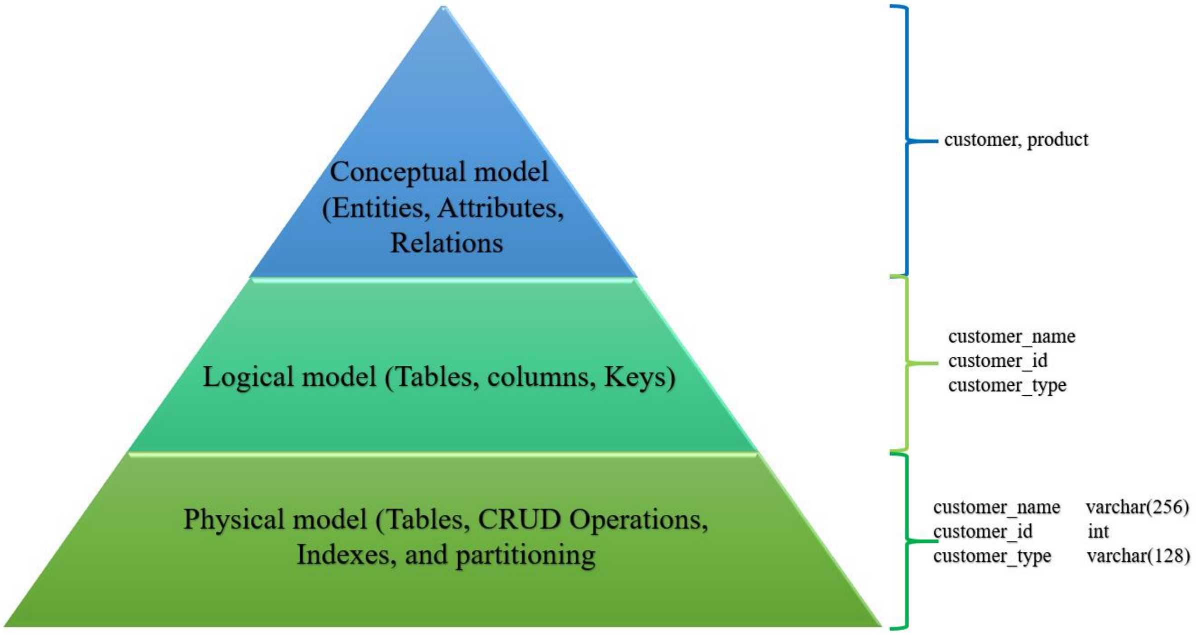

Conceptual model (Entities, Attributes, Relations
Logical model (Tables, columns, Keys)
Physical model (Tables, CRUD Operations, Indexes, and partitioning
customer, product
customer_name
customer_id
customer_type
customer_name varchar(256)
customer_id int
customer_type varchar(128)

Figure 7: Data level abstraction

1. **Conceptual Data Model:** This Data Model defines **WHAT** the system contains. It is an organized view of database concepts and their relationships. Its purpose is to create a conceptual data model by establishing the entities, their attributes, and relationships. This model is usually created by Business stakeholders and data architects and there is hardly any detail available on the actual database structure. .

   The 3 basic items at this level of Data Model are

   Entity: A real-world thing

   Attribute: Characteristics or properties of an entity

   Relationship: Dependency or association between two entities

Examples of items at this level are Data model example as represented by Figure 8:

a. Customer and Product are two entities.

b. Customer ID and name are attributes of the Customer entity

c. Product name and price are attributes of product entity

d. Sale is the relationship between the customer and product

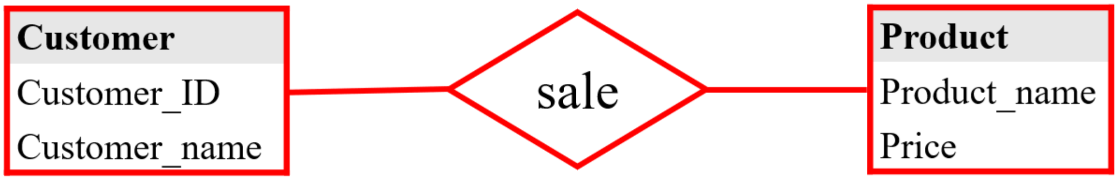


Figure 8 Conceptual model

2. **Logical Data Model:** This data model defines **HOW** the system should be implemented regardless of the DBMS. It is used to define the structure of data elements and to set relationships between them. The logical data model adds extra details to the conceptual data model elements as shown by Figure 9. The advantage of using a Logical data model is to provide a foundation to form the base for the Physical model. However, the modeling structure remains generic. This model is typically created by Data Architects and Business Analysts. The purpose is to developed technical map of rules and data structures.

Characteristics of a Logical data model

a. It describes data needs for a single project but could integrate with other logical data models based on the scope of the project.

b. It is designed and developed independently from the DBMS.

c. The data attributes will have datatypes with exact precisions and length.

d.

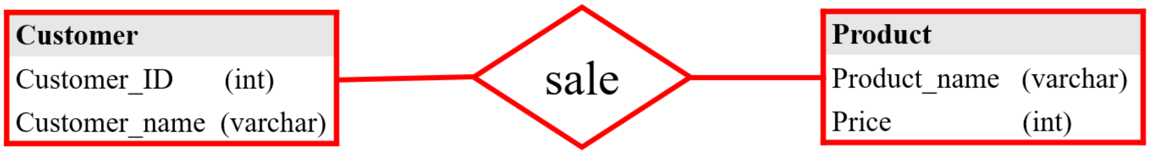


Figure 9 Logical model

3. **Physical Data Model**: This Data Model describes **HOW** the system will be implemented using a specific DBMS system. It describes a database-specific implementation of the data model as shown by Figure 10. It offers database abstraction and helps generate the schema. The physical data model also helps in visualizing database structure by replicating database column keys, constraints, indexes, triggers, and other RDBMS features. This model is typically created by Data Base Administrator (DBA) and developers.

Characteristics of a physical data model:

a. The physical data model describes data need for a single project or application though it may be integrated with other physical data models based on project scope.

b. Data Model contains relationships between tables that which addresses cardinality and nullability of the relationships.

c. Developed for a specific version of a DBMS, location, data storage or technology to be used in the project.

d. Columns should have exact datatypes, lengths assigned and default values.

e. Primary and Foreign keys, views, indexes, access profiles, and authorizations, etc. are defined.

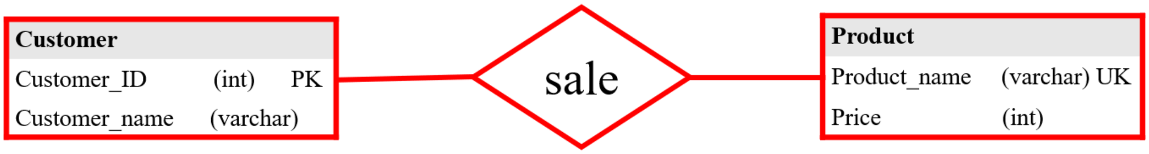


Figure 10 Physical model (PK: Primary key, UK: Unique Key

Advantages of Data model:

a. The main goal of a designing data model is to make certain that data objects offered by the functional team are represented accurately.

b. The data model should be detailed enough to be used for building the physical database.

c. The information in the data model can be used for defining the relationship between tables, primary and foreign keys, and stored procedures.
d. Data Model helps business to communicate the within and across organizations.
e. Data model helps to documents data mappings in ETL process
f. Help to recognize correct sources of data to populate the model

Disadvantages of Data model:

a. To develop Data model, one should know physical data stored characteristics.
b. This is a navigational system produces complex application development, management. Thus, it requires a knowledge of the biographical truth.
c. Even smaller change made in structure require modification in the entire application.
d. There is no set data manipulation language in DBMS.

E-R Model

An entity-relationship model (ER) diagram is a graphical representation of entities and their relationships to each other, typically used in computing in regard to the organization of data within databases or information systems. An entity is a piece of data-an object or concept about which data is stored. The entity-relationship (ER) data model allows the description of the data involved in a real-world enterprise in terms of objects and their relationships and is widely used to develop an initial database design. An ER diagram shows the relationship among entity sets. An **entity** is an object in the real world that is different from other objects. Examples are person, student, department, course, employee etc. A group or collection of similar entities is called an **entity set**. An attribute describes the characteristics of an entity. All entities in a given entity set have the same attributes. Example of attribute of a person are name, color, gender etc. Another example of attribute of a student are name, course, student ID, department ID etc. Another example attribute of entity employee will be employee name, Employee ID, Department ID etc. Figures 11-12 shows an example of an entity with three (3) attributes

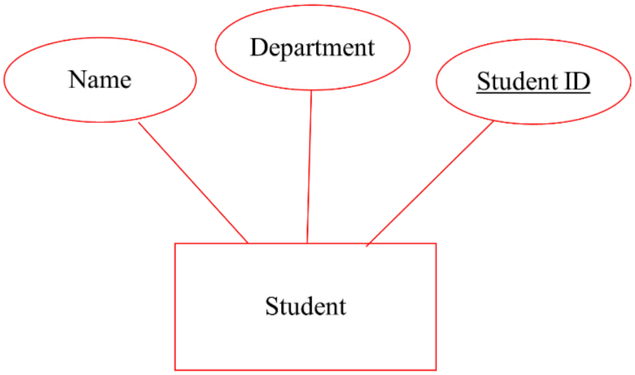


Figure 11 Student entity set

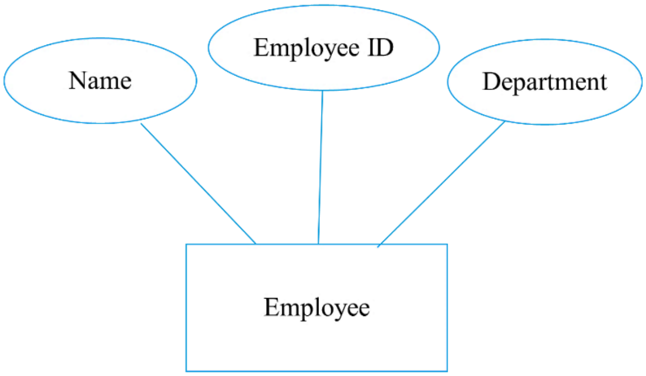


Figure 12 Employee entity set

An entity key is the minimum set of attributes that uniquely identifies an entity. A primary key is one attribute that uniquely identifies an entity. Examples are student ID, department ID, Employee ID etc. The key is denoted by an underline a particular attribute. An example is shown by Figure 13 (a) and (b)

(a) (b)

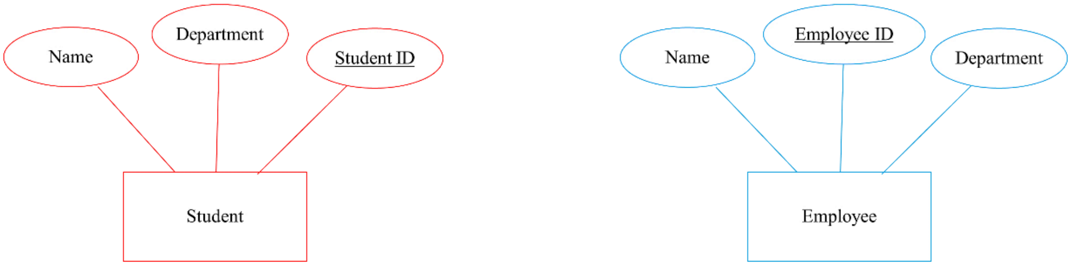
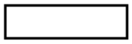
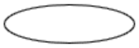
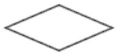
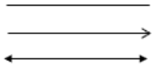
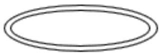
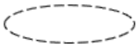
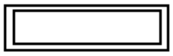
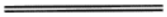


Figure 13 (a) and (b) showing key of student and Employee entity set

Modelling an entity

An entity set is represented by a rectangle, and an attribute is represented by an oval. Each attribute in the primary key is underlined. The domain information could be listed along with the attribute name. The full representation is illustrated by Table 6

Table 6 Geometric shapes used for an E-R modelling

SHAPE	SHAPE NAME	DESCRIPTION
	Rectangle	Entity sets
	Ellipses	Attributes
	Diamond	Relationship Set
	Lines	They link attributes to Entity Sets and Entity sets to Relationship Set
	Double Ellipses	Multivalued Attributes
	Dashed Ellipses	Derived Attributes
	Double Rectangles	Weak Entity Sets
	Double Lines	Total participation of an entity in a relationship set

Components of an ER Diagram

The three main components of the ER Model (ERM) are entities, attributes and relationships. Figure 14 shows the different elements of the components of an ER diagram.

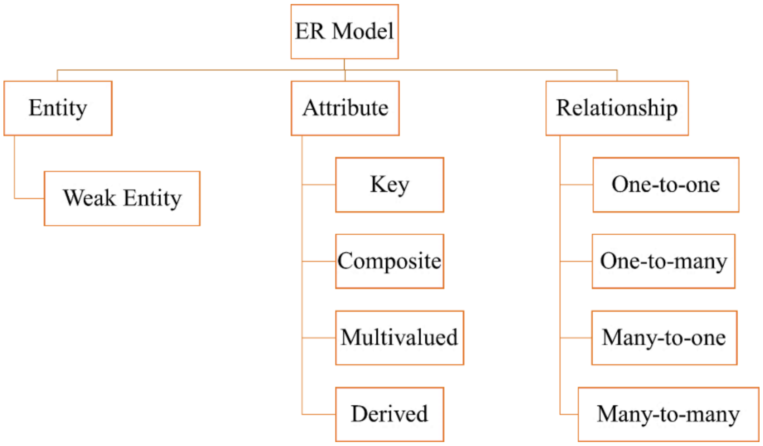


Figure 14 Components of an ER diagram

1. **Entity**

An entity is an object or component of data. An entity is represented as rectangle in an ER diagram. Figure 15 shows an ER diagram with two entities: Student and College. The relationship between the two entities is many-to-one relationship. Meaning that many students study in a single (1) university.

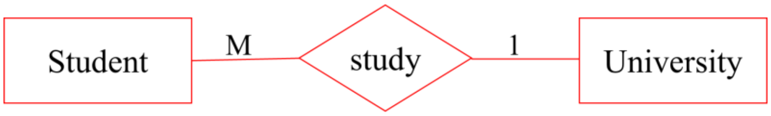


Figure 15 Two entity sets

Weak Entity:

An entity that cannot be uniquely identified by its own attributes and relies on the relationship with other entity is called weak entity. The weak entity is represented by a double rectangle. For example in Figure 16 – a bank account cannot be uniquely identified without knowing the bank to which the account belongs, so bank account is a weak entity.

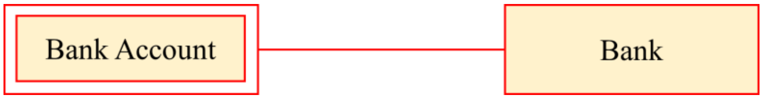


Figure 16 Weak entity set

Attribute

An attribute describes the property of an entity. An attribute is represented as Oval in an ER diagram. There are four types of attributes:

1. Key attribute
2. Composite attribute
3. Multivalued attribute
4. Derived attribute

Key attribute:

A key attribute can uniquely identify an entity from an entity set. For example, student ID can uniquely identify a student from a set of students. Key attribute is represented by oval same as other attributes however the text of key attribute is underlined as shown by Figure 17.

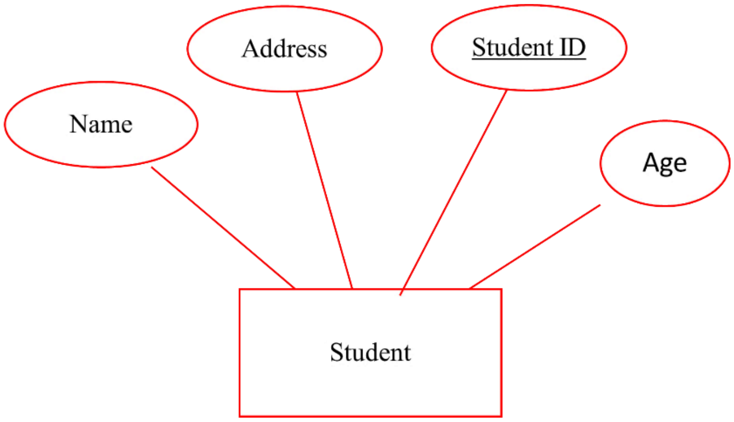


Figure 17 showing key of student entity set

Composite attribute

An attribute that is a combination of other attributes is known as composite attribute. Example illustrated by Figure 18 shows that student name is composed of first and last name. Another example illustrated by Figure 20, for student entity, the student address is a composite attribute as an address is composed of other attributes like street number, street name and area code.

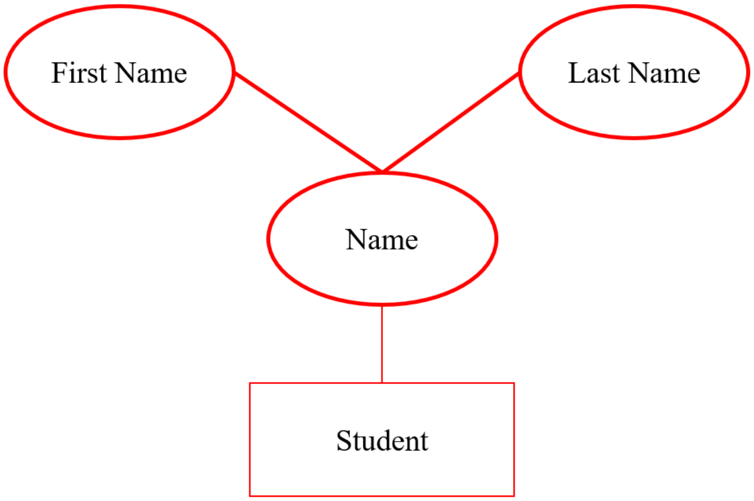


Figure 19 Composite model of student’s name

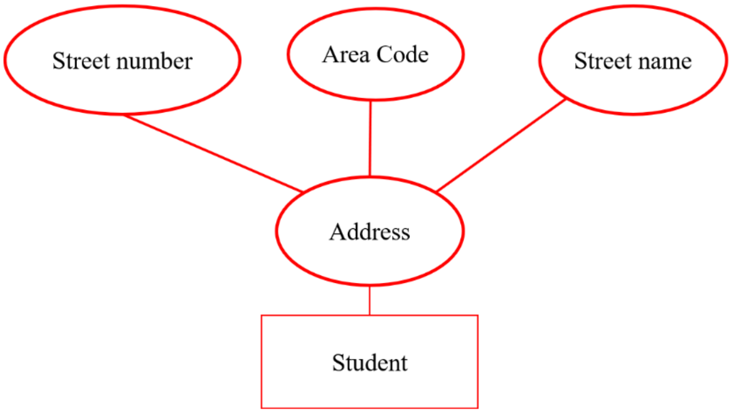


Figure 20 Composite model of student’s address

3. Multivalued attribute:

An attribute that can hold multiple or more than one values is known as multivalued attribute. It is represented with double ovals in an ER Diagram. An example is illustrated by Figure 21 where a student can have more than one phone numbers so the phone number attribute is multivalued.

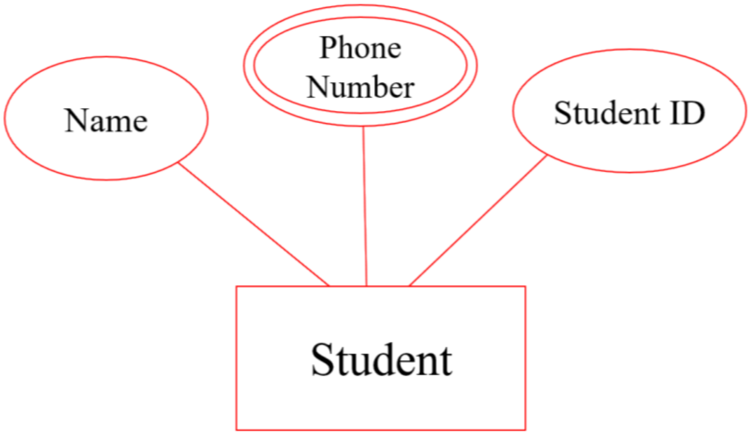


Figure 21 Student multivalued attribute

4. Derived attribute:

A derived attribute is one whose value is dynamic and derived from another attribute. It is represented by dashed oval in an ER Diagram. An example is illustrated by Figure 22 for a student age is a derived attribute as it changes over time and can be derived from another attribute DOB (Date of Birth).

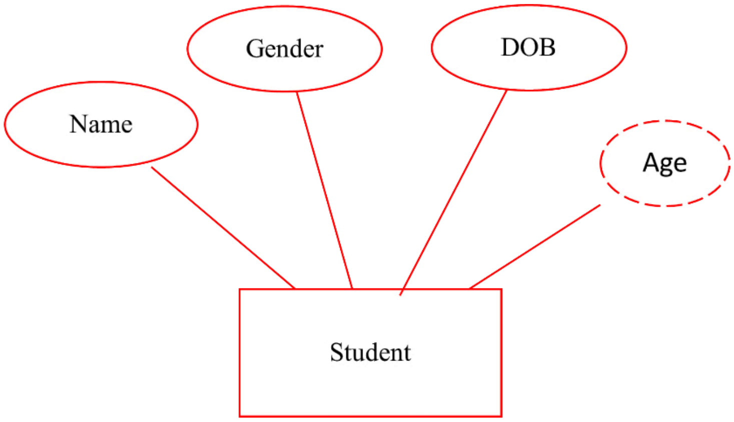


Figure 22 Student derived attribute

3. Relationship

A relationship is represented by diamond shape in ER diagram, it shows the relationship among entities. There are four types of relationships:

1. One to One
2. One to Many

3. Many to One
4. Many to Many

1. One to One Relationship

When a single instance of an entity is associated with a single instance of another entity then it is called one to one relationship. An example is illustrated in Figure 23 where a student has only one Student ID and a Student ID is given to one student.

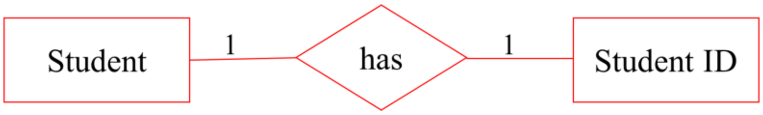


Figure 23 one-to-one relationship

2. One to Many Relationship

When a single instance of an entity is associated with more than one or multiple instances of another entity then it is called one-to-many relationship. An example is illustrated by Figure 24 where a University has many departments but a department cannot more than one university.

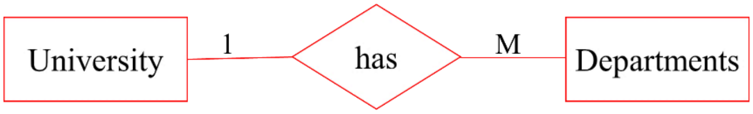


Figure 24 one-to-many relationship

3. Many to One Relationship

When more than one instances of an entity are associated with a single instance of another entity then it is called many to one relationship. An example is illustrated by Figure 25 where many students can study in a single university but a student cannot study in many universities at the same time.

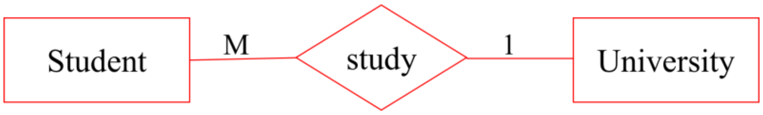


Figure 25 many-to-one relationship

4. Many to Many Relationship

When more than one instances of an entity are associated with more than one instances of another entity then it is called many to many relationships. An example is illustrated by Figure 26 where many students can be assigned to many projects and many projects can be assigned to many students.

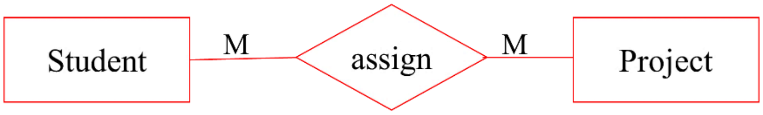


Figure 26 many-to-many relationship

E-R model using Crow's feet

Crow's foot diagrams represent entities as boxes, and relationships as lines between the boxes as presented by Figures 27-28

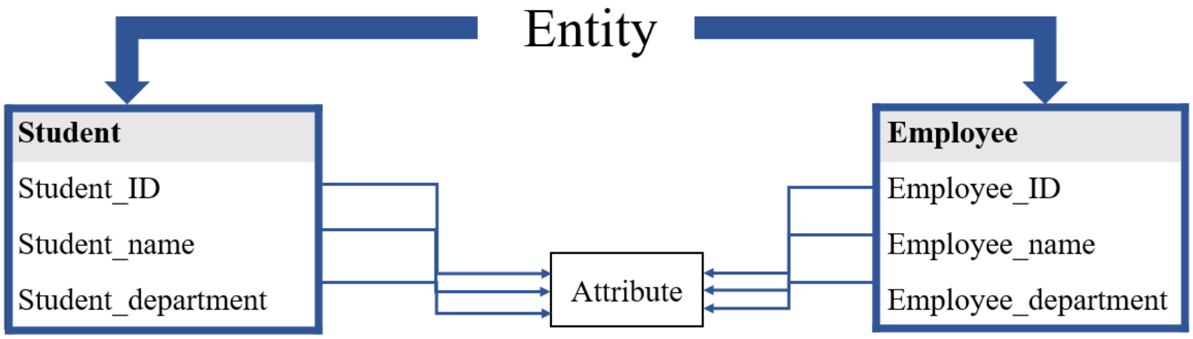


Figure 27 Entity representation for Crow's feet model

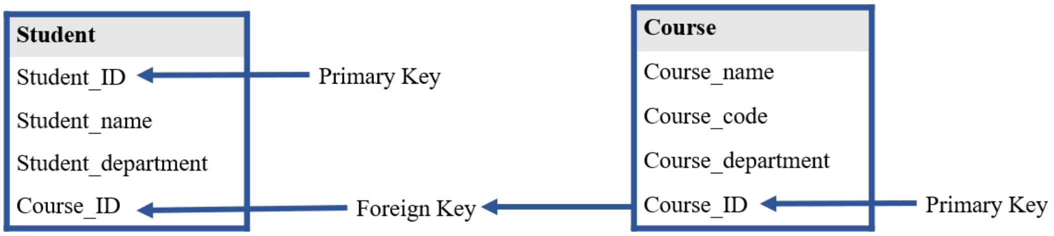


Figure 28 Primary and Foreign keys

Different shapes at the ends of these lines represent the relative cardinality of the relationship. The three shapes used to illustrate cardinality are presented in Tables 7-8:

Table 7 Shapes for the cardinality

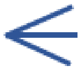
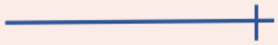
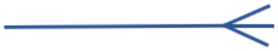
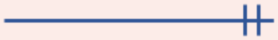
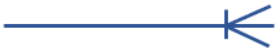
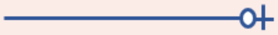
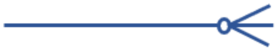
Cardinality	Shape
A loop represents "zero"	○
A crow's foot represents "many" or "infinite"	
A cross represents "one"	+

Table 8 Symbols in crow's foot notation

Shape	Name	Description
	One	Cross: Minimum one, maximum one (optional)
	many	Crow’s feet: Many
	One and only one (1)	Cross and cross: Minimum one, maximum one (mandatory)
	One or many (1…*)	Cross and crow’s foot: Minimum one, maximum many (mandatory)
	Zero or one (0…1)	Loop and cross: Minimum zero, maximum one (optional)
	Zero or many (0…*)	Loop and crow's foot: Minimum zero, maximum many (optional)

Each entity in the crow’s feet model is represented as a table with each field acting as an attribute of the entity containing it. Entities are connected using a system of notation called crow’s foot notation. The styling of the endpoint of each line distinguishes the relationship as illustrate by Figure 29.

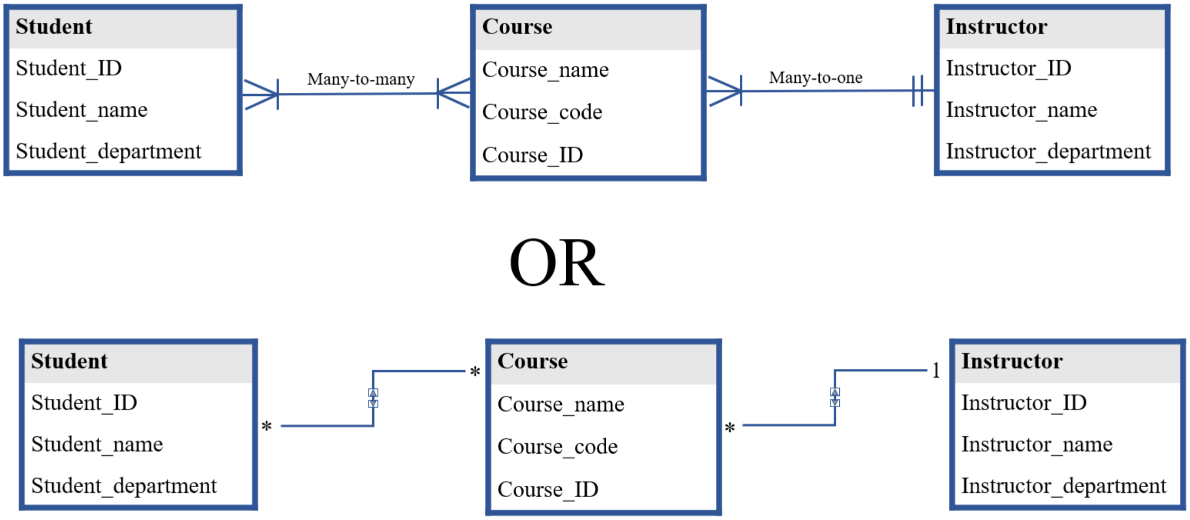


Figure 29 Entity set using Crow's Feet notation

Tips for designing the diagram

- Identify and draw all the entities you need in the diagram.
- Determine which entities have relationships with each other, and connect them. Some of the entities will have relationships while some may have multiple relationships.)
- Each entity should appear only once in the diagram.
- Revise and remove any redundant relationships?
- Review your diagram for any missing or redundant entity or relationship

Peer to Peer Interaction

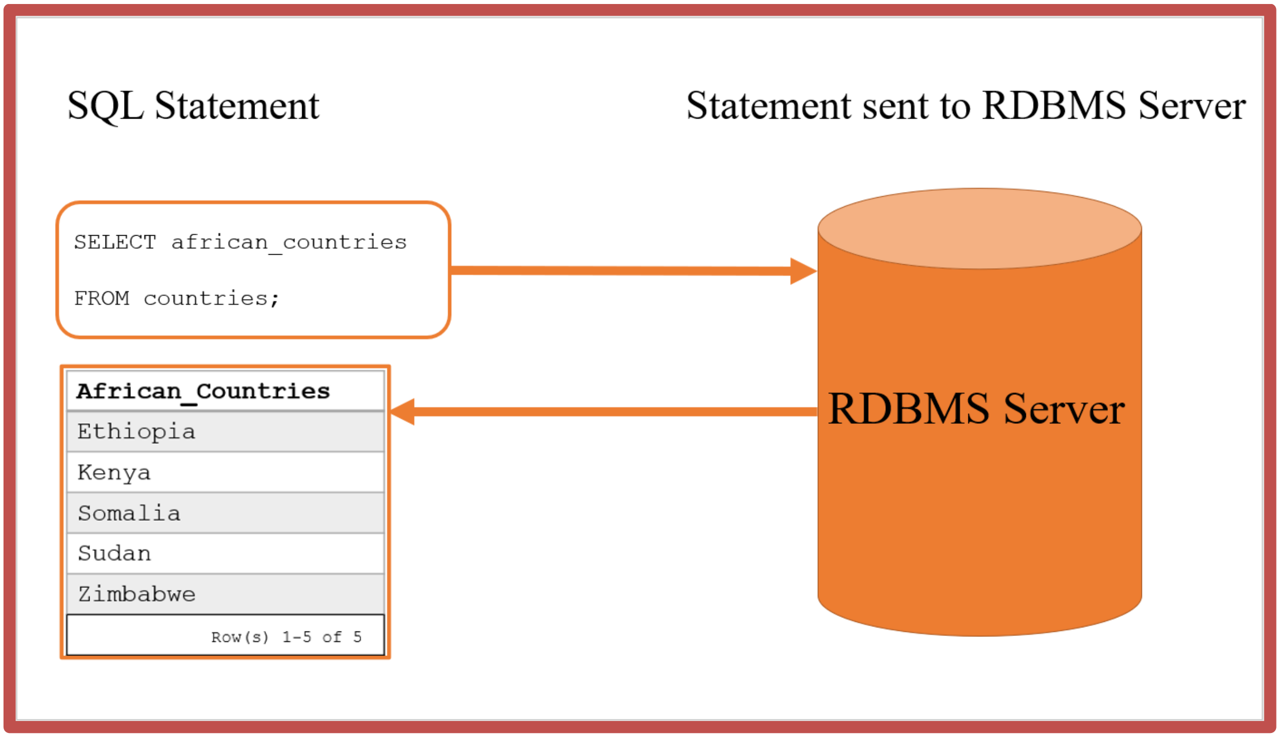
Prepare an E-R using crow’s feet alongside their cardinality using the following scenario

a. Company

b. University

8.3 Structured Query Language (SQL)

Structured **Q**uery **L**anguage (SQL) is a programming language designed to manage data stored in relational databases. SQL operates through simple, declarative statements. This keeps data accurate and secure, and it helps maintain the integrity of databases, regardless of its size. It is a standard computer programming language used for accessing and manipulating database systems. It is used to manage data in RDBMS which stores data in the form of tables and relationship between data is also stored in the form of tables. SQL statements are used to retrieve and update data in a database. Figure 20 shows the communication with an RDBMS using SQL



```
SELECT african_countries

FROM countries;
```

```
African_Countries
Ethiopia
Kenya
Somalia
Sudan
Zimbabwe
                Row(s) 1-5 of 5
```

Figure 20 SQL and RDBMS

8.3.1 Constructs of SQL

This section consists of the lists and description of the vital elements of SQL.

a. Queries: Retrieves data given some conditions.

b. Statements: This is the syntax of the query language. It could be Controls transactions, program flow, connections, sessions, or diagnostics.

c. Clauses: Components of Queries and Statements.

d. Expressions: Combination of symbols and operators and a key part of the SQL statements.

e. Predicates: Specifies conditions.

8.3.2 Database Languages

a. **Data Definition Language (DDL):** SQL can be used to define data structures like database schemas. The main statements used to do this are CREATE, ALTER, and DROP. This allows users to create, change, and remove elements from the databases they're working with (tables, indexes, users, etc.).

b. **Data Manipulation Language (DML):** SQL can also be used to change the database itself with commands like INSERT, DELETE, and UPDATE. These commands affect the contents of a database, ensuring that users have only the information they need in the form that will work for them.

c. **Transaction Control Language (TCL):** SQL controls the data manipulation process. It monitors changes made to data until the session is finished. The COMMIT command locks in recent changes, and ROLLBACK turns the clock back to the last committed save state. SAVEPOINT is a middle ground that a user can ROLLBACK to when needed.

d. **Data Control Language (DCL):** SQL can be used to safeguard datasets by GRANTing and REVOKEing user privileges at either the system level (ability to create sessions, tables, etc.) or object level (what queries, commands, etc. the user can perform).

8.3.3 SQL Statements

This section describes popular SQL statements their syntax. The syntax column shows how each of the statements can be used alongside some other SQL statements too. Figure 9 presentments the SQL statements

Table 9 SQL STATEMENTS

SQL Statement	Description	SQL Syntax
ALTER TABLE	Allows you to add columns to a table in a database.	**ALTER TABLE** table_name **ADD** column_name datatype;

AND	is an operator that joins two conditions. Both conditions must be true for the row to be included in the result set.	**SELECT** column_name(s)
		FROM table_name
		WHERE column_1 = value_1
		AND column_2 = value_2;

AS	is a keyword in SQL that allows you to rename a column or table using an *alias*.	**SELECT** column_name **AS 'Alias'**
		FROM table_name;

AVG()	AVG() is an aggregate function that returns the average value for a numeric column	**SELECT AVG(**column_name**)**
		FROM table_name**;**

BETWEEN	The BETWEEN operator is used to filter the result set within a certain range. The values can be numbers, text or dates.	**SELECT** column_name(s)
		FROM table_name
		WHERE column_name **BETWEEN** value_1 **AND** value_2**;**

CASE	CASE statements are used to create different outputs (usually in the SELECT statement). It is SQL's way of handling if-then logic.	**SELECT** column_name**,**
		  CASE
		    WHEN condition **THEN** 'Result_1'
		    WHEN condition **THEN** 'Result_2'
		    ELSE 'Result_3'
		  END
		FROM table_name;

COUNT()	COUNT() is a function that takes the name of a column as an argument and counts the number of rows where the column is not NULL.	**SELECT COUNT(**column_name**)**
		FROM table_name**;**

CREATE TABLE
 	
CREATE DATABASE
	CREATE TABLE creates a new table in the database or a new database. It allows you to specify the name of the table and the name of each column	**CREATE TABLE** table_name (
		  column_1 datatype,
		  column_2 datatype,
		  column_3 datatype
		);

	in the table. A database could also be created with the command CREATE	

DELETE	DELETE statements are used to remove rows from a table.	**DELETE FROM** table_name
		WHERE some_column = some_value**;**

DROP TABLE table_name **DROP DATABASE database_name**	Delete a table in a database or deletes an entire database	

GROUP BY	GROUP BY is a clause in SQL that is only used with aggregate functions. It is used in collaboration with the SELECT statement to arrange identical data into groups.	**SELECT** column_name**, COUNT(*)**
		FROM table_name
		GROUP BY column_name**;**

HAVING	HAVING was added to SQL because
	the WHERE keyword could not be used with aggregate functions.
		SELECT column_name**, COUNT(*)**
		FROM table_name
		GROUP BY column_name
		HAVING COUNT(*) > value**;**

INNER JOIN	An inner join will combine rows from different tables if the *join condition* is true.	**SELECT** column_name(s)
		FROM table_1
		JOIN table_2
		ON table_1.column_name = table_2.column_name;

INSERT	INSERT statements are used to add a new row to a table.	**INSERT INTO** table_name (column_1, column_2, column_3)
		VALUES (value_1, 'value_2', value_3**);**

IS NULL / IS NOT NULL	IS NULL and IS NOT NULL are operators used with the WHERE clause to test for empty values.	**SELECT** column_name(s)
		FROM table_name
		WHERE column_name **IS NULL;**

LIKE	LIKE is a special operator used with the WHERE clause to search for a specific pattern in a column.	**SELECT** column_name(s)
		FROM table_name
		WHERE column_name **LIKE** pattern**;**

LIMIT	LIMIT is a clause that lets you specify the maximum number of rows the result set will have.	**SELECT** column_name(s)
		FROM table_name
		LIMIT number**;**

MAX()	MAX() is a function that takes the name of a column as an argument and returns the largest value in that column.	**SELECT MAX(**column_name**)**
		FROM table_name**;**

MIN()	MIN() is a function that takes the name of a column as an argument and returns the smallest value in that column.	**SELECT MIN(**column_name**)**
		FROM table_name**;**

OR	OR is an operator that filters the result set to only include rows where either condition is true.	**SELECT** column_name
		FROM table_name
		WHERE column_name = value_1
		OR column_name = value_2;

ORDER BY	ORDER BY is a clause that indicates you want to sort the result set by a particular	**SELECT** column_name
		FROM table_name
		ORDER BY column_name **ASC | DESC;**

	column either alphabetically or numerically.	

OUTER JOIN	An outer join will combine rows from different tables even if the join condition is not met. Every row in the *left* table is returned in the result set, and if the join condition is not met, then NULL values are used to fill in the columns from the *right* table.	**SELECT** column_name(s)
		FROM table_1
		LEFT JOIN table_2
		ON table_1.column_name = table_2.column_name;

ROUND()	ROUND() is a function that takes a column name and an integer as arguments. It rounds the values in the column to the number of decimal places specified by the integer.	**SELECT ROUND(**column_name, integer**)**
		FROM table_name**;**

SELECT	SELECT statements are used to fetch data from a database. Every query will begin with SELECT.	**SELECT** column_name
		FROM table_name**;**

SELECT DISTINCT	SELECT DISTINCT specifies that the statement is going to be a query that returns unique values in the specified column(s).	**SELECT DISTINCT** column_name
		FROM table_name**;**

SUM	SUM() is a function that takes the name of a column as an argument and returns the sum of all the values in that column.	**SELECT SUM**(column_name**)**
		FROM table_name**;**

UPDATE	UPDATE statements allow you to edit rows in a table.	**UPDATE** table_name
		SET some_column = some_value
		WHERE some_column = some_value**;**

WHERE	WHERE is a clause that indicates you want to filter the result set to include only rows where the following *condition* is true.	**SELECT** column_name(s)
		FROM table_name
		WHERE column_name operator value**;**

WITH	WITH clause lets you store the result of a query in a temporary table using an alias. You can also define multiple temporary tables using a comma and with one instance of the WITH keyword.
	The WITH clause is also known as common table expression (CTE) and subquery factoring.
		WITH temporary_name **AS (**
		  SELECT *
		  FROM table_name**)**
		SELECT *
		FROM temporary_name
		WHERE column_name operator value;

Arithmetic Function
abs()
ceil()
floor()
exp()
ln()
mod()
power()
sqrt()
	A mathematical function executes a mathematical operation usually based on input values that are provided as arguments, and return a numeric value as the result of the operation.	**SELECT ABS**(column_name)
		FROM table_name**;**

Character Function
lower()
upper()
trim()
translate()
	A character or string function is a function which takes one or more characters or numbers as parameters and returns a character value.	

Character Function
lower()
upper()
trim()
translate()
	A character or string function is a function which takes one or more characters or numbers as parameters and returns a character value.	

Table 10 Customer

```
customer_id |    name      |  town     | discount (%)
-------------+--------------+-----------+------------
     C2569| Tsega Melod  | Dire Dawa |       10
     J4572| Rooa Nadia   | Harare    |       11
     Q4899| Alimah Hiba  | Hargeisa  |        9
     H8981| Talai Fatum  | Juba      |       13
     F8900| Fariah Tata  | Kampala   |       17
     P8989| Mutesi Bali  | Muhanga   |        8
```

Example 1: Write a SQL statement to display all the information of all Customers in Table 10

Answer: SELECT * FROM customer;

Output of the Query:

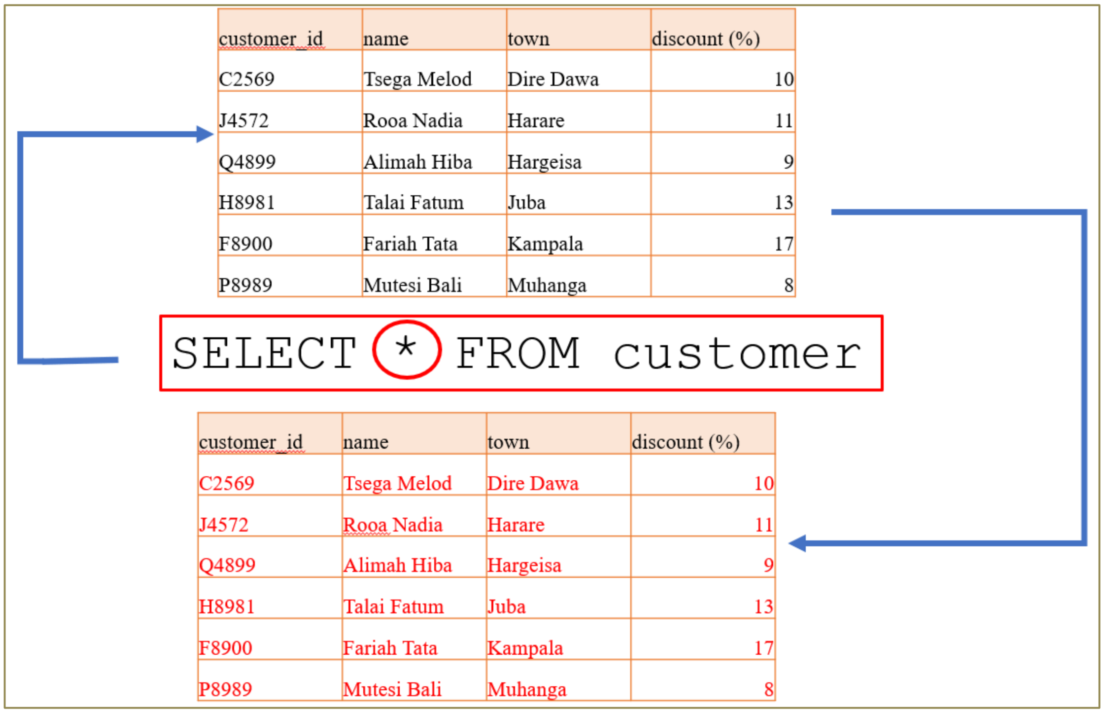
```
customer_id    name         town         discount (%)


C2569          Tsega Melod   Dire Dawa      10
J4572          Rooa Nadia    Harare         11
Q4899          Alimah Hiba   Hargeisa        9
H8981          Talai Fatum   Juba           13
F8900          Fariah Tata   Kampala        17
P8989          Mutesi Bali   Muhanga         8
```

	Use * to get a complete list of the columns from a table.

Alternate command: SELECT customer_id, name, town, discount

Figure 31 shows a Pictorial Representation of Example 1



customer_id	name	town	discount (%)
C2569	Tsega Melod	Dire Dawa	10
J4572	Rooa Nadia	Harare	11
Q4899	Alimah Hiba	Hargeisa	9
H8981	Talai Fatum	Juba	13
F8900	Fariah Tata	Kampala	17
P8989	Mutesi Bali	Muhanga	8

```
SELECT * FROM customer
```

customer_id	name	town	discount (%)
C2569	Tsega Melod	Dire Dawa	10
J4572	Rooa Nadia	Harare	11
Q4899	Alimah Hiba	Hargeisa	9
H8981	Talai Fatum	Juba	13
F8900	Fariah Tata	Kampala	17
P8989	Mutesi Bali	Muhanga	8

Figure 31 Pictorial Representation of Example 1

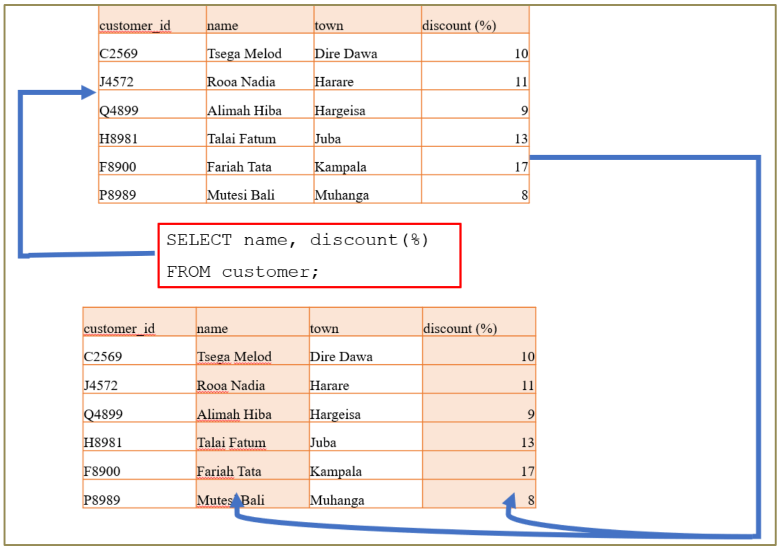
Example 2: Write a query to display only name and discount from table Customer.

Answer:

```
SELECT name, discount(%)
FROM customer;
```

Output of the Query:

name	discount (%)
Tsega Melod	10
Rooa Nadia	11
Alimah Hiba	9
Talai Fatum	13
Fariah Tata	17
Mutesi Bali	8



customer_id	name	town	discount (%)
C2569	Tsega Melod	Dire Dawa	10
J4572	Rooa Nadia	Harare	11
Q4899	Alimah Hiba	Hargeisa	9
H8981	Talai Fatum	Juba	13
F8900	Fariah Tata	Kampala	17
P8989	Mutesi Bali	Muhanga	8

```
SELECT name, discount(%)
FROM customer;
```

customer_id	name	town	discount (%)
C2569	Tsega Melod	Dire Dawa	10
J4572	Rooa Nadia	Harare	11
Q4899	Alimah Hiba	Hargeisa	9
H8981	Talai Fatum	Juba	13
F8900	Fariah Tata	Kampala	17
P8989	Mutesi Bali	Muhanga	8

Figure 31 Pictorial Representation of Example 2

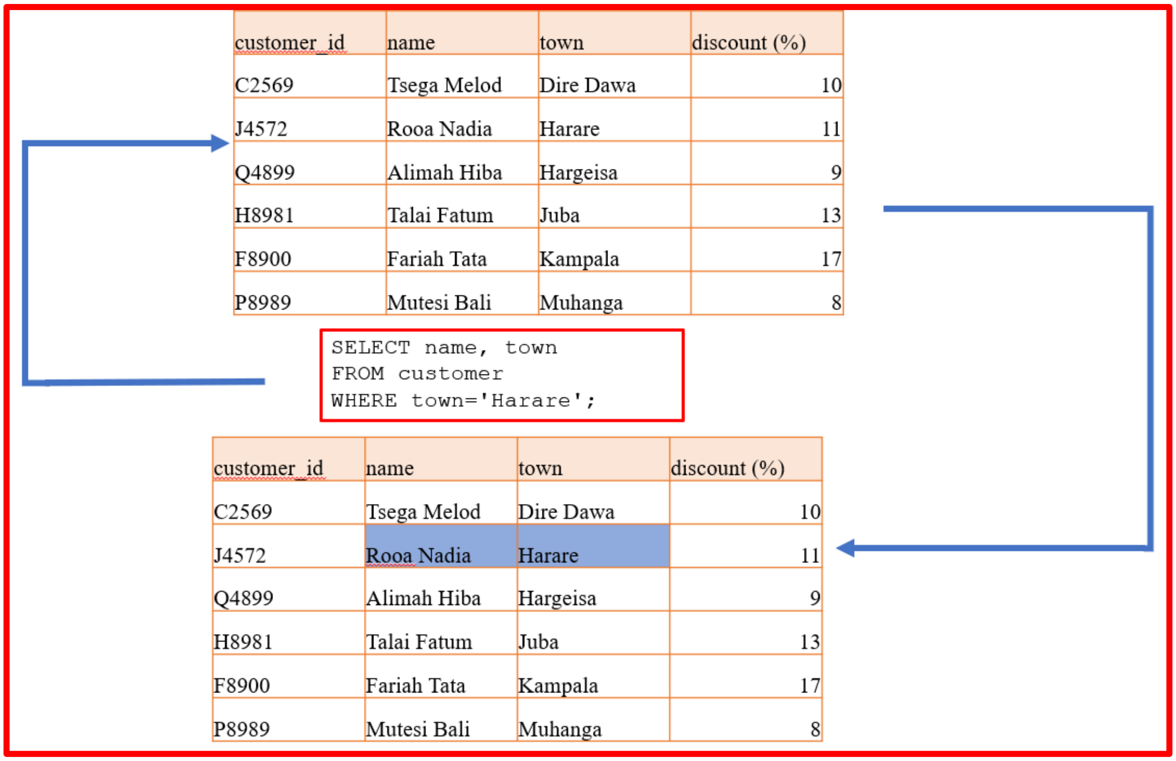
Example 3: Write a SQL statement to display names and town of Customer, who belongs to the town of Harare.

Answer:

```
SELECT name, town
FROM customer
WHERE town='Harare';
```

Output of the Query:

name	town
Rooa Nadia	Harare



customer_id	name	town	discount (%)
C2569	Tsega Melod	Dire Dawa	10
J4572	Rooa Nadia	Harare	11
Q4899	Alimah Hiba	Hargeisa	9
H8981	Talai Fatum	Juba	13
F8900	Fariah Tata	Kampala	17
P8989	Mutesi Bali	Muhanga	8

```
SELECT name, town
FROM customer
WHERE town='Harare';
```

customer_id	name	town	discount (%)
C2569	Tsega Melod	Dire Dawa	10
J4572	Rooa Nadia	Harare	11
Q4899	Alimah Hiba	Hargeisa	9
H8981	Talai Fatum	Juba	13
F8900	Fariah Tata	Kampala	17
P8989	Mutesi Bali	Muhanga	8

Figure 32 Pictorial Representation of Example 3

Example 3:

```
CREATE TABLE Persons (
    ID int NOT NULL,
    LastName varchar(255) NOT NULL,
    FirstName varchar(255),
    Age int,
    );
```

Answer:

Table 11 Persons

```
      id |  lastname   |  firstname |     age
-------------+--------------+--------------+------------
```

```
          6067 | Tsega       |   Melod    |      21
          7889 | Nadia       |   Rooa     |      19
          1010 | Alimah      |   Hiba     |      19
          0987 | Fatum       |   Talai    |      24
          5793 | Tata        |   Fariah   |      25
          2200 | Bali        |   Mutesi   |      27
```

Example 4: SELECT Persons Id, lastname, firstname, age

FROM persons

WHERE (age BETWEEN 19 AND 24 OR id = 7889);

Answer

Table 12 Persons

```
           id  |   lastname   |  firstname  |     age
      -------------+--------------+--------------+------------
          7889 | Nadia       |   Rooa     |      19
          1010 | Alimah      |   Hiba     |      19
          6067 | Tsega       |   Melod    |      21
          0987 | Fatum       |   Talai    |      24
```

Example 5: UPDATE Persons

SET Name = “Sakinat Folorunso”

WHERE Id = 2200;

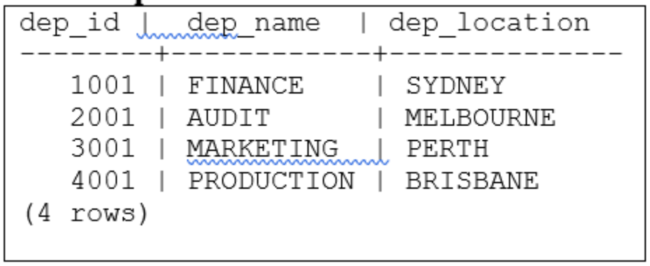
Answer

Table 13 Persons

```
          id |   lastname   |  firstname |     age
-------------+--------------+------------+------------
        6067 | Tsega        |   Tsega    |     21
        7889 | Nadia        |   Nadia    |     19
        1010 | Alimah       |    Hiba    |     19
        0987 | Fatum        |   Talai    |     24
        5793 | Tata         |  Fariah    |     25
        2200 | Sakinat      |  Folorunso |     27
```

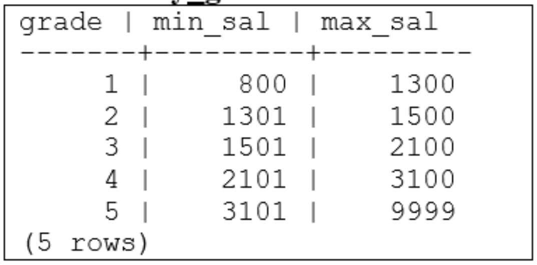
Peer to Peer Interaction

Table Department



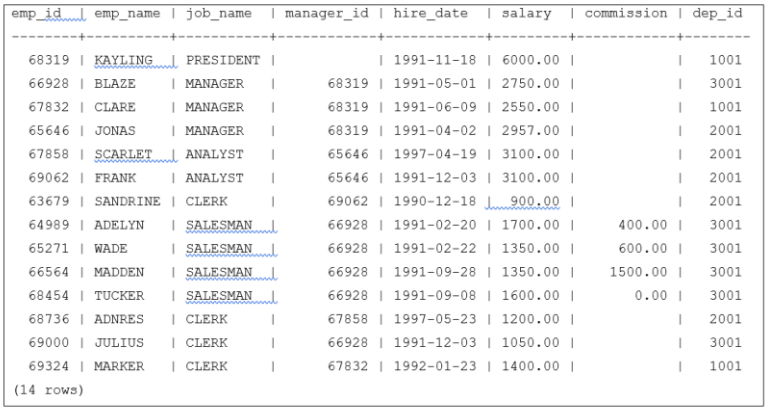
```
dep_id |  dep_name   | dep_location
--------+------------+--------------
   1001 | FINANCE    | SYDNEY
   2001 | AUDIT      | MELBOURNE
   3001 | MARKETING  | PERTH
   4001 | PRODUCTION | BRISBANE
(4 rows)
```

Table Salary_grade



```
grade | min_sal | max_sal
-------+---------+---------
     1 |     800 |    1300
     2 |    1301 |    1500
     3 |    1501 |    2100
     4 |    2101 |    3100
     5 |    3101 |    9999
(5 rows)
```

Table Employee



```
emp_id  | emp_name | job_name  | manager_id | hire_date  | salary  | commission | dep_id
--------+----------+-----------+------------+------------+---------+------------+--------
  68319 | KAYLING  | PRESIDENT |            | 1991-11-18 | 6000.00 |            |   1001
  66928 | BLAZE    | MANAGER   |      68319 | 1991-05-01 | 2750.00 |            |   3001
  67832 | CLARE    | MANAGER   |      68319 | 1991-06-09 | 2550.00 |            |   1001
  65646 | JONAS    | MANAGER   |      68319 | 1991-04-02 | 2957.00 |            |   2001
  67858 | SCARLET  | ANALYST   |      65646 | 1997-04-19 | 3100.00 |            |   2001
  69062 | FRANK    | ANALYST   |      65646 | 1991-12-03 | 3100.00 |            |   2001
  63679 | SANDRINE | CLERK     |      69062 | 1990-12-18 |  900.00 |            |   2001
  64989 | ADELYN   | SALESMAN  |      66928 | 1991-02-20 | 1700.00 |     400.00 |   3001
  65271 | WADE     | SALESMAN  |      66928 | 1991-02-22 | 1350.00 |     600.00 |   3001
  66564 | MADDEN   | SALESMAN  |      66928 | 1991-09-28 | 1350.00 |    1500.00 |   3001
  68454 | TUCKER   | SALESMAN  |      66928 | 1991-09-08 | 1600.00 |       0.00 |   3001
  68736 | ADNRES   | CLERK     |      67858 | 1997-05-23 | 1200.00 |            |   2001
  69000 | JULIUS   | CLERK     |      66928 | 1991-12-03 | 1050.00 |            |   3001
  69324 | MARKER   | CLERK     |      67832 | 1992-01-23 | 1400.00 |            |   1001
(14 rows)
```

1. From Table Employee, write a query in SQL to display all the details of managers

2. Write a query in SQL to list all the employees of grade 2 and 3.

3. Using crow's feet notation, draw the three relational tables Employee, department and salary_grade and their relationships

8.3.4 Data Warehouse

A data warehouse is a system that build-up data from different sources into a single, central, consistent data store to support data analysis, data mining, artificial intelligence (AI), and machine learning. A data warehouse system enables an organization to run powerful analytics on huge volumes (petabytes and petabytes) of historical data in ways that a standard database cannot.

Data warehousing systems have been a part of business intelligence (BI) solutions for over three decades, but they have evolved recently with the emergence of new data types and data hosting methods.

The data warehouse architecture shown by Figure 33 is 3-tiered consisting of:

a. **Bottom tier:** The bottom tier consists of a data warehouse server, usually a relational database system, which collects, cleanses, and transforms data from multiple data sources through a process known as Extract, Transform, and Load (ETL).

b. **Middle tier:** The middle tier consists of an Online Analytical Processing (OLAP) server which enables fast query speeds.

c. **Top tier:** The top tier is representing the front-end user interface or reporting tool, which enables end users to conduct ad-hoc data analysis on their business data.

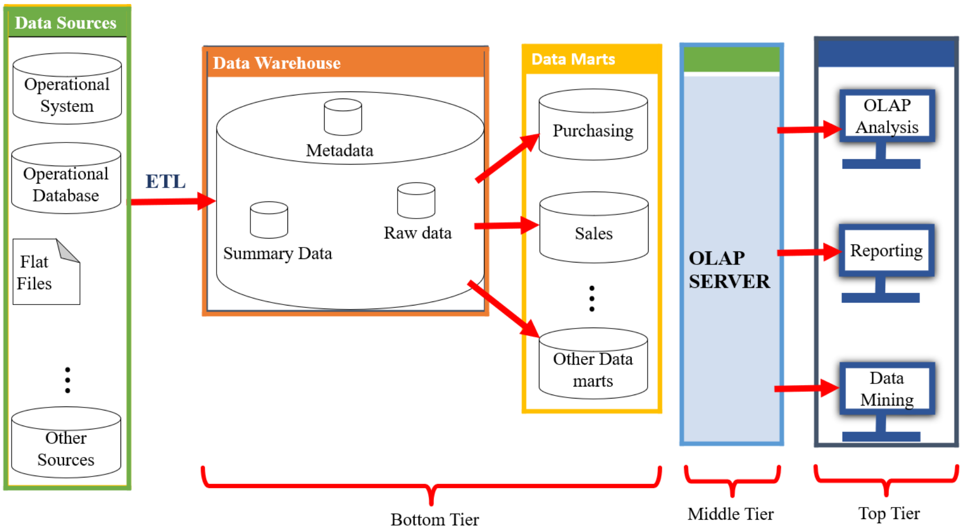


Figure 33 Data Warehouse Architecture

A data mart is a subset of a data warehouse that contains data specific to a particular business line or department. Because they contain a smaller subset of data, data marts enable a department or business line to discover more-focused insights more quickly than possible when working with the broader data warehouse data set.

OLAP is software for performing multidimensional analysis at high speeds on large volumes of data from unified, centralized data store, like a data warehouse. OLAP tools are designed for multidimensional analysis of data in a data warehouse, which contains both historical and transactional data. Common uses of OLAP include data mining and other business intelligence applications, complex analytical calculations, and predictive scenarios, as well as business reporting functions like financial analysis, budgeting, and forecast planning.

Schemas in data warehouses

schema refers to the structure or organization of a database. It comprises the logical description of the whole database, which contains the names and descriptions of tables, records, views, and indexes. The two major types of schema structures are the star and snowflake schema.

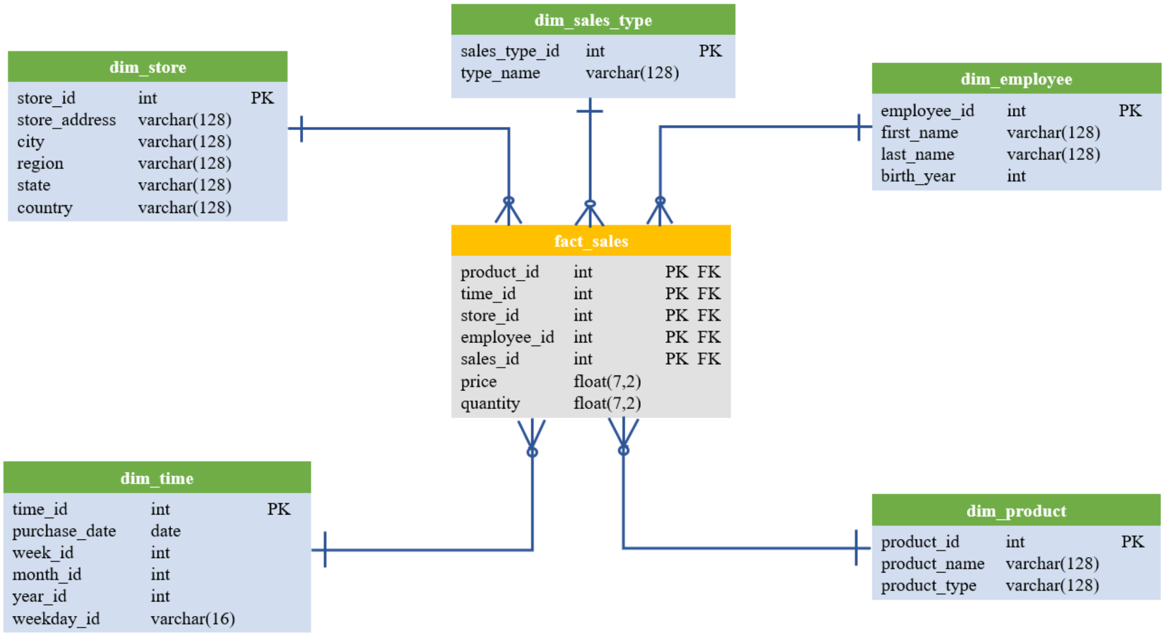
Dimension Table (dim_)

A dimension is similar to relational table and it contains attributes, which are grouped in the form of a dimension. The attribute's data type and key type are also included. It stores the descriptive details of each business process; the what, who, where, and when. They are basically a collection of information that can be referenced to answer meaningful business questions when used together with fact tables. For example, in Figure 34 instance, the dim_store dimension table here contains store id and its location while dim_employe contains employee id, name and birth year. When these descriptive attributes are used with the fact_sales table, a business can find out the quantity of a specific product sold over a defined period, or profits generated from a specific product. Joins between multiple fact and dimension tables are automatically performed to answer such business questions, and because dimension tables are generally denormalized, the number of joins needed to answer business queries is reduced.

Fact Table

Multiple dimension tables are linked to one fact table. It stores transaction or event data in a numeric format. The fact table contains keys and measures from the corresponding dimension table as shown in Table 34. A key is a unique identification of a table. So, in this case PK means Primary Key while FK means Foreign Key. Keys are used to perform joins with dimension tables to run queries. Measures refer to numeric data like price and quantity, which represents business events or transactions, used to add detail to dimension data, so that effective reports can be generated. Information in fact measures like price and quantity is useful, but on its own, this data doesn't give any context to the business to analyze sales.

Star schema: The star schema categorizes business data into facts and dimensions to optimize the structure for reporting. It contains one fact table which can be joined to a number of denormalized dimension tables.



dim_sales_type

sales_type_id	int	PK
type_name	varchar(128)	

dim_store

store_id	int	PK
store_address	varchar(128)	
city	varchar(128)	
region	varchar(128)	
state	varchar(128)	
country	varchar(128)	

dim_employee

employee_id	int	PK
first_name	varchar(128)	
last_name	varchar(128)	
birth_year	int	

fact_sales

product_id	int	PK FK
time_id	int	PK FK
store_id	int	PK FK
employee_id	int	PK FK
sales_id	int	PK FK
price	float(7,2)	
quantity	float(7,2)	

dim_time

time_id	int	PK
purchase_date	date	
week_id	int	
month_id	int	
year_id	int	
weekday_id	varchar(16)	

dim_product

product_id	int	PK
product_name	varchar(128)	
product_type	varchar(128)	

Figure 34 Star Schema

Snowflake schema: In the snowflake schema which is an extension of star schema, dimensions are stored in multiple dimension tables instead of a single table per dimension. The multiple tables associated with a particular dimension branch out further into child tables. An example is dim_store dimension in Figure 35. This “branching out” results in a diagram that resembles a snowflake, thus the name. The users of a snowflake schema benefit from its low levels of data redundancy, but it comes at a cost to query performance.

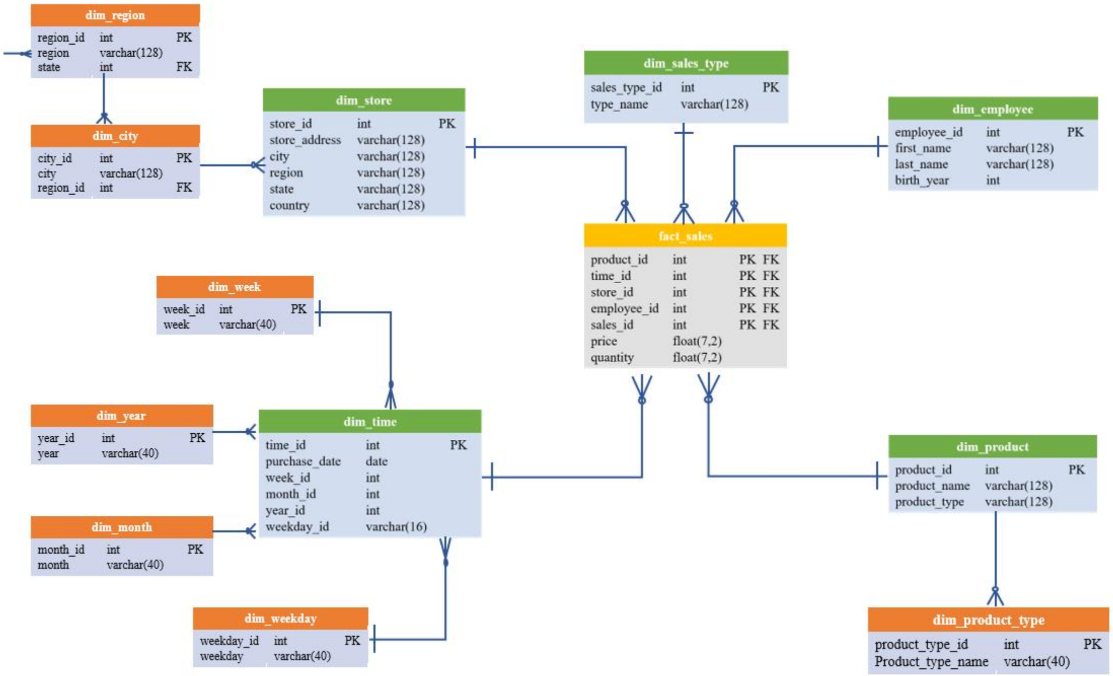


Figure 35 Snowflake Schema

Benefits of a data warehouse

A data warehouse provides a foundation for the following:

a. **Better data quality:** A data warehouse centralizes data from a variety of data sources, such as transactional systems, operational databases, and flat files. It then cleanses it, eliminates duplicates, and standardizes it to create a single source of the truth.

b. **Faster, business insights:** Data from disparate sources limit the ability of decision makers to set business strategies with confidence. Data warehouses enable data integration, allowing business users to leverage all of a company’s data into each business decision.

c. **Smarter decision-making:** A data warehouse supports large-scale BI functions such as data mining (finding unseen patterns and relationships in data), artificial intelligence, and machine learning—tools data professionals and business leaders can use to get hard evidence for making smarter decisions in virtually every area of the organization, from business processes to financial management and inventory management

d. **Gaining and growing competitive advantage:** All of the above combine to help an organization finding more opportunities in data, more quickly than is possible from disparate data stores.

 Peer to Peer Interaction	Prepare a star schema for these relations a. Student b. Department c. Instructor d. university

8.3.5 Online Analytical Processing (OLAP)

OLAP is the technology behind many Business Intelligence (BI) applications. OLAP is a powerful technology for data discovery, including capabilities for limitless report viewing, complex analytical calculations, and predictive "what if" scenario (budget, forecast) planning. It performs multidimensional analysis of business data and provides the capability for complex calculations, trend analysis, and sophisticated data modeling. It is the foundation for many kinds of business applications for Business Performance Management, Planning, Budgeting, Forecasting, Financial Reporting, Analysis, Simulation Models, Knowledge Discovery, and Data Warehouse Reporting. OLAP enables end-users to perform ad hoc analysis of data in multiple dimensions, thereby providing the insight and understanding they need for better decision making. A sample OLAP cube is presented by Figure 37.

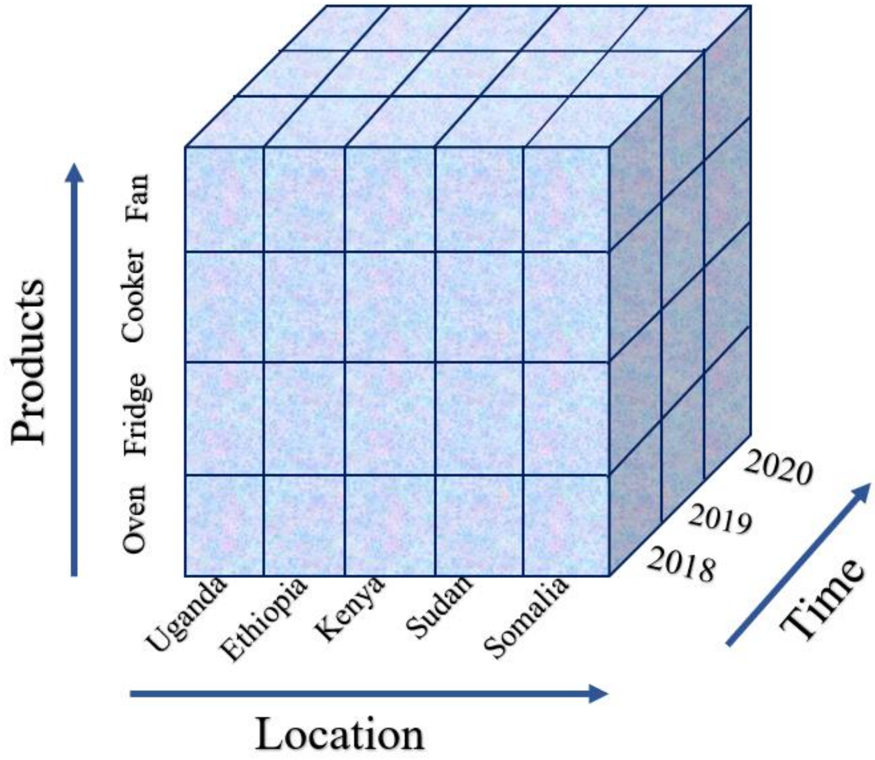


Figure 37 OLAP cube

OLAP for Multidimensional Analysis

OLAP is a group of software that permits users to analyze information from multiple database systems simultaneously. It is a technology that enables analysts to extract and view business data from different points of view. Business is a multidimensional activity and businesses are run on decisions based on multiple dimensions. Businesses track their activities by considering many attributes. When these attributes are tracked on a spreadsheet, they are set on axes (x and y) where each axis represents a logical grouping of attributes in a category. The process grouping, aggregating and joining data is regularly done by Analysts. With OLAP data can be pre-calculated and pre-aggregated, making analysis faster. OLAP databases are divided into one or more **cubes**. The cubes are designed in such a way that creating and viewing reports become easy. The OLAP cube is a data structure optimized for very quick data analysis. The OLAP Cube consists of numeric facts called measures which are categorized by dimensions. OLAP Cube is also called the **hypercube**.

Basic Operations of OLAP

There are basically five kinds of operations in OLAP:

i. Roll-up

The Roll-up operation is also known as consolidation or aggregation. This type of operation is performed by reducing dimensions and mounting up concept hierarchy. Concept hierarchy is a system of grouping items based on their order or level and identical to taxonomy. For example in Figure 38, cities Kampala, Namsana, Kira, Mbarara and Mukono are rolled up into country Uganda. The sales figure of these cities (Kampala, Namsana, Kira, Mbarara and Mukono) are 200, 110, 210, 90, 110 respectively. They became 720 after roll-up. In this aggregation process, data is location hierarchy moves up from city to the country. In the roll-up process at least one or more dimensions need to be removed. In this example, time dimension is removed.

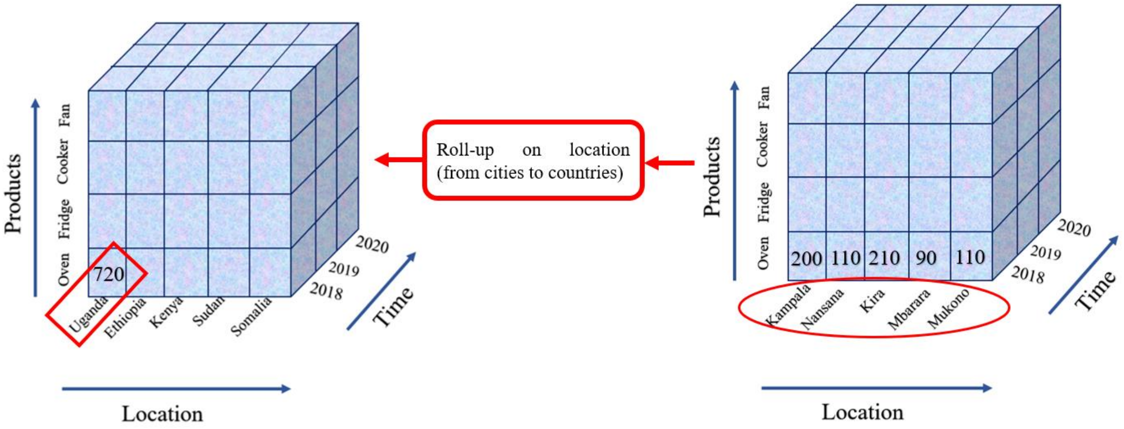


Figure 38 Roll-up process

ii. Drill-down

In drill-down operation, the drilled-down data is split into smaller parts. This can be done by moving down the concept hierarchy and increasing a dimension. For Figure ?, Year ‘2019’ is drilled down to months January-May-September. Corresponding sales are also registers. For Figure 39, dimension months are also added.

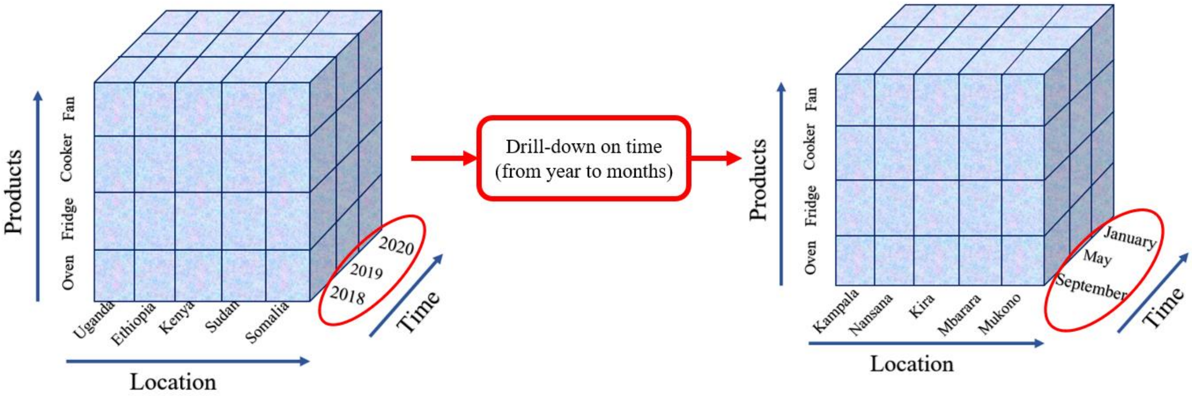


Figure 39 Drill-down process

iii. Slice

For the slice operation, a single dimension is selected, and a new sub-cube is created. For Figure 40, dimension Time is Sliced with ‘2019 as the filter and a new cube is created altogether.

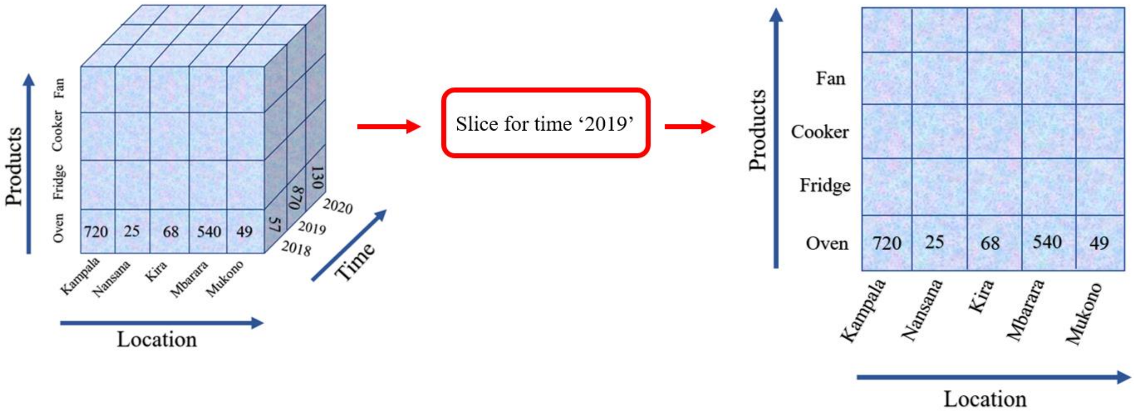


Figure 40 Slice process

iv. Dice

The dice operation is identical to the slice operation. The difference in dice is you select 2 or more dimensions that result in the creation of a sub-cube as shown by Figure 41.

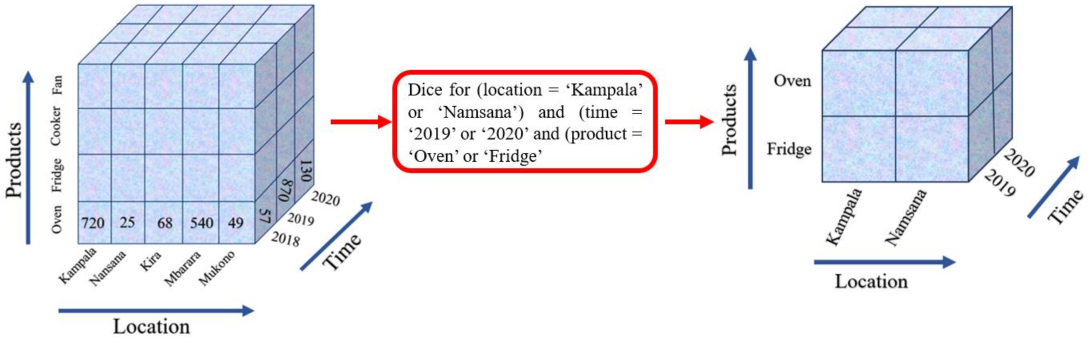


Figure 41 Dice process

v. Pivot (rotate)

The pivot method rotates the data axes to provide a substitute presentation of data. Figure 42 shows a pivot based on product types.

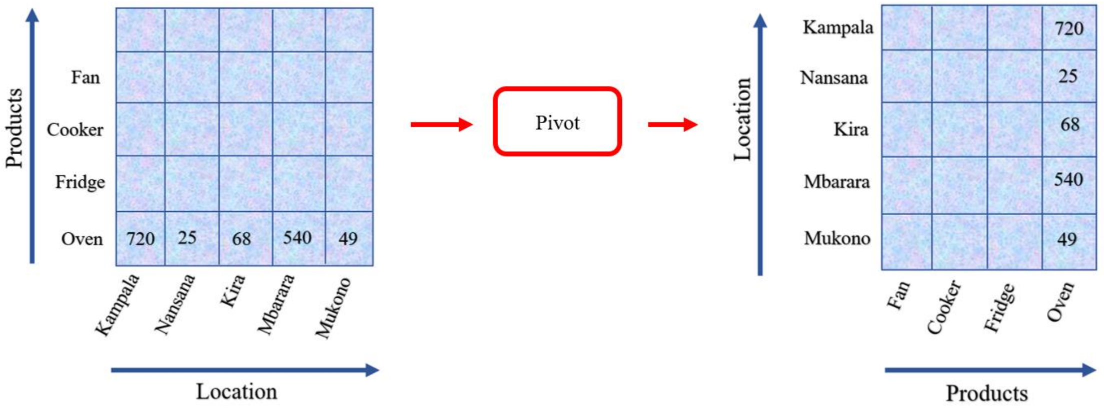


Figure 42 Pivot process

8.4 Data Mining

Data Mining is a process of finding potentially useful patterns from huge data sets. It is a multi-disciplinary skill that uses machine learning, statistics, and AI to extract information to evaluate future events probability. The insights derived from Data Mining are used for marketing, fraud detection, scientific discovery, etc. Data Mining is all about discovering hidden, unsuspected, and previously unknown yet valid relationships amongst the data. Data mining is also called Knowledge Discovery in Data (KDD), Knowledge extraction, data/pattern analysis, information.

The typical different stages involved in the Data mining process are explained below and presented by Figure 43

1. **Identification of the business goals:** The goal of the business is being identified at this stage. Goals like the type of business problem being solved, Customer Acquisition? Retention? Reduce maintenance costs or operational costs?
2. **Identification of data:** This is the stage at which the required data to achieve the business goal is identified and collected in appropriate manner. The dataset is also described based on the data domain, number of attributes, the attribute data type, the number of observations, period of collection, missing values, number of labels etc.
3. **Data Preparation**: We also call this stage the pre-processing stage. Here, dataset is cleaned and give a uniform formatting to all data items. Some of the common tasks at this stage is treating missing values, class imbalance, data transformation etc. The dataset is put in the appropriate format for the data modelling and analysis by the data mining techniques.
4. **Modelling, Analysis and Evaluation:** Based on the data type of the dataset, an appropriate data mining technique is selected for analysis. The data mining task is majorly divided into two parts: Descriptive, Predictive and Prescriptive analysis. Also, based on the type of analytics done, the evaluation metrics vary from accuracy, sensitivity, specificity, F1-Score, to RMSE, MSE, MAE etc.
5. **Knowledge discovery:** This is the last stage where result obtained are interpreted for decision making. The hidden pattern and useful insights are revealed at this stage.

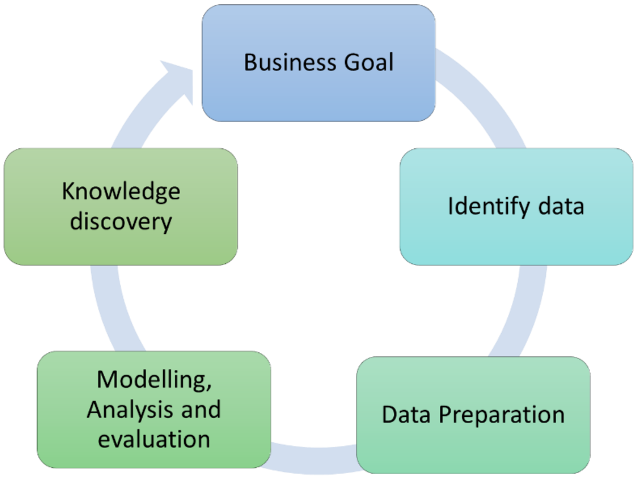


Figure 43 The Data mining process flow

8.4.1 Data Analysis

Data analysis can be divided into descriptive, prescriptive and predictive analytics as shown by Figure 44 Descriptive analytics aims to help uncover valuable insight from the data being analyzed. Prescriptive analytics suggests conclusions or actions that may be taken based on the analysis. Predictive analytics focuses on the application of statistical models to help forecast the behavior of people and markets.

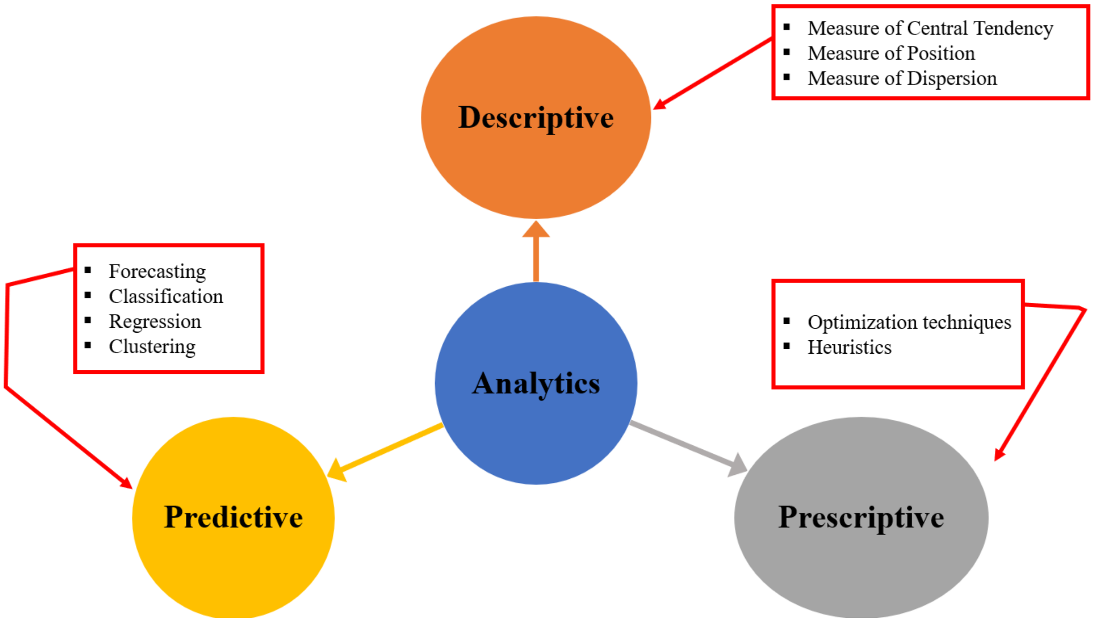


Figure 44 Types of data mining analytics

Descriptive Analytics

Descriptive analytics is a field of statistics that focuses on gathering and summarizing raw data to be easily interpreted. Generally, descriptive analytics concentrate on historical data, providing the context that is vital for understanding information and numbers. The field usually serves as a preliminary step in the business intelligence process, creating a foundation for further analysis and understanding. Essentially, descriptive analytics seeks answers about what happened, without performing the more complex analyses required in diagnostics and predictive models. In business intelligence, descriptive analytics is usually the first step, and will result in visualizations like pie charts, line graphs, bar charts, and other simpler graphical displays.

Descriptive Analytics, the conventional form of Business Intelligence and data analysis, seeks to provide a depiction or "summary view" of facts and figures in an understandable format, to either inform or prepare data for further analysis. It uses two primary techniques, namely data aggregation and data mining to report past events. It presents past data in an easily digestible format for the benefit of a wide business audience.

A common example of Descriptive Analytics is company reports that simply provide a historic review of an organization's operations, sales, financials, customers, and stakeholders. Descriptive Analytics helps to describe and present data in a format which can be easily understood by a wide variety of business readers. Figure 45 illustrates the 5 key points of measure of central tendency.

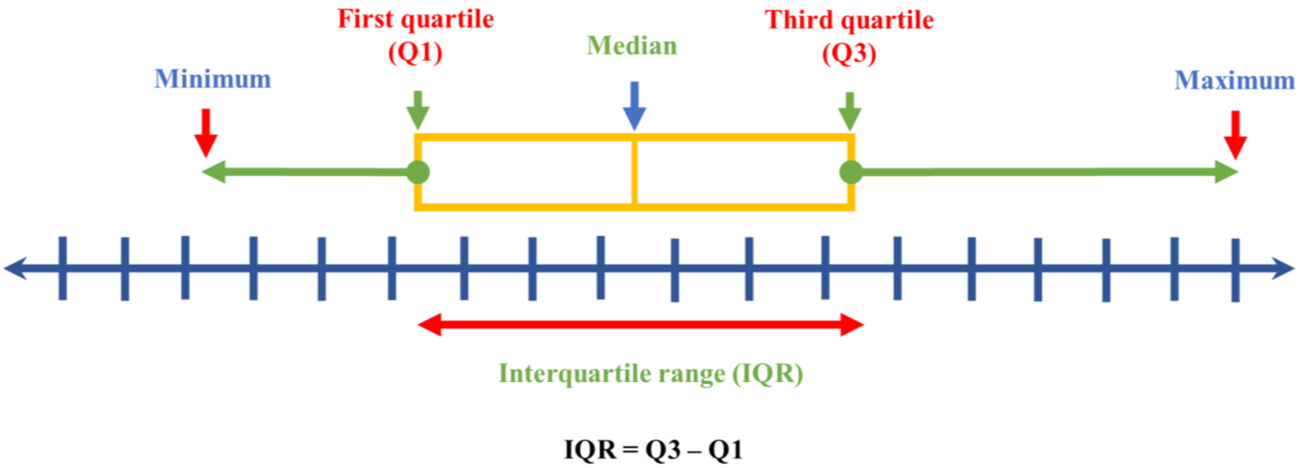


Figure 45 5 key points of central tendency

MEASURE OF CENTRAL TENDENCY

This is a central value that the data values are grouped around.

Mean: This is the sum of values in the dataset divided by the number of values. It is also the average of the values

Median: This is the middle value or the average of the two middle values for a numerically ordered set.

Mode: This is the value (s) that appeared most often. A dataset can one or more modes or no mods at all

Minimum: This is the smallest value in the entire dataset

Maximum: This is the maximum value in the entire dataset

The **quartiles** (Q1 and Q3) are the medians of the lower and upper halves of the dataset.

The **interquartile range** (IQR) is the difference between the 1st and 3rd quartiles (Q3 - Q1). This represents the middle 50% of the data

MEASURE OF DISPERSION

Dispersion is the measure of variation of data items. It shows how the data is spread around the mean as expressed by Figure 46. It measures the extent to which the items vary from central value. Dispersion is also known as average of the second order.

Range: This is the difference between the maximum and minimum observation values

Variance: This measure uses mean as a point of reference

Standard Deviation: This measure is equal to the square root of the variance

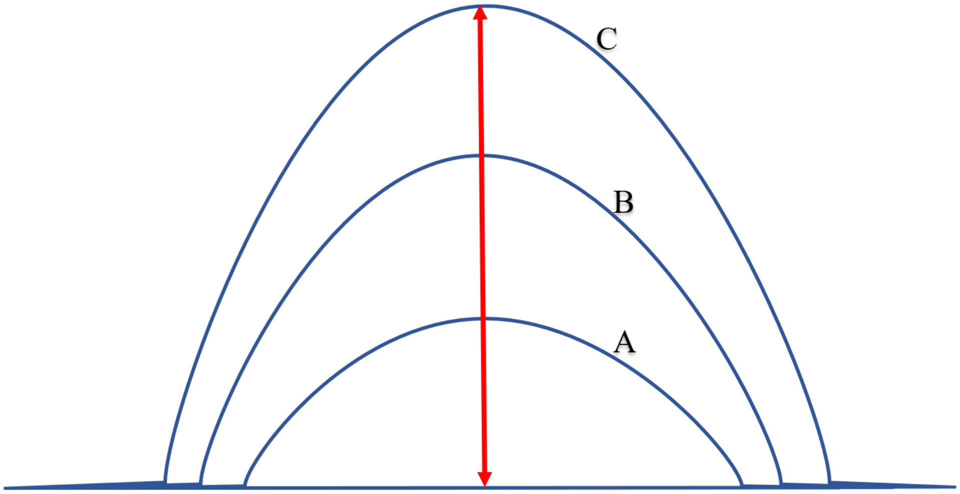


Figure 46 Example of samples from two populations with the same mean but different dispersion. Population A is much more dispersed than the population B and C.

Measures of Position

Measures of position give a range where a certain percentage of the data fall. It is used to describe the location of a particular observation in relation to the rest of the data.

Percentile: The dataset is divided into 100 equal parts (percentile) as shown by Figure 47. The p^{th} percentile of the data set is a measurement such that after the data are ordered from smallest to largest, at most, p% of the data are at or below this value and at most, (100 - p) % at or above it. The median is the value where fifty percent or the data values fall at or below it. Therefore, the median is the 50th percentile.

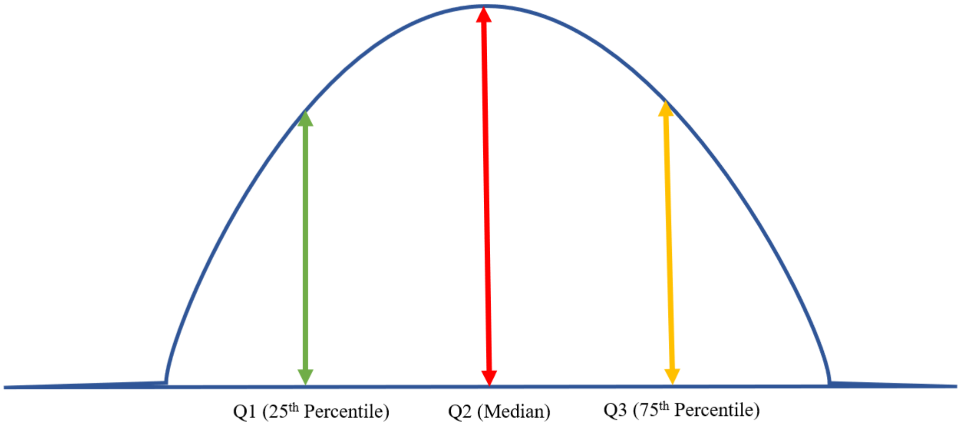


Figure 47 Measure of position

Deciles: The dataset is divided into 10 equal parts (Decile).

Quartile: The dataset is divided into 4 equal parts (Quartile)

All the quartiles and deciles are percentiles. For example, the 50th percentile is the 5th decile and 2nd quartile of a distribution and are all the same and correspond to the median.

Shape of Distribution

Distributions can be symmetric or skewed depending whether there are more frequencies at one end of the distribution than the other

Predictive Analytics

Predictive analytics has its roots in the ability to “Predict” what might happen. These analytics are about understanding the future. Predictive analytics provides companies with actionable insights based on data. Figure 48 presents the data mining techniques employed for the predictive analytics tasks. Predictive analytics provide estimates about the likelihood of a future outcome. It is important to remember that no statistical algorithm can “predict” the future with 100% certainty.

Companies use these statistics to forecast what might happen in the future. This is because the foundation of predictive analytics is based on probabilities.

These statistics try to take the data that you have, and fill in the missing data with best guesses. They combine historical data found in ERP, CRM, HR and POS systems to identify patterns in the data and apply statistical models and algorithms to capture relationships between various data sets. Companies use Predictive statistics and analytics anytime they want to look into the future. Predictive analytics can be used throughout the organization, from forecasting customer behavior and purchasing patterns to identifying trends in sales activities. They also help forecast demand for inputs from the supply chain, operations and inventory.

One common application most people are familiar with is the use of predictive analytics to produce a credit score. These scores are used by financial services to determine the probability of customers making future credit payments on time. Typical business uses include, understanding how sales might close at the end of the year, predicting what items customers will purchase together, or forecasting inventory levels based upon a myriad of variables.

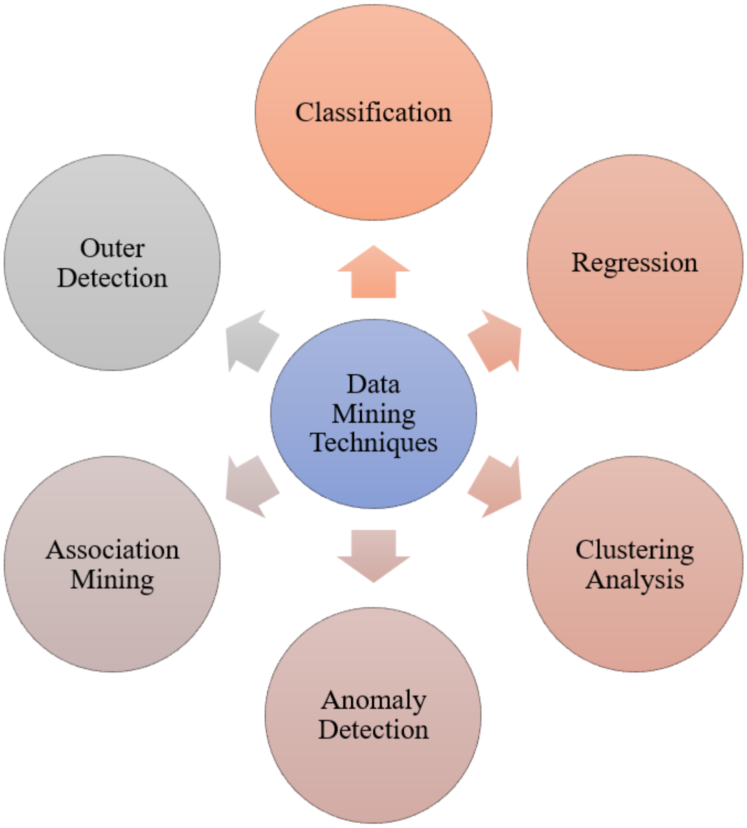


Figure 48 Data Mining Techniques

a. **Classification:** This analysis is used to retrieve important and relevant information about data, and metadata. This data mining method helps to classify data in different classes.

b. **Clustering:** Clustering analysis is a data mining technique to identify data that are like each other. This process helps to understand the differences and similarities between the data.

c. **Regression:** Regression analysis is the data mining method of identifying and analyzing the relationship between variables. It is used to identify the likelihood of a specific variable, given the presence of other variables.

d. **Association Mining:** This data mining technique helps to find the association between two or more Items. It discovers a hidden pattern in the data set.

e. **Outer detection:** This type of data mining technique refers to observation of data items in the dataset which do not match an expected pattern or expected behavior. This technique can be used in a variety of domains, such as intrusion, detection, fraud or fault detection, etc. Outer detection is also called Outlier Analysis or Outlier mining.

f. **Prediction:** Prediction has used a combination of the other techniques of data mining like trends, sequential patterns, clustering, classification, etc. It analyzes past events or instances in a right sequence for predicting a future event.

Prescriptive Analytics: Advise on possible outcomes

Prescriptive Analytics, as the name implies, is about obtaining a “prescription” on how to solve a specific problem. Here are a few typical examples: “How should we source products during high season to keep up with demand and minimize total supply chain cost?” “What is the optimal safety stock level to guarantee service level objectives and minimize stock investment?” and “How can manufacturing orders on the bottleneck be sequenced to maximize efficiency and minimize set-up time? Optimization, heuristics and machine learning methods can be applied to these types of issues depending on the industry and the specific challenge. In a nut-shell, these analytics are all about providing advice. Prescriptive analytics attempt to quantify the effect of future decisions in order to advise on possible outcomes before the decisions are actually made. At their best, prescriptive analytics predicts not only what will happen, but also why it will happen providing recommendations regarding actions that will take advantage of the predictions.

These analytics go beyond descriptive and predictive analytics by recommending one or more possible courses of action. Essentially they predict multiple futures and allow companies to assess a number of possible outcomes based upon their actions. Prescriptive analytics use a combination of techniques and tools such as business rules, algorithms, machine learning and computational modelling procedures. These techniques are applied against input from many different data sets including historical and transactional data, real-time data feeds, and big data.

At the core of prescriptive analytics is the idea of optimization, which means every little factor has to be taken into account when building a prescriptive model. Supply chain, labor costs, scheduling of workers, energy costs, potential machine failure--everything that could possibly be a factor is included in making a prescriptive model. Some common mathematical models used for prescriptive statistics are Linear Programming, Integer Programming, Non-Linear Programming and Mixed Integer Programming

8.4.2 Reporting

BI reporting, is the process of utilizing BI software to collect, visualize, and analyze business data for the purpose of finding relevant and actionable insights into business trends. The aim is to provide suggestions and observations about business trends, empowering decision-makers to act.

Benefits of BI Reporting Tools

i. **Faster Reporting and Analysis:** Modern BI reporting tools give all business users the power to create dynamic and detailed reports off of accurate, real-time data.

ii. **Make Better Business Decisions:** Visually robust reports can also help streamline decision-making across departments by allowing sharable, easy-to-understand reports to support projects, ideas, and initiatives.

iii. **Increased Productivity (and Savings):**

iv. Through report automation, IT can easily deliver the reports users need. With no reporting requests bogging down the IT department, organizations can focus more time and resources on the core application. And increased data democratization means more users can discover (and act on) insights that lead to new revenue streams and cost reductions.

Merits of Reporting

The main goal of BI reports is to deliver comprehensive data that can be easily accessed, interpreted, and provide actionable insights. These merits are shown by Figure 49

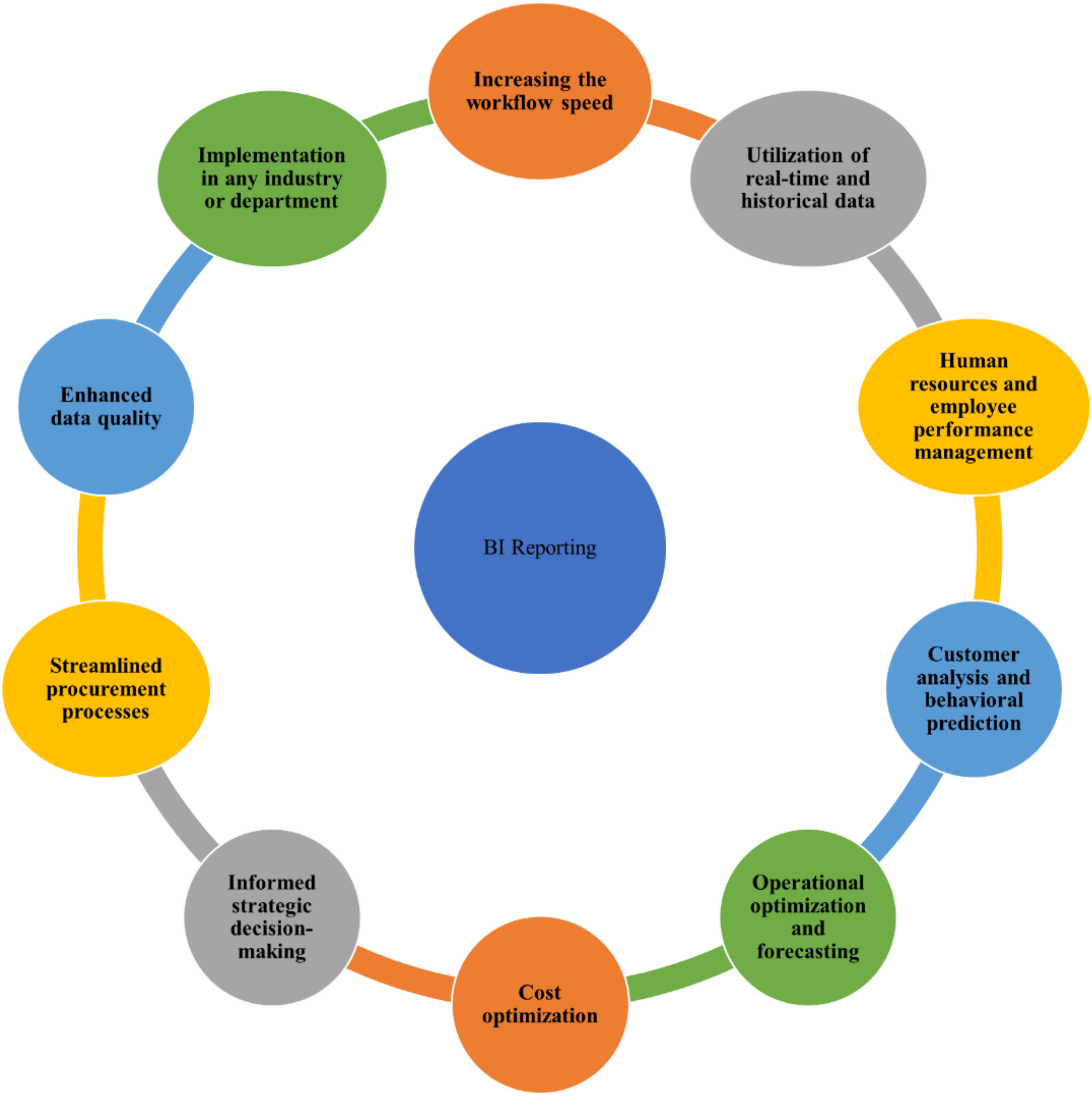

Increasing the workflow speed
Utilization of real-time and historical data
Human resources and employee performance management
Customer analysis and behavioral prediction
Operational optimization and forecasting
Cost optimization
Informed strategic decision-making
Streamlined procurement processes
Enhanced data quality
Implementation in any industry or department
BI Reporting

Figure 49: Merits of BI Reporting

To summarize, here are the top benefits of business intelligence reporting:

1. Increasing the workflow speed
2. Implementation in any industry or department
3. Utilization of real-time and historical data
4. Customer analysis and behavioral prediction
5. Operational optimization and forecasting
6. Cost optimization
7. Informed strategic decision-making
8. Streamlined procurement processes
9. Enhanced data quality
10. Human resources and employee performance management

8.5 Data Visualization

Data Visualization is a method of transforming raw data into graphical or pictographic representations such as charts, graphs, diagrams, pictures, and videos to ease the understanding of the data and gain quick insights into the data. This will allow quick data analysis and report preparation for effective business decisions. Data Visualization is the process of communicating complex information with simple graphics and charts. Data Visualization has the power to tell data-driven stories while allowing people to see patterns and relationships found in data.

Advantages of Visualization

- Clarifies which element influences customer behavior.
- Identifies the area on which you need to pay attention.
- Guides you to understand which product should be placed in which location.
- Predicts the sales volume.

8.5.1 Charting

A chart is a pictorial representation of data, where the data could be represented by symbols like bars in a bar chart, lines in a line chart, or slices in a pie chart. A chart can represent tabular numeric data, functions or some kinds of quality structure and provides different info.

The term "chart" as graphic representation of data has multiple meanings:

a. A data chart is a type of diagram or graph that organizes and represents a set of numerical or qualitative data.
b. Maps that are decorated with extra information (map surround) for a specific purpose are often known as charts, such as a nautical chart or aeronautical chart, typically spread over several map sheets.

Charts are often used to ease understanding of large quantities of data and the relationships between parts of the data. Charts can usually be read more quickly than the raw data. They are used in a wide variety of fields, and can be created by hand (often on graph paper) or by computer using a charting application. Certain types of charts are more useful for presenting a given data set than others. For example, data that presents percentages in different groups (such as "Male, Female") are often displayed in a pie chart, but may be more easily understood when presented in a horizontal bar chart. On the other hand, data that represents numbers that change over a period of time (such as "recovered cases of COVID-19 from 02/21/2020 to 05/12/2020") is best represented as a line chart.

Components of a Chart

A chart can take a large variety of forms, however there are common features that provide the chart with its ability to extract meaning from data.

Typically, the data in a chart is represented graphically, since humans are generally able to understand **pictures** more quickly than text. Text is generally used only to explain the data.

a. **Title:** This is an important aspect of a graph. It uses text in a graph is the **title**. A graph's title usually appears above the main graphic and provides a brief description of the data in the graph.

b. **Dimensions:** This are often displayed on axes. If a horizontal and a vertical axis are used, they are usually referred to as the x-axis and y-axis respectively. Each axis will have a scale, denoted by periodic graduations and usually accompanied by numerical or categorical indications. Each axis will typically also have a label displayed outside or beside it, briefly describing the dimension represented. If the scale is numerical, the label will often be suffixed with the unit of that scale in parentheses.

c. **Grid of Lines**: This appears in the graph to aid in the pictorial alignment of data. The grid can be enhanced by visually highlighting the lines at regular or significant graduations. The emphasized lines are then called major grid lines and the remainders are minor grid lines.

d. **Legend:** This is the key to identifying various attributes involved in the graph. When the data appearing in a chart contains multiple attributes, legend will be included. A legend contains a list of the variables appearing in the chart and an example of their appearance. This information allows the data from each variable to be identified in the chart.

Types of Charts

A data of a chart can appear in different formats, and might contain individual textual labels describing the datum associated with the indicated position in the chart. The data might also appear as dots or shapes, connected or unconnected, and in different combination of colors and patterns. Inferences or points of interest can be overlaid directly on the graph to further aid information extraction. There different types of charts to give insights to your data. They all tell their stories differently. The right chart helps you to boost the impact of your data. The commonest charts used for BI are illustrated by Figure 50

(a) A histogram consists of tabular frequencies, shown as adjacent rectangles, erected over discrete intervals (bins), with an area equal to the frequency of the observations in the intervals.

(b) A bar chart is a chart with rectangular bars with lengths proportional to the values that they represent. The bars can be plotted vertically or horizontally.

(c) A pie chart shows percentage values as a slice of a pie. A pie chart is a circular statistical chart showing the whole data in parts. Each portion of a pie chart represents the proportions, and the sum of all parts should be equal to 100%. The whole data can be divided into slices to show the numerical fractions of each part of the data.

(d) A line graph is a two-dimensional plot of ordered observations where the observations are connected by a straight line in their respective order. Each point in the line corresponds to a data value in the given group. Line charts should only be used to measure the trends over a period of time, e.g. dates, months, and years

(e) Doughnut Chart is identical to pie charts. It shows the whole data into the proportion of the whole data. It is the helpful when displaying various proportions that sum up the final value.

(f) Area Chart shows a line chart (comparing two or more quantities) illustrating quantitative graphical data. The area between the axes and lines are filled with colors, textures, and patterns. It shows the trend changes over time.

(g) Ribbon Chart show which data category has the largest value. Ribbon charts are effective at showing rank change, with the highest range (value) always displayed on top for each time period.

(h) Waterfall chart also identified as a "Walk" chart, is a special type of floating-column chart It is used to show how initial values are increasing and decreasing gradually by a series of values to arrive at the final value.

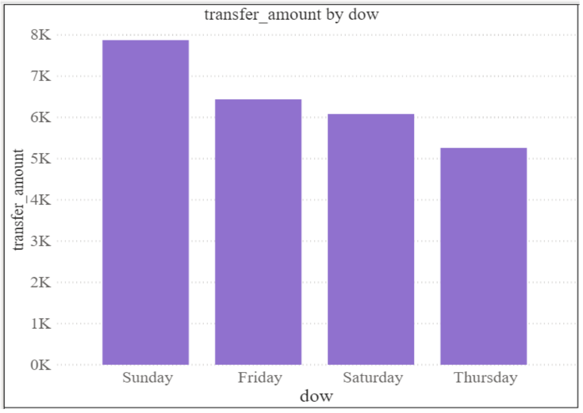
(i) Funnel Chart is a kind of chart used to picture the data that flows from one stage to another. The whole data is considered as 100%, and in each stage, it is represented as numerical proportions of the data.

(j) Scatter Plot charts are used to picture the relationship between the attributes of data using the dots that represent the values obtained from two different attributes on both the y-axis and x-axis. It is identical to a correlation plot because it shows how two attributes are correlated to each other. It is also similar to a bubble chart where a third variable is represented by the size of the points.

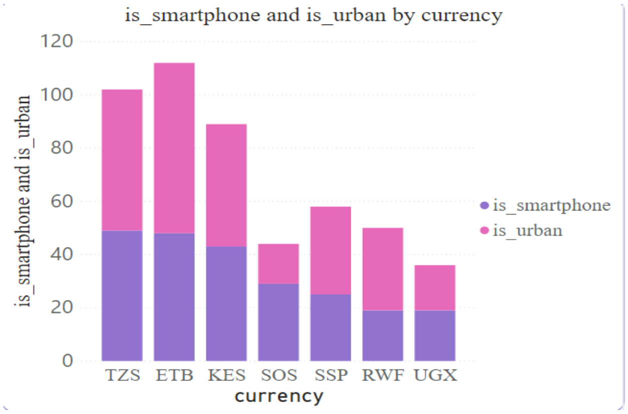
(k) Tree Map is a chart that display the hierarchy of the data values of the dataset in a nestled rectangle. At each level, hierarchy is represented by a color. The size of the space in the rectangle depends on the data values. The rectangular boxes are arranged in size from top left to bottom right.

(l) Guage Chart is also synonymous to a speedometer or dial chart. It uses the needle to read the data, and it shows the data value on the dial. These charts are useful to compare the values between the attributes either by using multiple needles on the same gauge or different gauges.

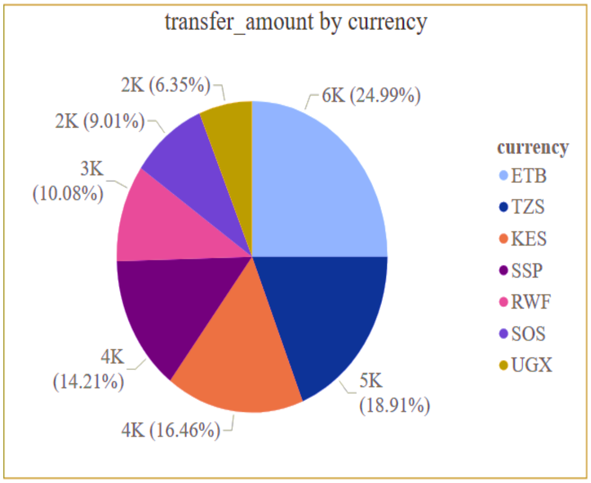
(m) Tree Decomposition is a chart that display the hierarchy of the data attribute values of the dataset in a tree. The desired attribute is the root of the tree. At each level, hierarchy is represented by another attribute. The concept is likened to a taxonomy tree. This type of chart shows the micro relationship between the attributes of the dataset.



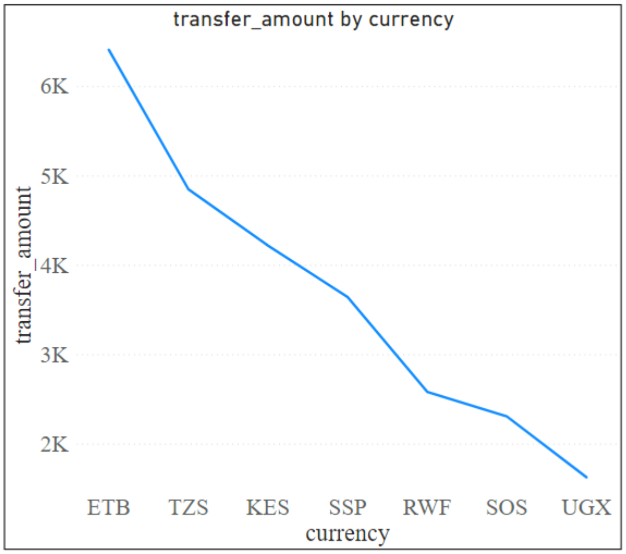
(a) Histogram



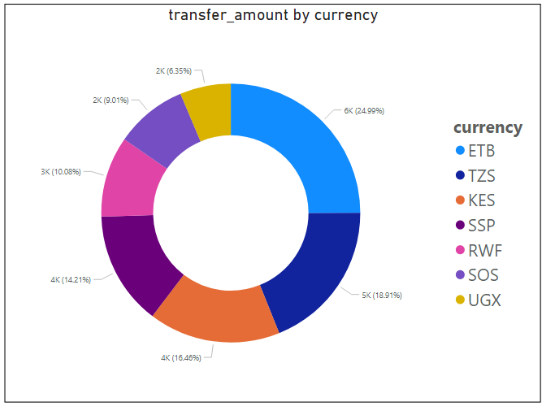
(b) Bar chart

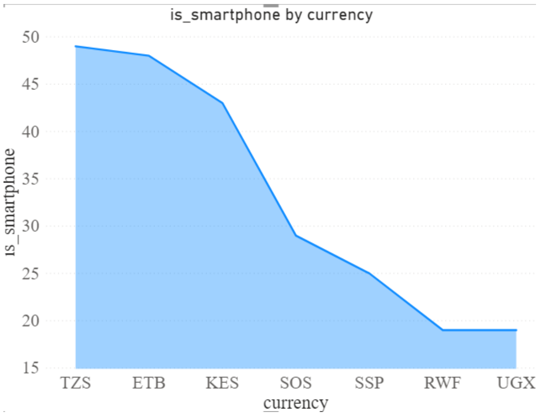


(c) Pie Chart



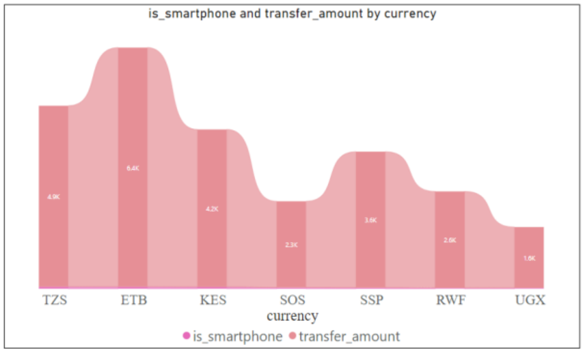
(d) Line Graph

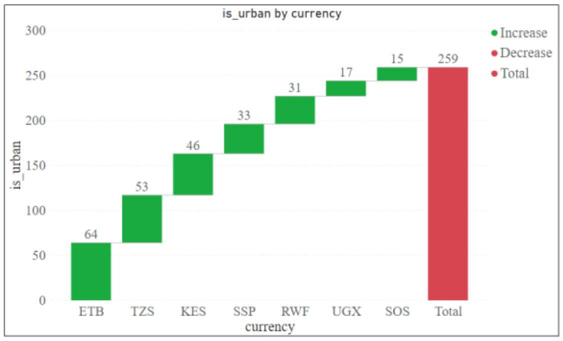




(e) Doughnut Chart

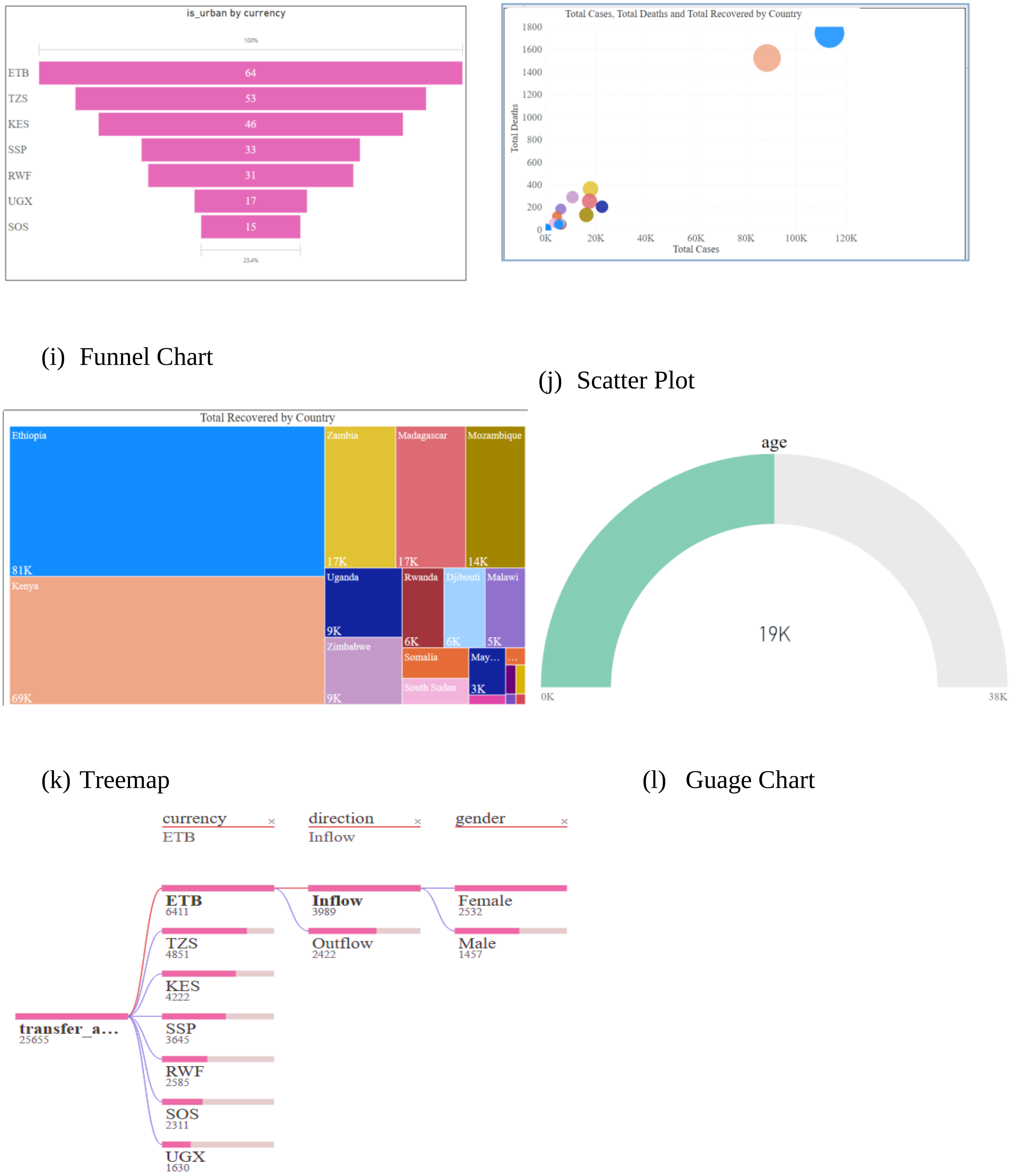
(f) Area Chart





(h) Waterfall chart

(g) Ribbon Chart



(i) Funnel Chart

(j) Scatter Plot

(k) Treemap

(l) Guage Chart

(m) Decomposition Tree

Figure 50 Types of chart

8.5.2 BI Dashboard

Business Intelligence dashboards are information management and data visualization solutions used to analyze your data. Content creators can use interactive elements like filters and actions to combine charts, graphs and reports in a single screen for snapshot overviews. Dashboards are one of the most popular capabilities of BI platforms because they present easily understandable data analysis, allow you to customize which information you want to view, and provide a way to share the results of your analysis with others.

The purpose of a BI dashboard

Dashboards are integral to an organizations' business intelligence strategy. They should be purpose-built and designed to analyze data from key datasets, to improve business decisions. Instead of analysts manually compiling spreadsheets, modern BI platforms can access, analyze, display, and share data via web-based dashboards. With a powerful, automated business intelligence tool, stakeholders can build dashboards to review, draw conclusions, and act. Figure 51 shows a sample dashboard of COVID-19 cases in African countries. It shows the total number of confirmed, recovered and death cases in each country.

Key features of BI dashboards

Modern BI platforms offer a lot of the same key features, with many real-world dashboard examples showcasing some or all of these:

- Customizable interface
- Interactivity
- Ability to pull near real-time data
- Accessible from a web browser
- Standard templates
- Sharing capability to foster collaboration

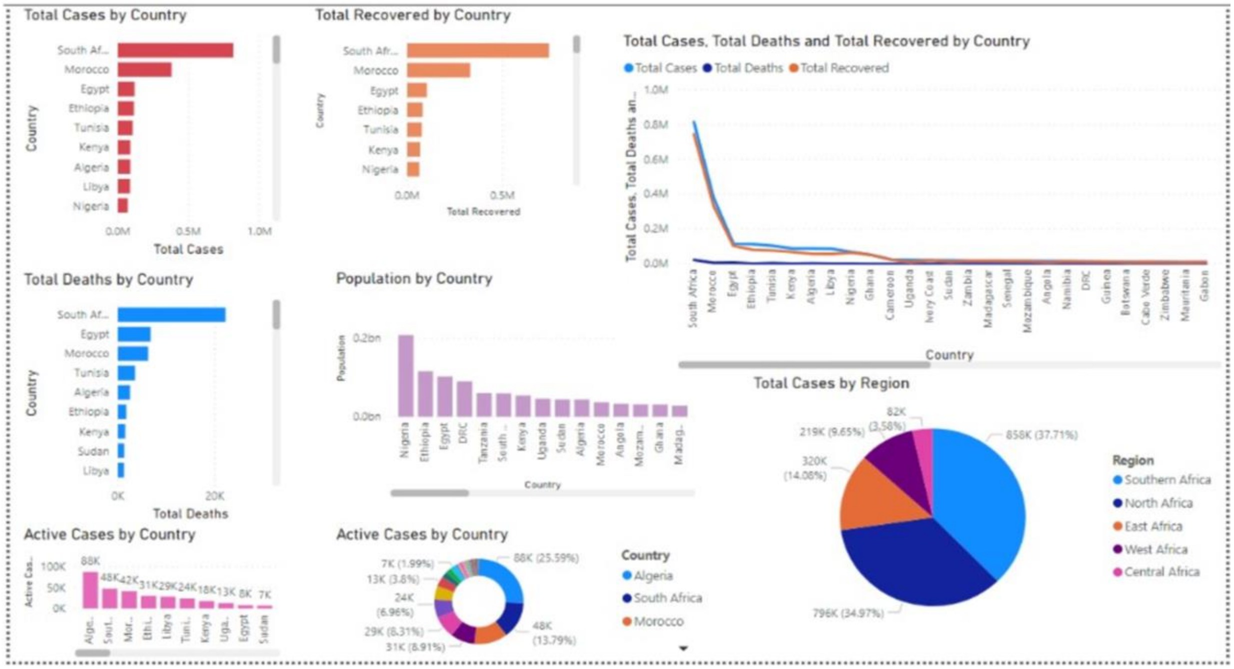


Figure 51 Sample dashboard for COVID-19 cases in Africa

8.6 Power Business Intelligence (BI)

Power BI is a collection of software services, apps, and connectors that work together to transform your different sources of data into rational, pictorial and interactive understandings. Your data may be an in different formats like Excel spreadsheet, Comma Separated V (CSV), Text, or a collection of cloud-based and on-premises hybrid data warehouses. Power BI lets you easily connect to your data sources, visualize and discover what's important, and share that with anyone or everyone you want. All necessary click buttons are either circled red or have a red arrow pointing to them.
Link to video to use and install Power BI https://www.youtube.com/watch?v=Qgam9M8I0xA

With Power BI Desktop, you can:

a. Get data

- The Power BI Desktop makes discovering data easy. You can import data from a wide variety of data sources. After you connect to a data source, you can shape the data to match your analysis and reporting needs.

b. Create relationships and enrich your data model with new measures and data formats

- When you import two or more tables, oftentimes you'll need to create relationships between those tables. The Power BI Desktop includes the Manage Relationships dialog and the Relationships view, where you can use Autodetect to let the Power BI Desktop find and create any relationships, or you can create them yourself. You can also very easily create your own measures and calculations or customize data formats and categories to enrich your data for additional insights.

c. Create reports

- The Power BI Desktop includes the Report View. Select the fields you want, add filters, choose from dozens of visualizations, format your reports with custom colors, gradients and several other options. The Report View gives you the same great report and visualizations tools just like when creating a report on PowerBI.com.

d. Save your reports

- With the Power BI Desktop, you can save your work as a Power BI Desktop file. Power BI Desktop files have a .pbix extension.

e. Upload or Publish your reports

- You can upload the reports you created and saved in the Desktop to your Power BI site. You can also publish them to Power BI right from Power BI Desktop.

8.6.1 Supported Browsers for Power BI

Power BI is designed to work with any of these supported browsers, but performance does differ depending on your choice of browser. Power BI supports these browsers on all platforms where they're available:

- Microsoft Edge
- Internet Explorer 11.
- Chrome desktop latest version
- Safari Mac latest version

- Firefox desktop latest version.
- Power BI doesn't run in any browsers in iOS10 or previous versions.

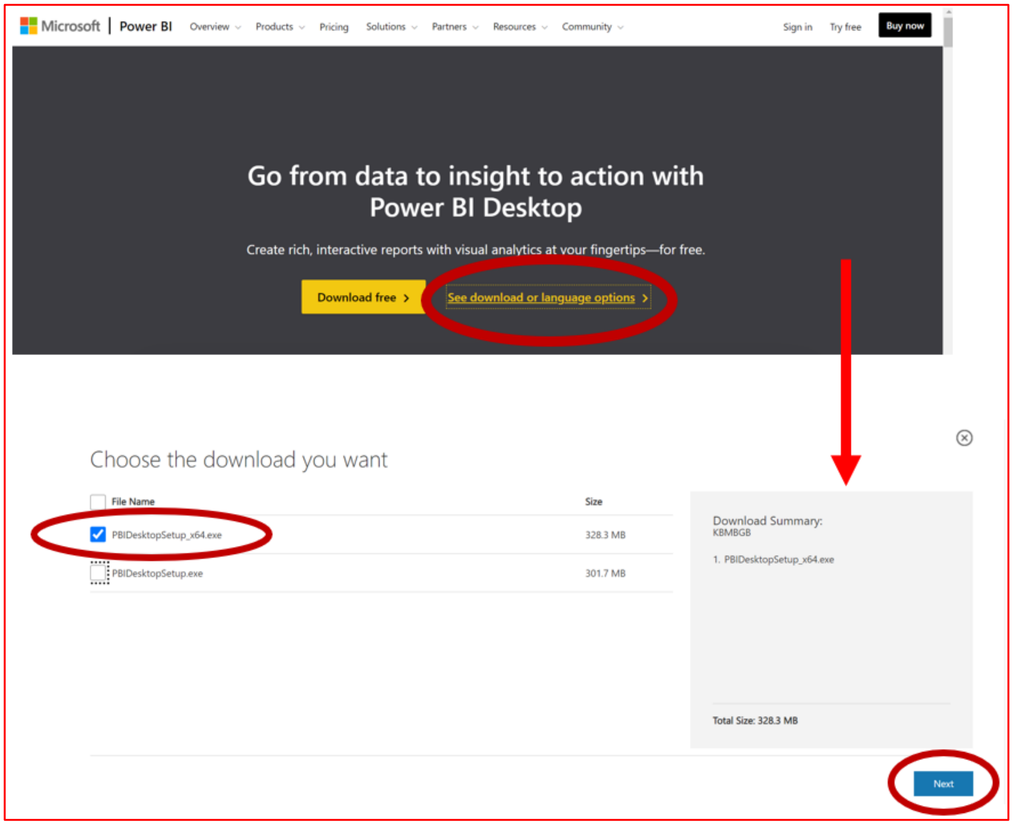
8.6.2 Downloading and Installing Power BI

To get started with Power BI Desktop, the first thing you need is to download and install the application. All necessary click buttons are either circled red or have a red arrow pointing to them.

There are two ways to get Power BI Desktop:

Get Power BI Desktop from the Windows Store: Link

Download Power BI Desktop from the web: Link



Go from data to insight to action with Power BI Desktop

Create rich, interactive reports with visual analytics at your fingertips—for free.

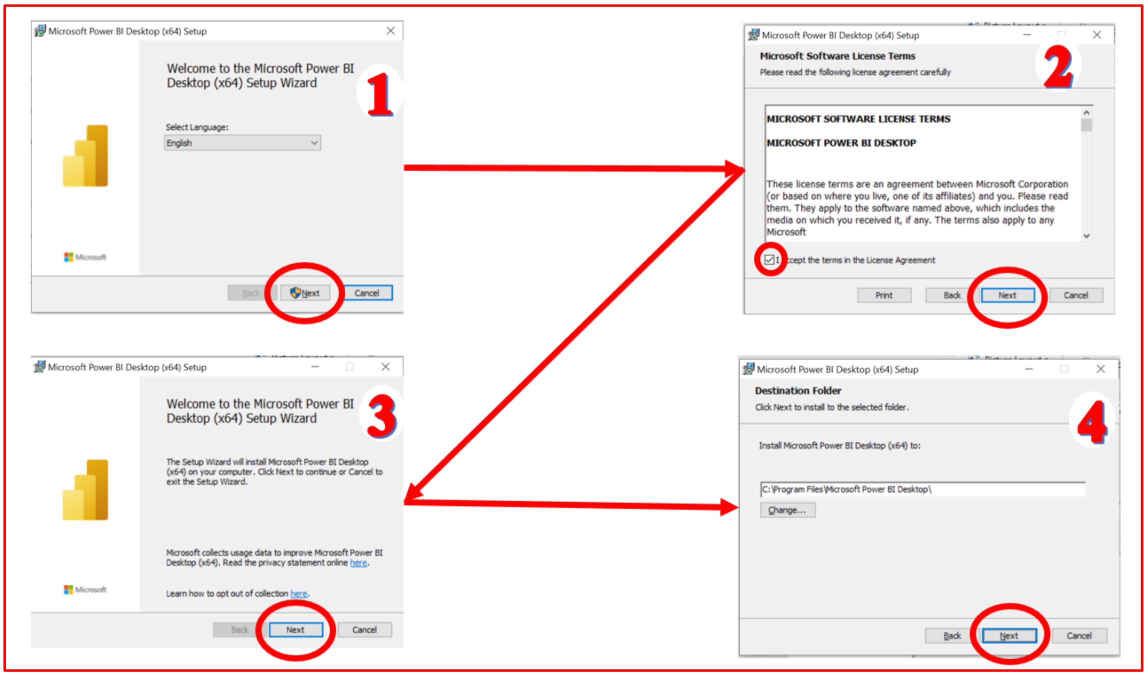
Choose the download you want

	File Name	Size
☑	PBIDesktopSetup_x64.exe	328.3 MB
□	PBIDesktopSetup.exe	301.7 MB

Download Summary:
KBMBGB

1. PBIDesktopSetup_x64.exe

Total Size: 328.3 MB



1 Welcome to the Microsoft Power BI Desktop (x64) Setup Wizard

Select Language: English

2 Microsoft Software License Terms

Please read the following license agreement carefully

MICROSOFT SOFTWARE LICENSE TERMS

MICROSOFT POWER BI DESKTOP

These license terms are an agreement between Microsoft Corporation (or based on where you live, one of its affiliates) and you. Please read them. They apply to the software named above, which includes the media on which you received it, if any. The terms also apply to any Microsoft

☑ I accept the terms in the License Agreement

3 Welcome to the Microsoft Power BI Desktop (x64) Setup Wizard

The Setup Wizard will install Microsoft Power BI Desktop (x64) on your computer. Click Next to continue or Cancel to exit the Setup Wizard.

Microsoft collects usage data to improve Microsoft Power BI Desktop (x64). Read the privacy statement online here.

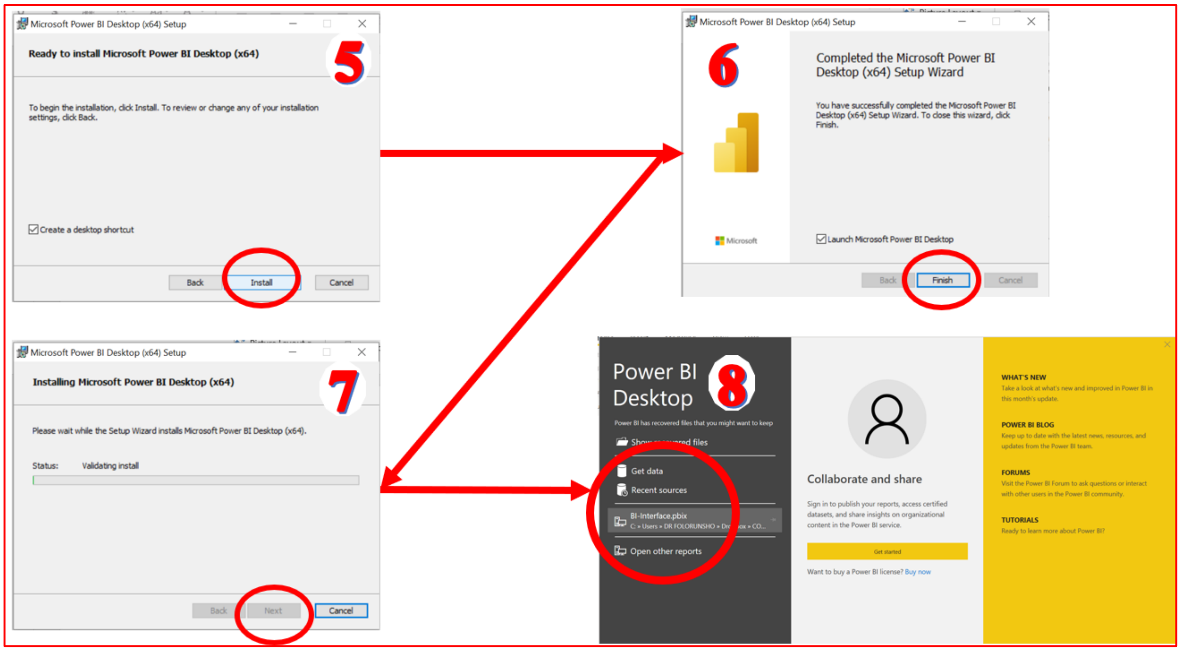
Learn how to opt out of collection here.

4 Destination Folder

Click Next to install to the selected folder.

Install Microsoft Power BI Desktop (x64) to:

C:\Program Files\Microsoft Power BI Desktop\



5 Ready to install Microsoft Power BI Desktop (x64)

To begin the installation, click Install. To review or change any of your installation settings, click Back.

☑ Create a desktop shortcut

6 Completed the Microsoft Power BI Desktop (x64) Setup Wizard

You have successfully completed the Microsoft Power BI Desktop (x64) Setup Wizard. To close this wizard, click Finish.

☑ Launch Microsoft Power BI Desktop

7 Installing Microsoft Power BI Desktop (x64)

Please wait while the Setup Wizard installs Microsoft Power BI Desktop (x64).

Status: Validating install

8 Power BI has recovered files that you might want to keep

BI-Interface.pbix
C: » Users » DR FOLORUNSHO » Dr...ox » CO...

Collaborate and share

Sign in to publish your reports, access certified datasets, and share insights on organizational content in the Power BI service.

Want to buy a Power BI license? Buy now

WHAT'S NEW
Take a look at what's new and improved in Power BI in this month's update.

POWER BI BLOG
Keep up to date with the latest news, resources, and updates from the Power BI team.

FORUMS
Visit the Power BI Forum to ask questions or interact with other users in the Power BI community.

TUTORIALS
Ready to learn more about Power BI?

8.6.3 Power BI Desktop

Power BI Desktop is a free application that can be installed on a local computer that allows connection, transform, and visualize a dataset as shown by Figure 52. All necessary click buttons are either circled red or have a red arrow pointing to them. Power BI Desktop permits the connection to multiple diverse sources of data, and combine them into a data model. This data model lets you build visuals, and collections of visuals you can share as reports, with other people inside your organization. Most users who work on business intelligence projects use Power BI Desktop to create reports, and then use the Power BI service to share their reports with others.

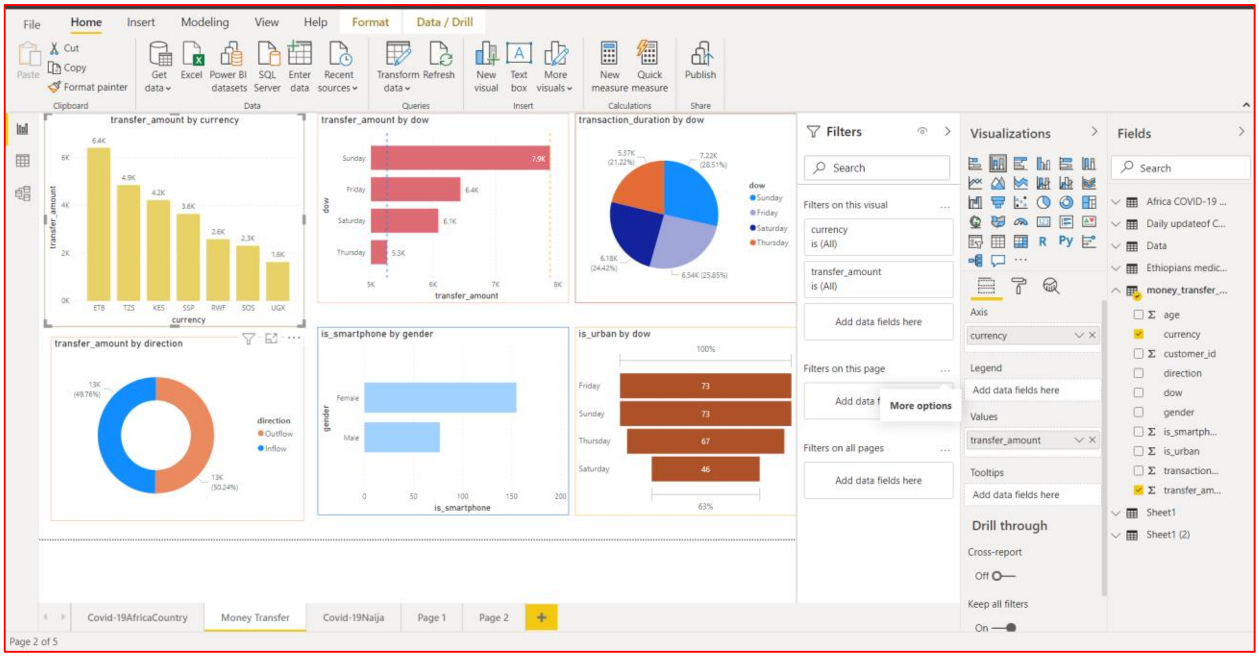


Figure 52 A typical Power BI Desktop visual display

Common uses for Power BI Desktop

The commonest uses of Power BI Desktop are:

Connect to data

Transform and clean that data, to create a data model

Create visuals, such as charts or graphs, that provide visual representations of the data

Create reports that are collections of visuals, on one or more report pages

Share reports with others by using the Power BI service

Views to Power BI Desktop

There are three views that are accessible in Power BI Desktop appearing on the left side of the canvas. The views, menu bar and result pane are shown in Figure 53

Report view: Reports and visuals are created from charts here.

Data view: Tables, measures, and other data used in the data model associated with report, and transform the data for best use in the report's model.

Model view: Tables and their relationships in the data model are seen and managed here.

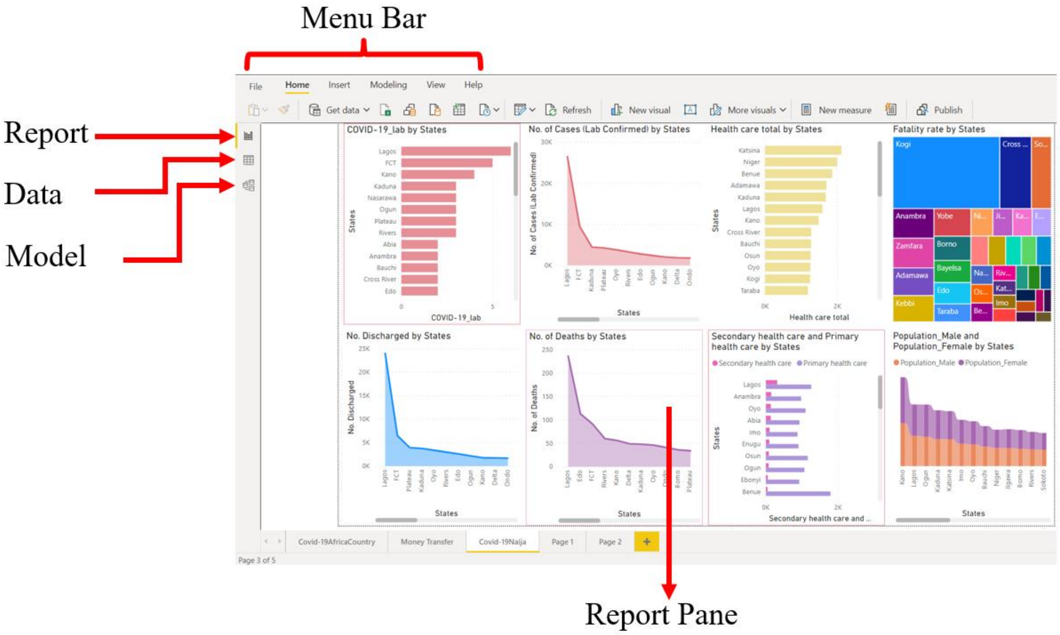


Figure 53 The three views of Power BI Desktop

Connect to data

To get started with Power BI Desktop, the first step is to connect to data. There are many different data sources you can connect to from Power BI Desktop.

To connect to data:

1. From the Home ribbon, select ‘**Get Data’** > More.

The Get Data window appears, showing the many categories to which Power BI Desktop can connect. Choose ‘**All**’ to display all available data sources or ‘**other’** for specific formats as displayed by Figure 54

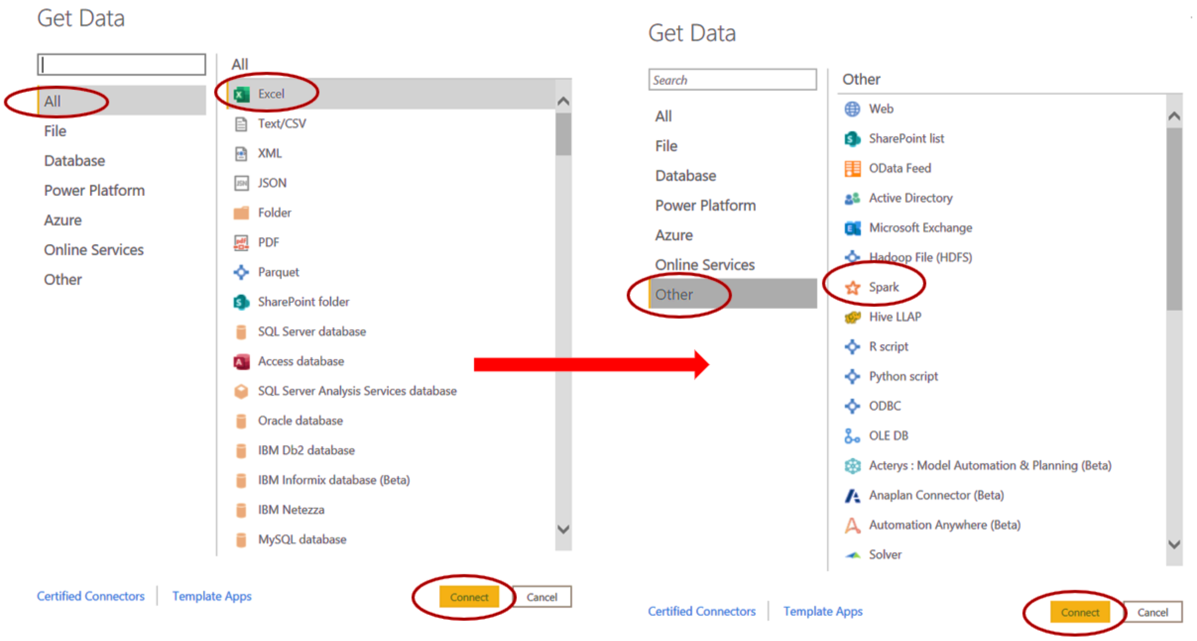


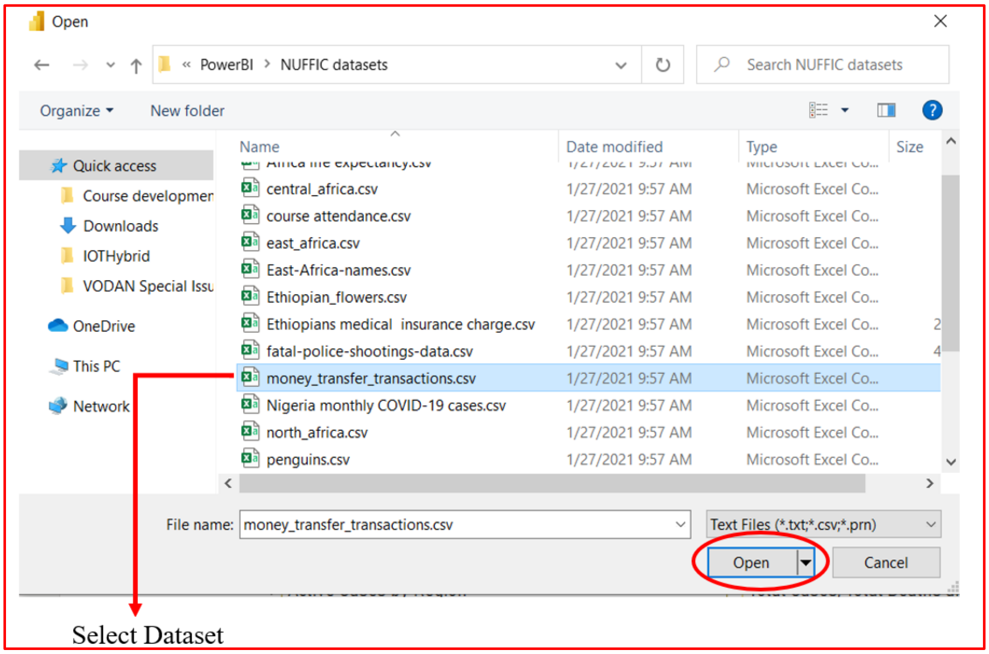
Figure 54 Data Sources

8.6.4 Data Sources for the Power BI

Data source	Description
Excel (.xlsx, xls)	A Microsoft Excel workbook data format. This can be loaded from external data sources.
Comma Separated Value (.csv) or Text	A text file format separated by a comma. Each of the rows (instances) will contain data in a particular domain.
Power BI Desktop (.pbi)	This is an online Power BI Desktop to query and load data from Power BI online services
SQL Server	Data from a SQL server database
Web	Data imported from a web page. It could be from any format described above
Databases in the Cloud	It allows you to connect live to Azure SQL Database, Azure SQL Data Warehouse, etc.
Databases on-premises	You can connect directly to SQL Server Analysis Services Relational model databases. A Power BI Enterprise Gateway is required.

More…	More dataset format

5. Once the preferred data type is selected, you're prompted for information, such as the data location source on your computer necessary for Power BI Desktop to connect to the data source on your behalf.



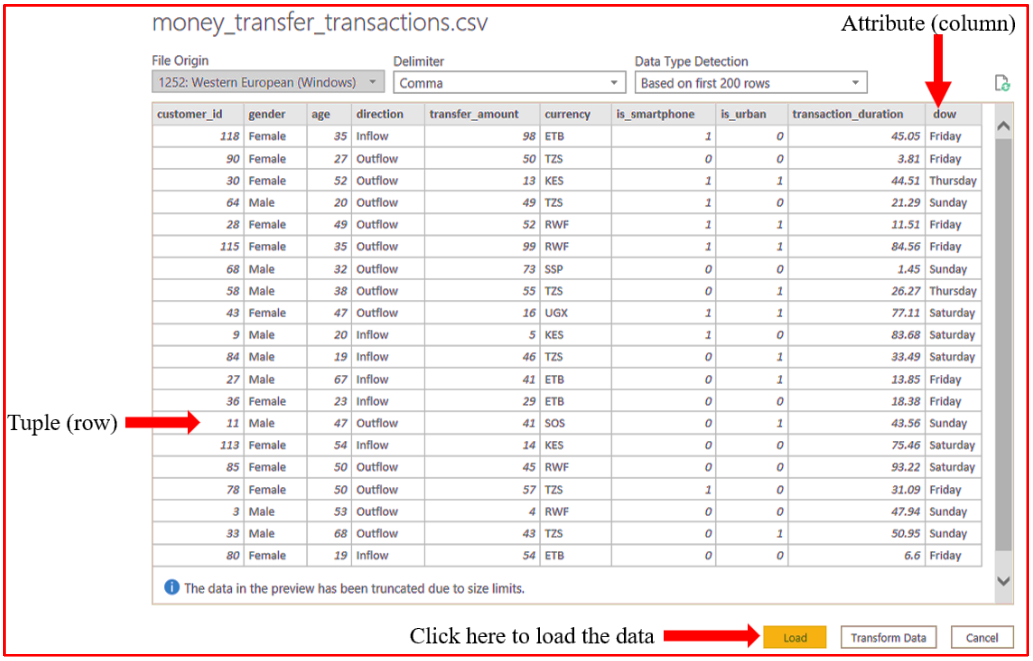
Name	Date modified	Type	Size
Africa life expectancy.csv	1/27/2021 9:57 AM	Microsoft Excel Co...	
central_africa.csv	1/27/2021 9:57 AM	Microsoft Excel Co...	
course attendance.csv	1/27/2021 9:57 AM	Microsoft Excel Co...	
east_africa.csv	1/27/2021 9:57 AM	Microsoft Excel Co...	
East-Africa-names.csv	1/27/2021 9:57 AM	Microsoft Excel Co...	
Ethiopian_flowers.csv	1/27/2021 9:57 AM	Microsoft Excel Co...	
Ethiopians medical insurance charge.csv	1/27/2021 9:57 AM	Microsoft Excel Co...	2
fatal-police-shootings-data.csv	1/27/2021 9:57 AM	Microsoft Excel Co...	4
money_transfer_transactions.csv	1/27/2021 9:57 AM	Microsoft Excel Co...	
Nigeria monthly COVID-19 cases.csv	1/27/2021 9:57 AM	Microsoft Excel Co...	
north_africa.csv	1/27/2021 9:57 AM	Microsoft Excel Co...	
penguins.csv	1/27/2021 9:57 AM	Microsoft Excel Co...	

File name: money_transfer_transactions.csv

Text Files (*.txt;*.csv;*.prn)

Select Dataset

3. After you connect to one or more data sources, you may want to transform the data so it's useful for you.



money_transfer_transactions.csv

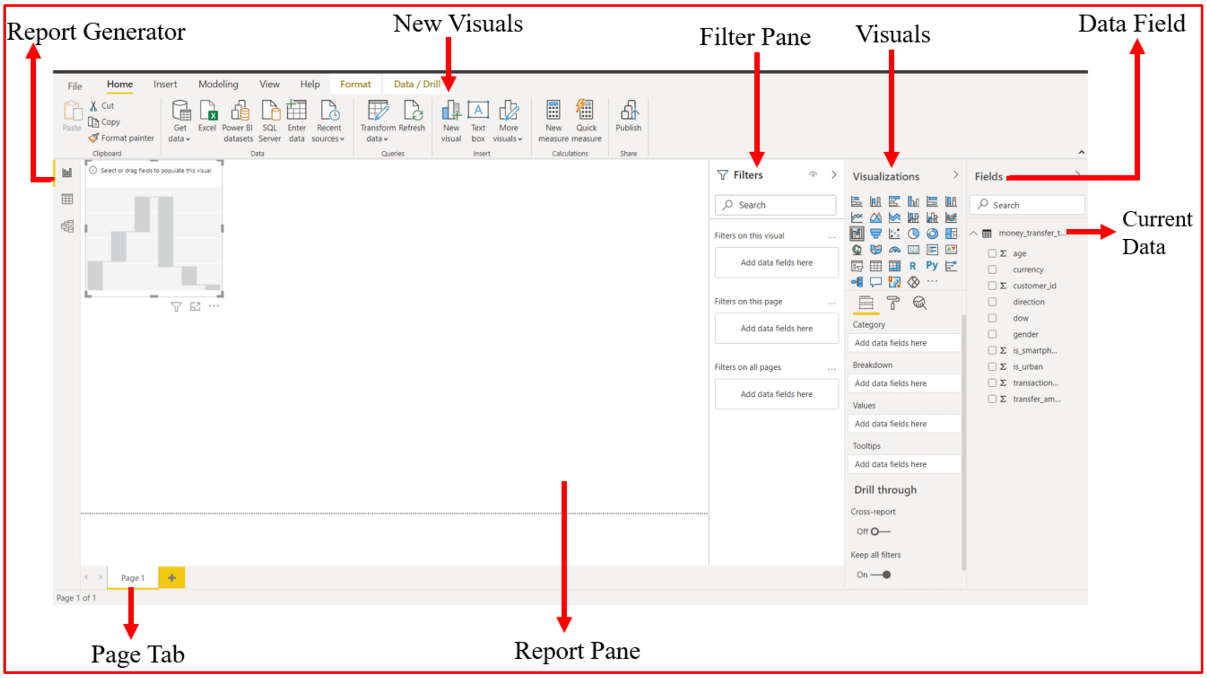
File Origin: 1252: Western European (Windows)

Delimiter: Comma

Data Type Detection: Based on first 200 rows

customer_id	gender	age	direction	transfer_amount	currency	is_smartphone	is_urban	transaction_duration	dow
118	Female	35	Inflow	98	ETB	1	0	45.05	Friday
90	Female	27	Outflow	50	TZS	0	0	3.81	Friday
30	Female	52	Outflow	13	KES	1	1	44.51	Thursday
64	Male	20	Outflow	49	TZS	1	0	21.29	Sunday
28	Female	49	Outflow	52	RWF	1	1	11.51	Friday
115	Female	35	Outflow	99	RWF	1	1	84.56	Friday
68	Male	32	Outflow	73	SSP	0	0	1.45	Sunday
58	Male	38	Outflow	55	TZS	0	1	26.27	Thursday
43	Female	47	Outflow	16	UGX	1	1	77.11	Saturday
9	Male	20	Inflow	5	KES	1	0	83.68	Saturday
84	Male	19	Inflow	46	TZS	0	1	33.49	Saturday
27	Male	67	Inflow	41	ETB	0	1	13.85	Friday
36	Female	23	Inflow	29	ETB	0	0	18.38	Friday
11	Male	47	Outflow	41	SOS	0	1	43.56	Sunday
113	Female	54	Inflow	14	KES	0	0	75.46	Saturday
85	Female	50	Outflow	45	RWF	0	0	93.22	Saturday
78	Female	50	Outflow	57	TZS	1	0	31.09	Friday
3	Male	53	Outflow	4	RWF	0	0	47.94	Sunday
33	Male	68	Outflow	43	TZS	0	1	50.95	Sunday
80	Female	19	Inflow	54	ETB	0	0	6.6	Friday

The data in the preview has been truncated due to size limits.



a. The Report generator icon activates the report mode

b. The new visuals tab at the top displays the common tasks related to reports and visualizations.

c. Visualizations are created and arranged in the report pane.

d. The pages tab area at the bottom lets you create, select or add report pages.

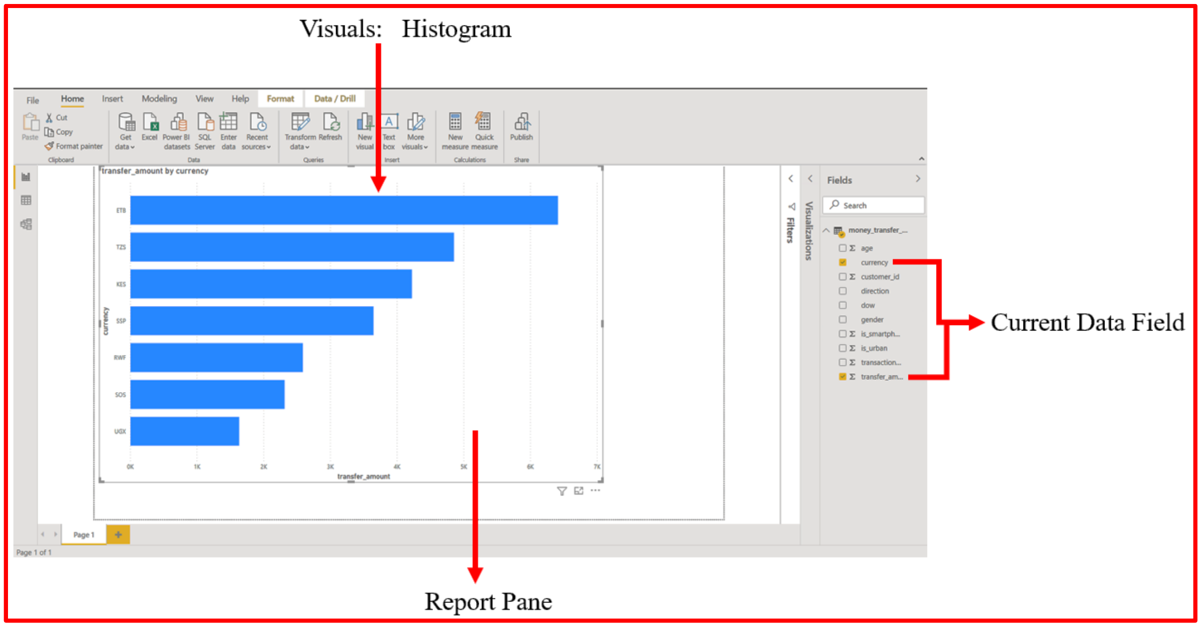
e. The Filter pane is where you can filter data visualizations by selecting data fields in the data fields pane.

f. The Visuals pane permits you to pick, add, change, or customize visualizations, and apply drill through on your data.

g. The Data fields pane shows the available data fields in your queries. You can drag these fields onto the report pane, the Filter pane, or the Visuals pane to create or modify visualizations.

h. You can expand and collapse the Filters, Visualizations, and Fields panes by selecting the arrows (>) at the tops of the panes. Collapsing the panes provides more space on the canvas to build cool visualizations.

To create a simple visualization, you pick and double-click a preferred field in the fields' list, or drag the field to the report pane. For example, double-click or drag the 'currency' and 'transfer_amount' field from 'money_transfer_transactions onto the report pane.

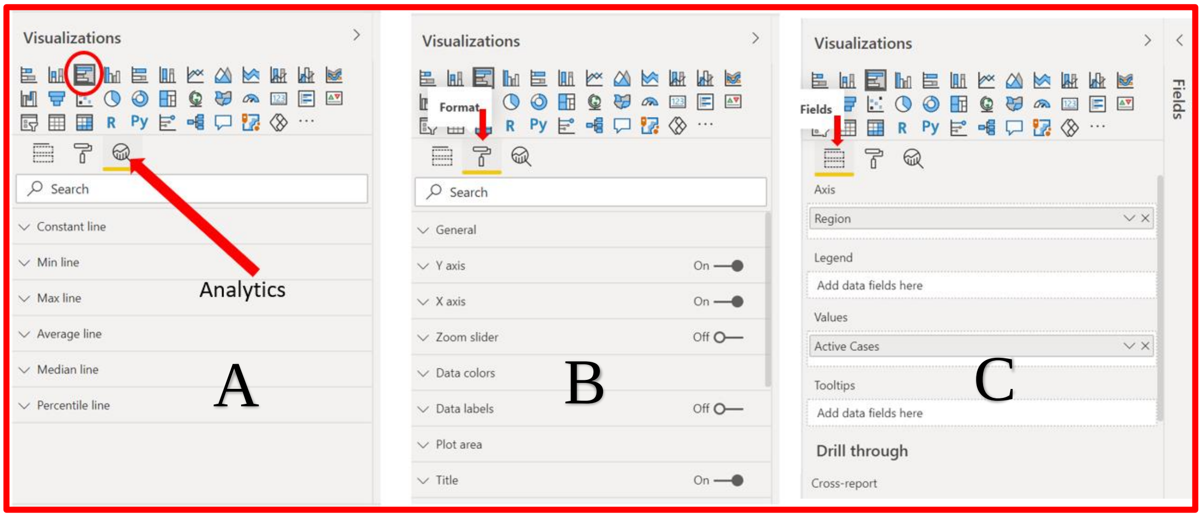

Visuals: Histogram

Current Data Field

Report Pane

Power BI Desktop recognized that the currency and transaction_duration field and automatically created a Histogram visualization. The visualization shows data points for the currency and transfer_amount from your data model.

The Visuals pane shows information about the visualization and allows you change it.



Visualizations

Search

Constant line
Min line
Max line
Average line
Median line
Percentile line

Visualizations

Search

General
Y axis On
X axis On
Zoom slider Off
Data colors
Data labels Off
Plot area
Title On

Visualizations

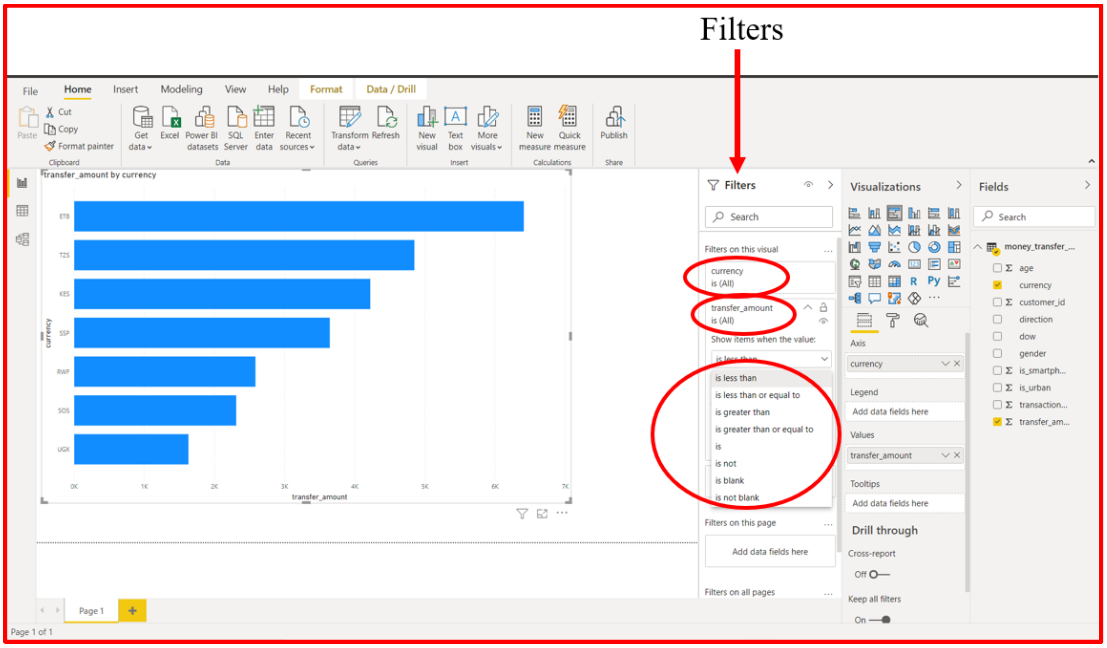
Axis
Region
Legend
Add data fields here
Values
Active Cases
Tooltips
Add data fields here
Drill through
Cross-report

A. The red-encircled-icon show the type of visualization created. You can change the type of a selected visualization by selecting a different icon, or create a new visualization by selecting an icon with no existing visualization selected. You can also perform different types of descriptive analytics like constant, Minimum, Maximum, Average, Median and Percentile lines on the data. Just click on the arrow (V).

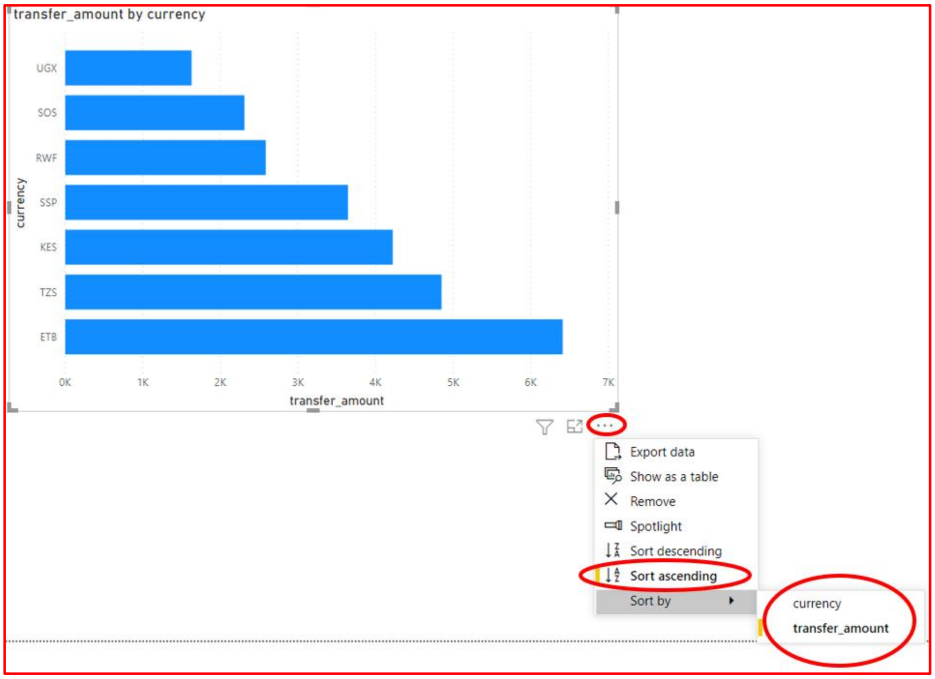
B. The Format selection allows you apply formatting and other controls to visualizations. For example, you can edit the values and title on either ‘X’ or ‘Y’ axis. You can also decide to show the labels of the data or change their colors.

C. The Fields choice in the Visuals pane allows you choose which field will be on which axis, what data fields to be the Legend etc.

The choices accessible in the Fields and Format pane depend on the type of visuals and data you have.

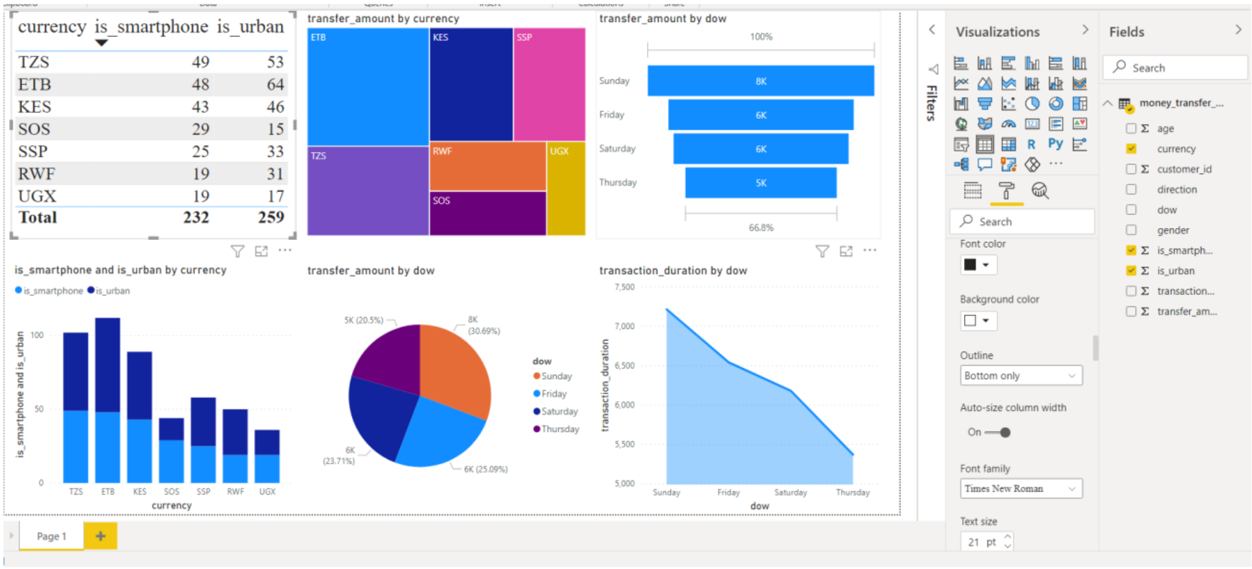


The Filters denoted by the funnel image allows you to control the data fields and hence the visualization. You can use all currency ‘is (All)’ type and decide on the transfer_amount. You can choose a particular value as standard and decide on the values less than the standard or more than. Then choose Apply filter.



To change the order of the position of the bars such that the first to appears, click the More options denoted by (⋯) at the bottom right of the visuals (Histogram), and select Sort ascending from the menu to give the desired visual. You could also decide to sort according to any of the data fields (currency or transfer_amount). You could also decide to export the data in a specified format.

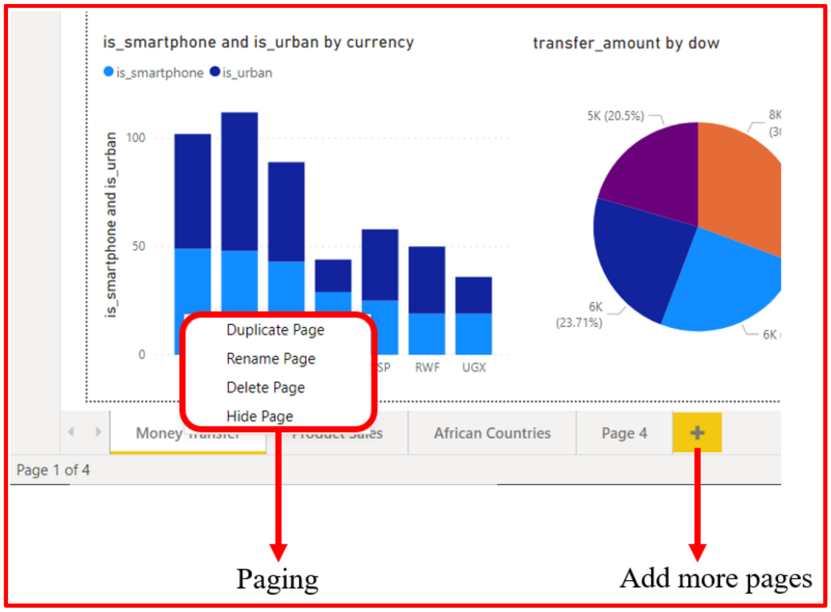
You can make similar or other visualizations for the other fields, or combine several fields into one visualization. There are all sorts of interesting reports and visualizations you can create. These Table and Line and clustered column chart visualizations shows the currency used by is_smartphone and is_urban. The treemap shows the transfer_amount by currency, the pie chart shows the slices and percentages of transfer_amount by the day of the week.



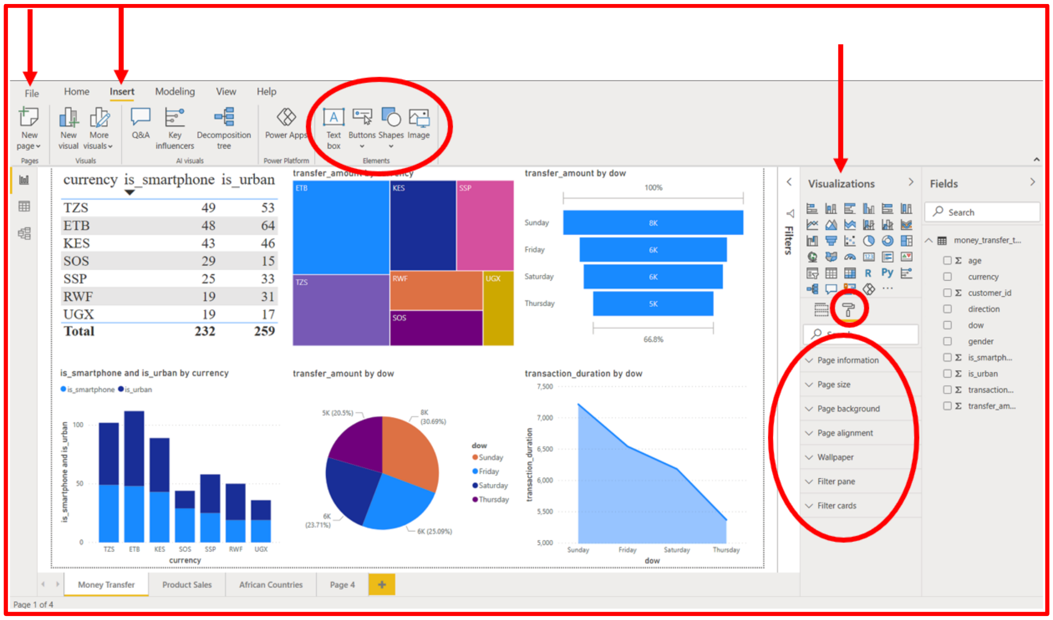
currency	is_smartphone	is_urban
TZS	49	53
ETB	48	64
KES	43	46
SOS	29	15
SSP	25	33
RWF	19	31
UGX	19	17
Total	**232**	**259**

Font color
Background color
Outline: Bottom only
Auto-size column width: On
Font family: Times New Roman
Text size: 21 pt

You can show different visuals on different report pages. To add a new page, select the (+) symbol next to the existing pages on the pages bar, or click on the Insert button on the menu bar. Then, click New Page tab of the ribbon. To rename a page, double-click the page name in the pages bar, or right-click it and select Rename Page, and then type the new name. To go to a different page of the report, select the page from the pages bar.



Text boxes, shapes, images, and buttons of different shades can be added to report pages for more clarifications. To achieve this, click on the Insert tab on the menu bar. To set formatting choices for visualizations, pick a preferred chart form the visualization pane, then click the Format icon in the Visualizations pane. To construct page sizes, alignment, backgrounds, and other page information, click on the Format icon with no chart in the visualization pane selected. When you finish creating your pages and visualizations, click on File at the upper left corner on the page, then click Save to save your report.



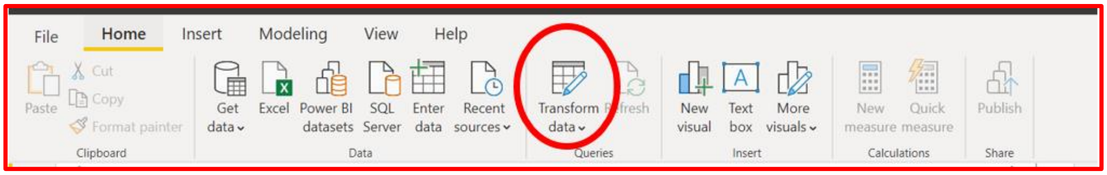
currency	is_smartphone	is_urban
TZS	49	53
ETB	48	64
KES	43	46
SOS	29	15
SSP	25	33
RWF	19	31
UGX	19	17
Total	232	259

The Power BI Query Editor allows you edit or 'shape' data before it gets visualized.

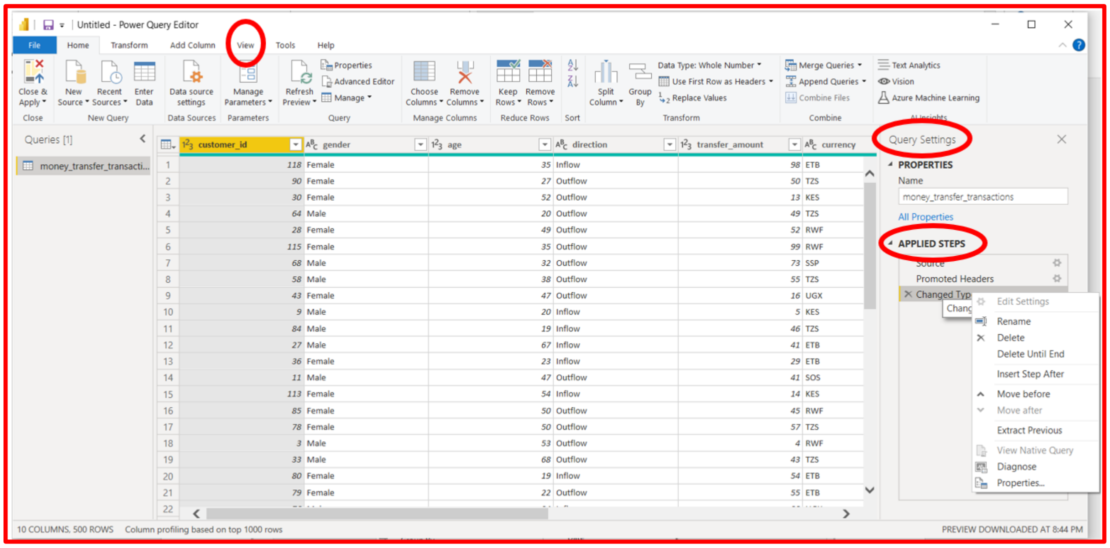
The Query Editor permits you to remove duplicate rows, add, remove and pivot columns, change text fields to numbers and perform calculations on data prior to visualization. It also keeps a record of the steps taken, enabling it to apply the same manipulations every time the data is refreshed and provide a visual history so you can undo and further tweak the actions as required.

All necessary click buttons are either circled red or have a red arrow pointing to them.

To access the Power Query Editor with a representative view of the table or relation, click on the **Transform data** tab on the home menu bar.



On the Query Editor's page, you can set queries by clicking on the Query setting's pane on the far right, or use the **View** tab of Power Query Editor's menu bar.



	customer_id	gender	age	direction	transfer_amount	currency
1	118	Female	35	Inflow	98	ETB
2	90	Female	27	Outflow	50	TZS
3	30	Female	52	Outflow	13	KES
4	64	Male	20	Outflow	49	TZS
5	28	Female	49	Outflow	52	RWF
6	115	Female	35	Outflow	99	RWF
7	68	Male	32	Outflow	73	SSP
8	58	Male	38	Outflow	55	TZS
9	43	Female	47	Outflow	16	UGX
10	9	Male	20	Inflow	5	KES
11	84	Male	19	Inflow	46	TZS
12	27	Male	67	Inflow	41	ETB
13	36	Female	23	Inflow	29	ETB
14	11	Male	47	Outflow	41	SOS
15	113	Female	54	Inflow	14	KES
16	85	Female	50	Outflow	45	RWF
17	78	Female	50	Outflow	57	TZS
18	3	Male	53	Outflow	4	RWF
19	33	Male	68	Outflow	43	TZS
20	80	Female	19	Inflow	54	ETB
21	79	Female	22	Outflow	55	ETB
22						

Name: money_transfer_transactions

APPLIED STEPS
- Source
- Promoted Headers
- Changed Typ

10 COLUMNS, 500 ROWS Column profiling based on top 1000 rows

PREVIEW DOWNLOADED AT 8:44 PM

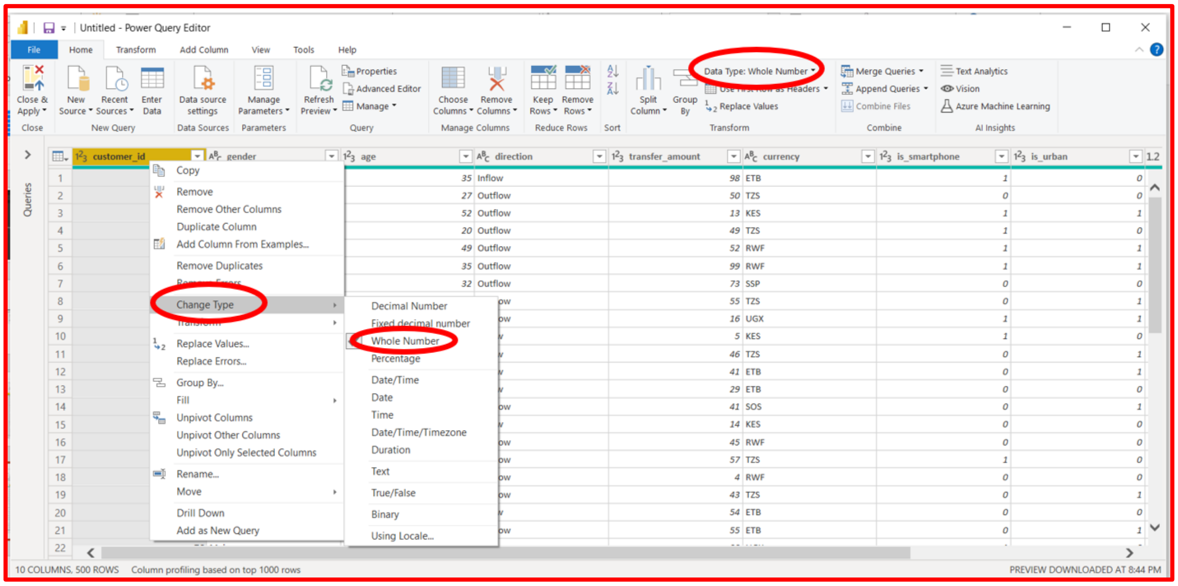
For more information about connecting to data, see Connect to data in Power BI Desktop.

Shaping your data

Now that you're connected to a data source, you can adjust the data to meet your needs. To *shape* data, you provide Power Query Editor with step-by-step instructions for adjusting the data while loading and presenting it. Shaping doesn't affect the original data source, only this particular view of the data.

Shaping can mean *transforming* the data, such as renaming columns or tables, removing rows or columns, or changing data types. Power Query Editor captures these steps sequentially under **Applied Steps** in the **Query Settings** pane. Each time this query connects to the data source, those steps are carried out, so the data is always shaped the way you specify. This process occurs when you use the query in Power BI Desktop, or when anyone uses your shared query, such as in the Power BI service.

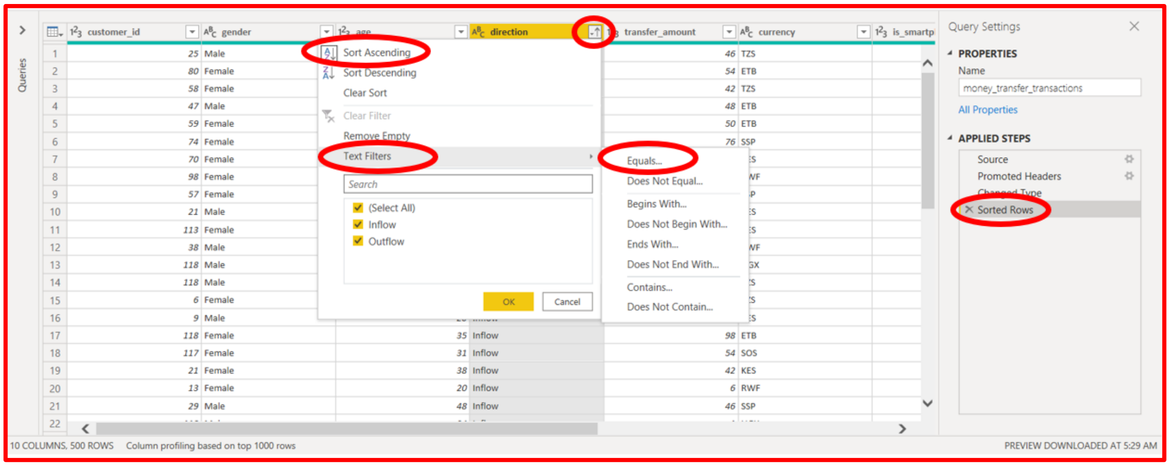
Notice that the **Applied Steps** in **Query Settings** already contain a few steps. You can select each step to see its effect in the Power Query Editor. First, you specified a web source, and then you previewed the table in the **Navigator** window. In the third step, **Changed type**, Power BI recognized whole number data when importing it, and automatically changed the original web **Text** *data type* to **Whole numbers**.



	customer_id	gender	age	direction	transfer_amount	currency	is_smartphone	is_urban
1			35	Inflow	98	ETB	1	0
2			27	Outflow	50	TZS	0	0
3			52	Outflow	13	KES	1	1
4			20	Outflow	49	TZS	1	0
5			49	Outflow	52	RWF	1	1
6			35	Outflow	99	RWF	1	1
7			32	Outflow	73	SSP	0	0
8				ow	55	TZS	0	1
9				ow	16	UGX	1	1
10				v	5	KES	1	0
11				v	46	TZS	0	1
12				v	41	ETB	0	1
13				v	29	ETB	0	0
14				ow	41	SOS	0	1
15				v	14	KES	0	0
16				ow	45	RWF	0	0
17				ow	57	TZS	1	0
18				ow	4	RWF	0	0
19				ow	43	TZS	0	1
20				v	54	ETB	0	0
21				ow	55	ETB	0	1

10 COLUMNS, 500 ROWS Column profiling based on top 1000 rows
PREVIEW DOWNLOADED AT 8:44 PM

If you need to change a data type, select the column or columns to change. Hold down the **Shift** key to select several adjacent columns, or **Ctrl** to select non-adjacent columns. Either right-click a column header, select **Change Type**, and choose a new data type from the menu, or drop down the list next to **Data Type** in the **Transform** group of the **Home** tab, and select a new data type.



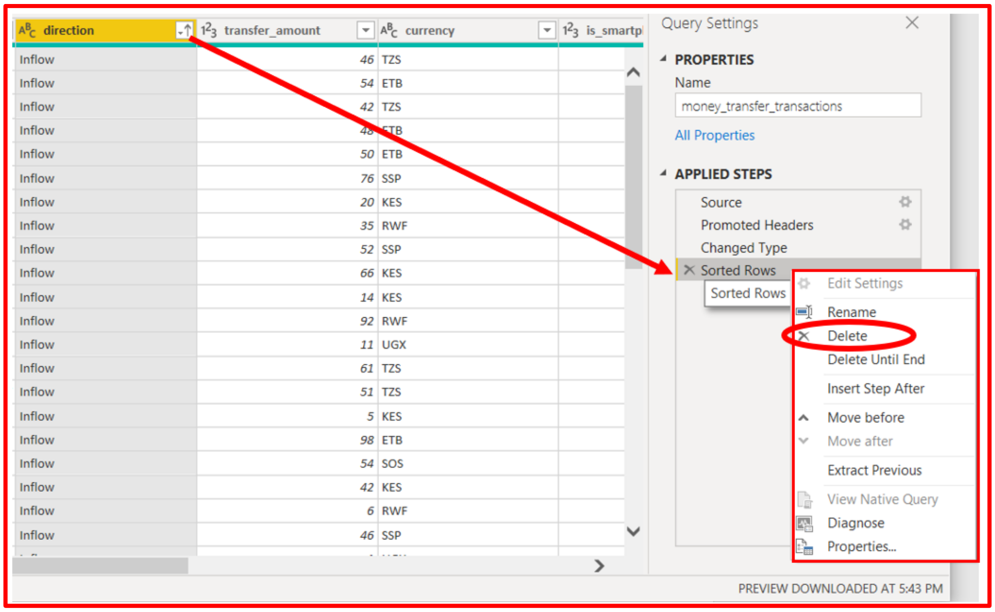
Name: money_transfer_transactions

	customer_id	gender	age	direction	transfer_amount	currency
1	25	Male			46	TZS
2	80	Female			54	ETB
3	58	Female			42	TZS
4	47	Male			48	ETB
5	59	Female			50	ETB
6	74	Female			76	SSP
7	70	Female				S
8	98	Female				VF
9	57	Female				P
10	21	Male				S
11	113	Female				S
12	38	Male				VF
13	118	Male				GX
14	118	Male				S
15	6	Female				S
16	9	Male				S
17	118	Female	35	Inflow	98	ETB
18	117	Female	31	Inflow	54	SOS
19	21	Female	38	Inflow	42	KES
20	13	Female	20	Inflow	6	RWF
21	29	Male	48	Inflow	46	SSP

10 COLUMNS, 500 ROWS Column profiling based on top 1000 rows
PREVIEW DOWNLOADED AT 5:29 AM

For example, in the money transfer dataset you wish to rank the direction of flow, so you decide to sort the table by the **direction** column instead of by **Overall rank**. Drop down the arrow next to the **direction** header, and select **Sort ascending**. The data now appears sorted by direction ranking, and the step **Sorted Rows** appears in **Applied Steps**.

You might no longer wish to sort the table (by **direction)**, you should delete the **Sorted Rows** step on the **APPLIED STEPS** pane. Right click on the **X Sorted Rows Power**, the **X Delete button**.



direction	transfer_amount	currency	is_smartp
Inflow	46	TZS	
Inflow	54	ETB	
Inflow	42	TZS	
Inflow	48	ETB	
Inflow	50	ETB	
Inflow	76	SSP	
Inflow	20	KES	
Inflow	35	RWF	
Inflow	52	SSP	
Inflow	66	KES	
Inflow	14	KES	
Inflow	92	RWF	
Inflow	11	UGX	
Inflow	61	TZS	
Inflow	51	TZS	
Inflow	5	KES	
Inflow	98	ETB	
Inflow	54	SOS	
Inflow	42	KES	
Inflow	6	RWF	
Inflow	46	SSP	

Query Settings

PROPERTIES

Name: money_transfer_transactions

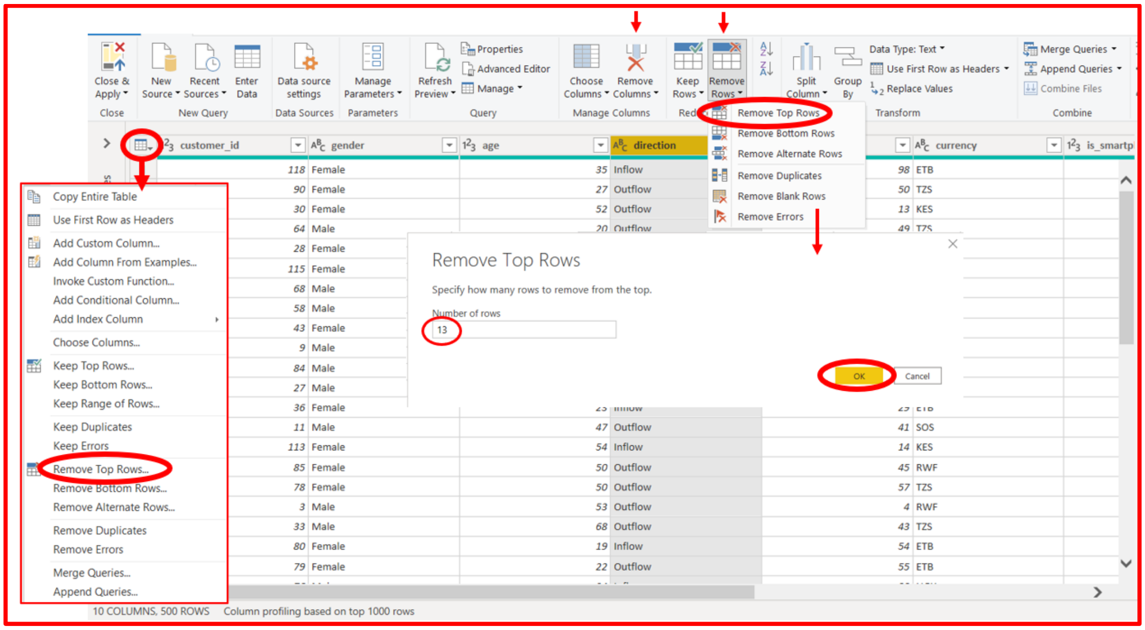
All Properties

APPLIED STEPS

Source
Promoted Headers
Changed Type
Sorted Rows

PREVIEW DOWNLOADED AT 5:43 PM

Situations could arise that you want to remove some rows or columns like **direction**. You can do this by either clicking small arrow beside the **Remove Columns** or **Remove Rows** tab. For example, to remove rows from **direction** column, click on **Reduce Rows** button and choose from the drop-down menu on the rows to remove. If you **Remove Top Rows, a** dialog box appears, enter *13* (or any number of your choice) to remove the top 13 rows from your data and then click **OK**.



Remove Top Rows

Specify how many rows to remove from the top.

Number of rows

13

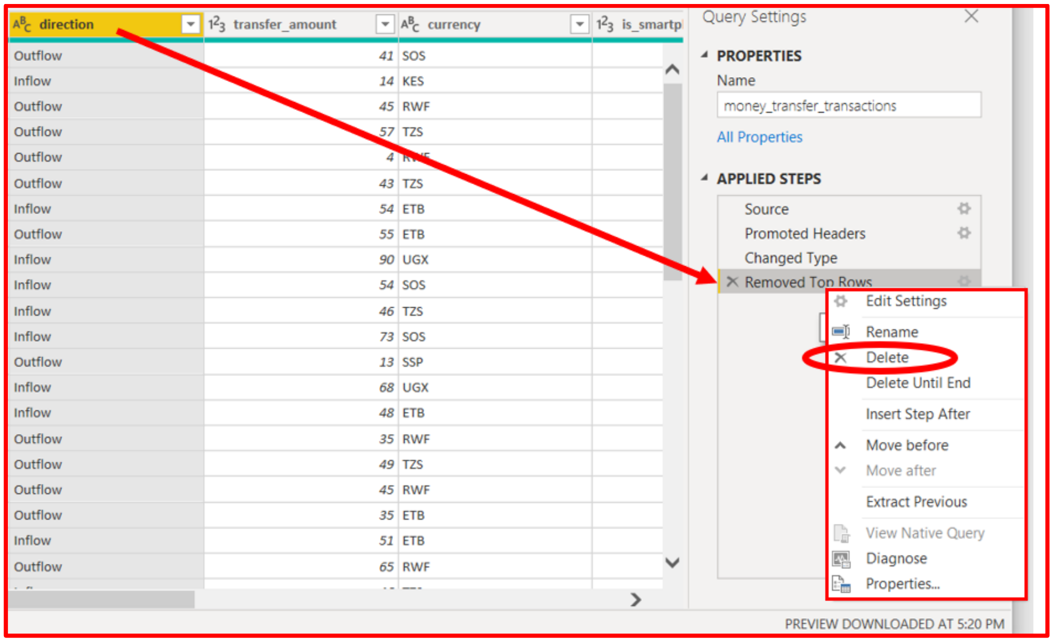
customer_id	gender	age	direction		currency	is_smartp
118	Female	35	Inflow	98	ETB	
90	Female	27	Outflow	50	TZS	
30	Female	52	Outflow	13	KES	
64	Male	20	Outflow	49	TZS	
28	Female					
115	Female					
68	Male					
58	Male					
43	Female					
9	Male					
84	Male					
27	Male					
36	Female	23	Inflow	29	ETB	
11	Male	47	Outflow	41	SOS	
113	Female	54	Inflow	14	KES	
85	Female	50	Outflow	45	RWF	
78	Female	50	Outflow	57	TZS	
3	Male	53	Outflow	4	RWF	
33	Male	68	Outflow	43	TZS	
80	Female	19	Inflow	54	ETB	
79	Female	22	Outflow	55	ETB	

10 COLUMNS, 500 ROWS Column profiling based on top 1000 rows

The top 13 distribution rows are removed from the table, and the step **Removed Bottom Rows** appears in **Applied Steps**. You can also do all these activities by clicking on the table icon on the far left the data column header.

You decide the table has too much extra information for your needs, and to remove more columns, select the header of each column that you want to remove. Hold down the **Shift** key to select several adjacent columns, or **Ctrl** to select non-adjacent columns.

Then, from the **Manage Columns** group of the **Home** tab, select **Remove Columns**. You can also right-click one of the selected column headers and select **Remove Columns** from the menu. The selected columns are removed, and the step **Removed Columns** appears in **Applied Steps**.



direction	transfer_amount	currency	is_smartp
Outflow	41	SOS	
Inflow	14	KES	
Outflow	45	RWF	
Outflow	57	TZS	
Outflow	4	RWF	
Outflow	43	TZS	
Inflow	54	ETB	
Outflow	55	ETB	
Inflow	90	UGX	
Inflow	54	SOS	
Inflow	46	TZS	
Inflow	73	SOS	
Outflow	13	SSP	
Inflow	68	UGX	
Inflow	48	ETB	
Outflow	35	RWF	
Outflow	49	TZS	
Outflow	45	RWF	
Outflow	35	ETB	
Inflow	51	ETB	
Outflow	65	RWF	

Query Settings

PROPERTIES

Name: money_transfer_transactions

All Properties

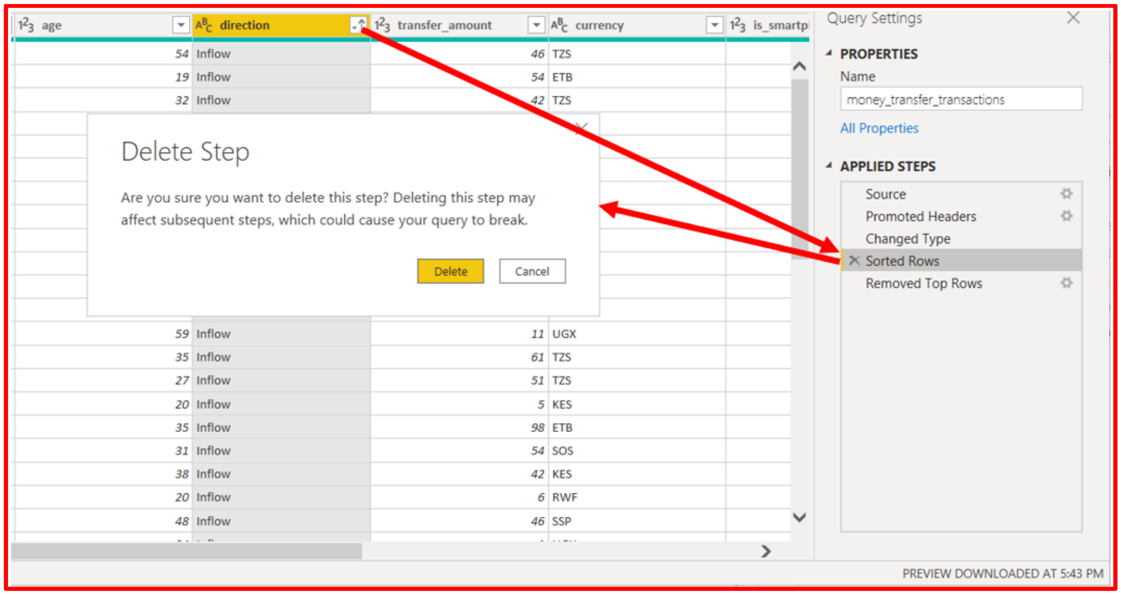
APPLIED STEPS

- Source
- Promoted Headers
- Changed Type
- Removed Top Rows

PREVIEW DOWNLOADED AT 5:20 PM

But if decide to get your deleted row or columns back, you can undo the last step in the **Applied Steps** pane by selecting the **X delete** icon next to the step. Now redo the step, selecting only the columns you want to delete. For more flexibility, you could delete each column as a separate step. You can right-click any step in the **Applied Steps** pane and choose to delete it, rename it, move it up or down in the sequence, or add or delete steps after it.

Remember that you had sorted the **direction** earlier. So, if you no longer wish to sort and try to delete this action, POWER BI Desktop will warn you that deleting this step could cause your query to break. You removed the top 13 rows after you sorted by **direction**, so if you remove the sort, different rows will be removed. You also get a warning if you select the **Sorted Rows** step and try to add a new intermediate step at that point.



age	direction	transfer_amount	currency	is_smartp
54	Inflow	46	TZS	
19	Inflow	54	ETB	
32	Inflow	42	TZS	
59	Inflow	11	UGX	
35	Inflow	61	TZS	
27	Inflow	51	TZS	
20	Inflow	5	KES	
35	Inflow	98	ETB	
31	Inflow	54	SOS	
38	Inflow	42	KES	
20	Inflow	6	RWF	
48	Inflow	46	SSP	

Delete Step

Are you sure you want to delete this step? Deleting this step may affect subsequent steps, which could cause your query to break.

Delete | Cancel

Query Settings

PROPERTIES

Name

money_transfer_transactions

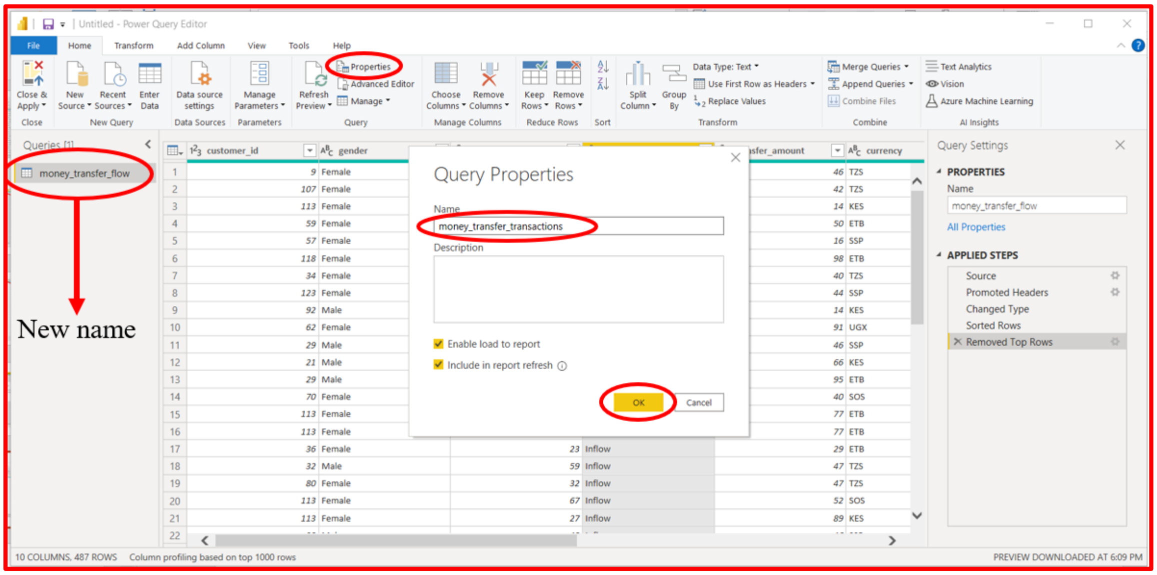
All Properties

APPLIED STEPS

- Source
- Promoted Headers
- Changed Type
- Sorted Rows
- Removed Top Rows

PREVIEW DOWNLOADED AT 5:43 PM

Finally, you change the table title from **money_transfer_transactions** to **money_transfer_flow** in the Query Properties. Click **Properties** tab in the menu bar of **Query Settings** pane, replace the old title with **money_transfer_flow**.



Queries [1]

money_transfer_flow

New name

	customer_id	gender			transfer_amount	currency
1	9	Female			46	TZS
2	107	Female			42	TZS
3	113	Female			14	KES
4	59	Female			50	ETB
5	57	Female			16	SSP
6	118	Female			98	ETB
7	34	Female			40	TZS
8	123	Female			44	SSP
9	92	Male			14	KES
10	62	Female			91	UGX
11	29	Male			46	SSP
12	21	Male			66	KES
13	29	Male			95	ETB
14	70	Female			40	SOS
15	113	Female			77	ETB
16	113	Female			77	ETB
17	36	Female	23	Inflow	29	ETB
18	32	Male	59	Inflow	47	TZS
19	80	Female	32	Inflow	47	TZS
20	113	Female	67	Inflow	52	SOS
21	113	Female	27	Inflow	89	KES

Query Properties

Name

money_transfer_transactions

Description

☑ Enable load to report

☑ Include in report refresh

OK | Cancel

Query Settings

PROPERTIES

Name

money_transfer_flow

All Properties

APPLIED STEPS

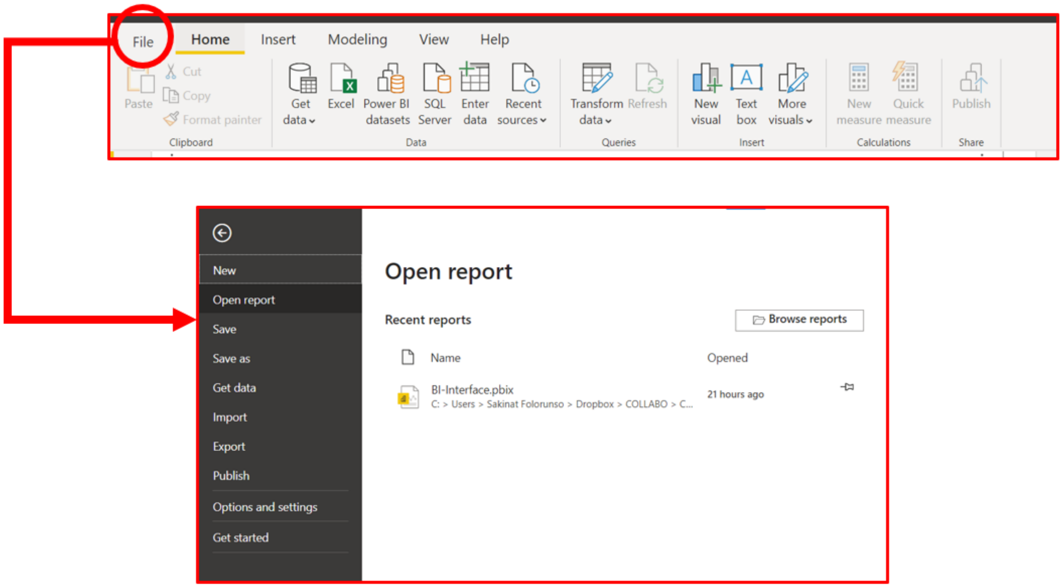
- Source
- Promoted Headers
- Changed Type
- Sorted Rows
- Removed Top Rows

10 COLUMNS, 487 ROWS   Column profiling based on top 1000 rows   PREVIEW DOWNLOADED AT 6:09 PM

For more information about shaping data, see Shape and combine data in Power BI Desktop.

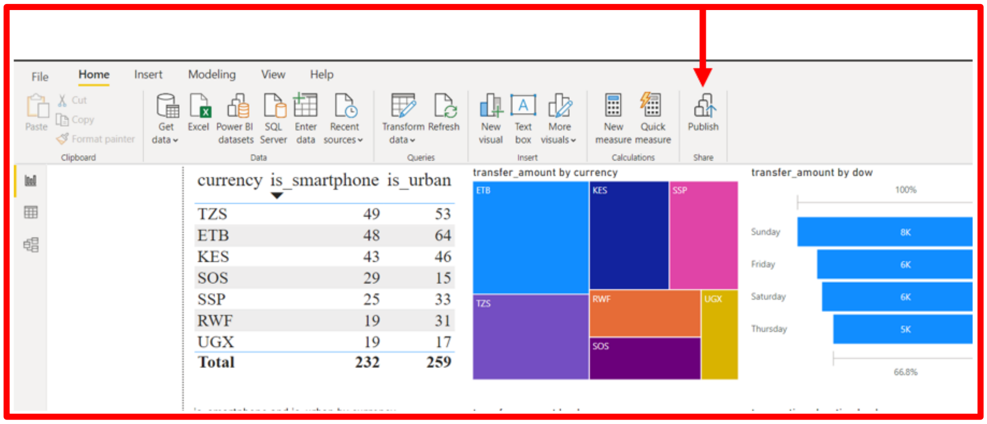
Saving your work

You can save your new work or edited work by clicking on the **File** tab on the home menu bar. The pop up window will give you series of options to save your work.



Sharing your work

Now that a report has been created on the Power BI Desktop, it can be shared with others. One of the ways is to click on the Publish tab directly on the menu bar to the Power BI service. Or you can upload the **.pbix** file from the Power BI service. To achieve this, you first create a Power BI account.



currency	is_smartphone	is_urban
TZS	49	53
ETB	48	64
KES	43	46
SOS	29	15
SSP	25	33
RWF	19	31
UGX	19	17
Total	**232**	**259**

Additional Resources

For more resources on Power BI, please consider the following:

- Power BI Desktop YouTube Channel
- To learn high-level overview of Power BI Desktop, from getting data and building report, to sharing your report with others: Link
- Learn how to build stunning reports using Power BI Desktop: Link
- Import, reshape, and transform data using Power Query Editor: Link

Select or click on the links to watch the videos available at the Power BI YouTube channel:

- Get started with Power BI Desktop: Link
- Create a report in Power BI Desktop: Link
- Use the Power Query Editor: Link
- Create relationships between tables: Link
- Publish from Power BI Desktop to the Power BI service: Link
- Add a calculated column: Link

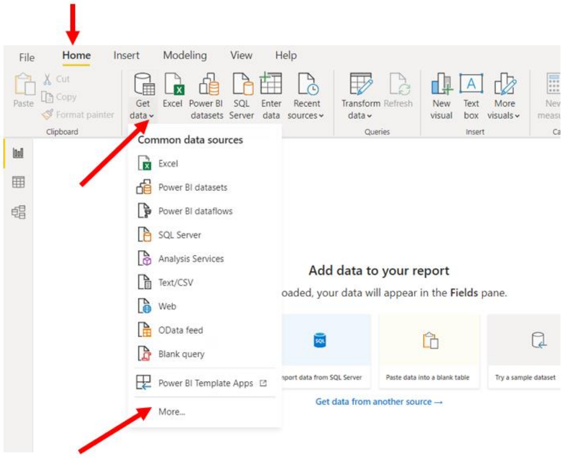
Practice Exercise 1:

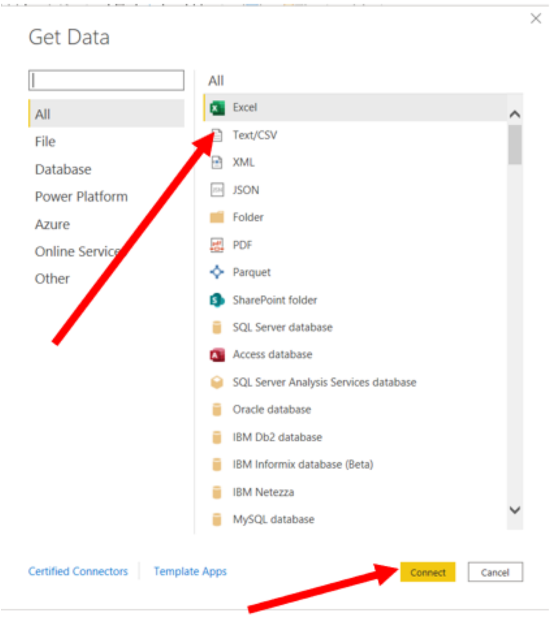
Load the **money_tranfer** dataset saved on your local computer into the power BI. How many attributes are there? List them out all the attributes. Use two attributes ‘currency’ and ‘transfer_amount’ to draw a stacked column chart.

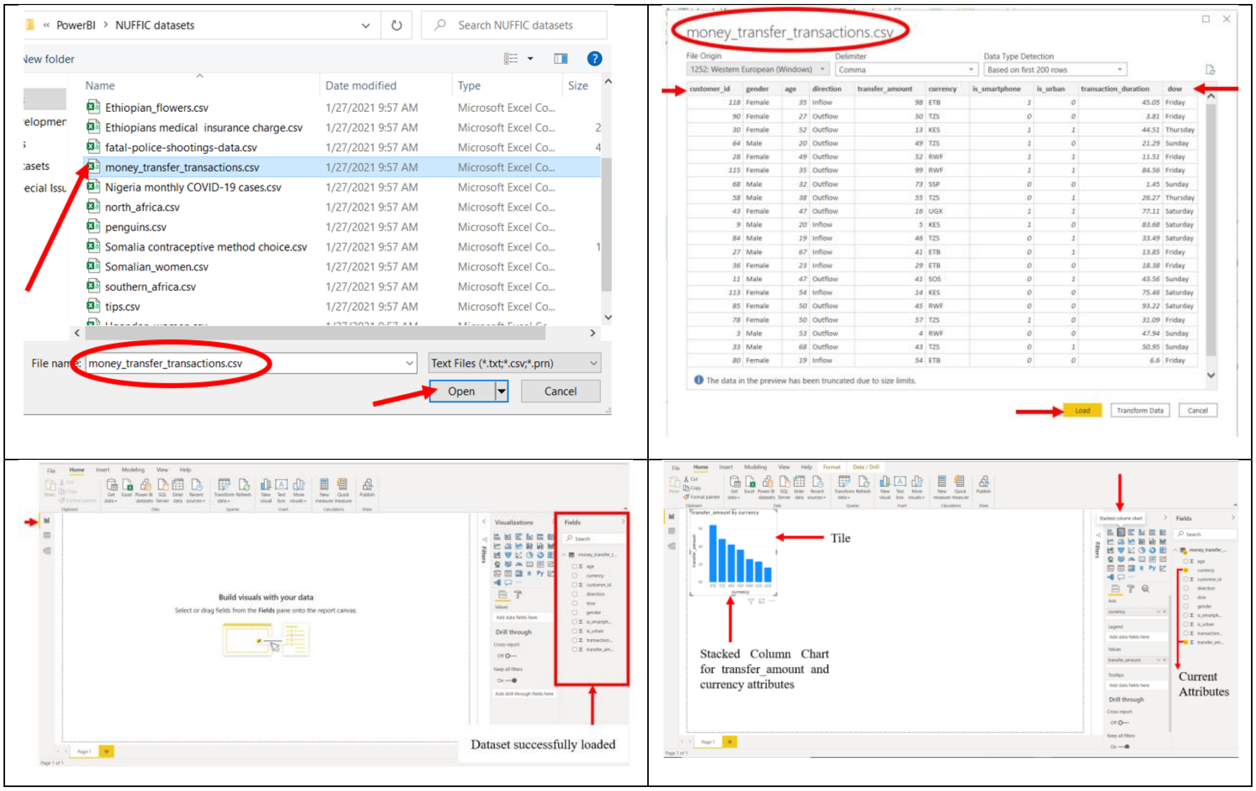
Answer

Please, follow the diagrams in the table to load the dataset and draw the stacked column chart.

1. The attributes are:
2. customer_id
3. gender age
4. direction
5. transfer_amount
6. currency
7. is_smartphone
8. is_urban
9. transaction_duration
10. day of week (dow)







PowerBI > NUFFIC datasets

Search NUFFIC datasets

Name	Date modified	Type
Ethiopian_flowers.csv	1/27/2021 9:57 AM	Microsoft Excel Co...
Ethiopians medical insurance charge.csv	1/27/2021 9:57 AM	Microsoft Excel Co...
fatal-police-shootings-data.csv	1/27/2021 9:57 AM	Microsoft Excel Co...
money_transfer_transactions.csv	1/27/2021 9:57 AM	Microsoft Excel Co...
Nigeria monthly COVID-19 cases.csv	1/27/2021 9:57 AM	Microsoft Excel Co...
north_africa.csv	1/27/2021 9:57 AM	Microsoft Excel Co...
penguins.csv	1/27/2021 9:57 AM	Microsoft Excel Co...
Somalia contraceptive method choice.csv	1/27/2021 9:57 AM	Microsoft Excel Co...
Somalian_women.csv	1/27/2021 9:57 AM	Microsoft Excel Co...
southern_africa.csv	1/27/2021 9:57 AM	Microsoft Excel Co...
tips.csv	1/27/2021 9:57 AM	Microsoft Excel Co...

File name: money_transfer_transactions.csv

Text Files (*.txt;*.csv;*.prn)

money_transfer_transactions.csv

File Origin: 1252: Western European (Windows)

Delimiter: Comma

Data Type Detection: Based on first 200 rows

customer_id	gender	age	direction	transfer_amount	currency	is_smartphone	is_urban	transaction_duration	dow
118	Female	35	Inflow	98	ETB	1	0	45.05	Friday
90	Female	27	Outflow	50	TZS	0	0	3.81	Friday
30	Female	52	Outflow	13	KES	1	1	44.51	Thursday
64	Male	20	Outflow	49	TZS	1	0	21.29	Sunday
28	Female	49	Outflow	52	RWF	1	1	11.51	Friday
115	Female	35	Outflow	99	RWF	1	1	84.56	Friday
68	Male	32	Outflow	73	SSP	0	0	1.45	Sunday
58	Male	38	Outflow	55	TZS	0	1	26.27	Thursday
43	Female	47	Outflow	16	UGX	1	1	77.11	Saturday
9	Male	20	Inflow	5	KES	1	0	83.68	Saturday
84	Male	19	Inflow	46	TZS	0	1	33.49	Saturday
27	Male	67	Inflow	41	ETB	0	1	13.85	Friday
36	Female	23	Inflow	29	ETB	0	0	18.38	Friday
11	Male	47	Outflow	41	SOS	0	1	43.56	Sunday
113	Female	54	Inflow	14	KES	0	0	75.46	Saturday
85	Female	50	Outflow	45	RWF	0	0	93.22	Saturday
78	Female	50	Outflow	57	TZS	1	0	31.09	Friday
3	Male	53	Outflow	4	RWF	0	0	47.94	Sunday
33	Male	68	Outflow	43	TZS	0	1	50.95	Sunday
80	Female	19	Inflow	54	ETB	0	0	6.6	Friday

The data in the preview has been truncated due to size limits.

Dataset successfully loaded

Stacked Column Chart for transfer_amount and currency attributes

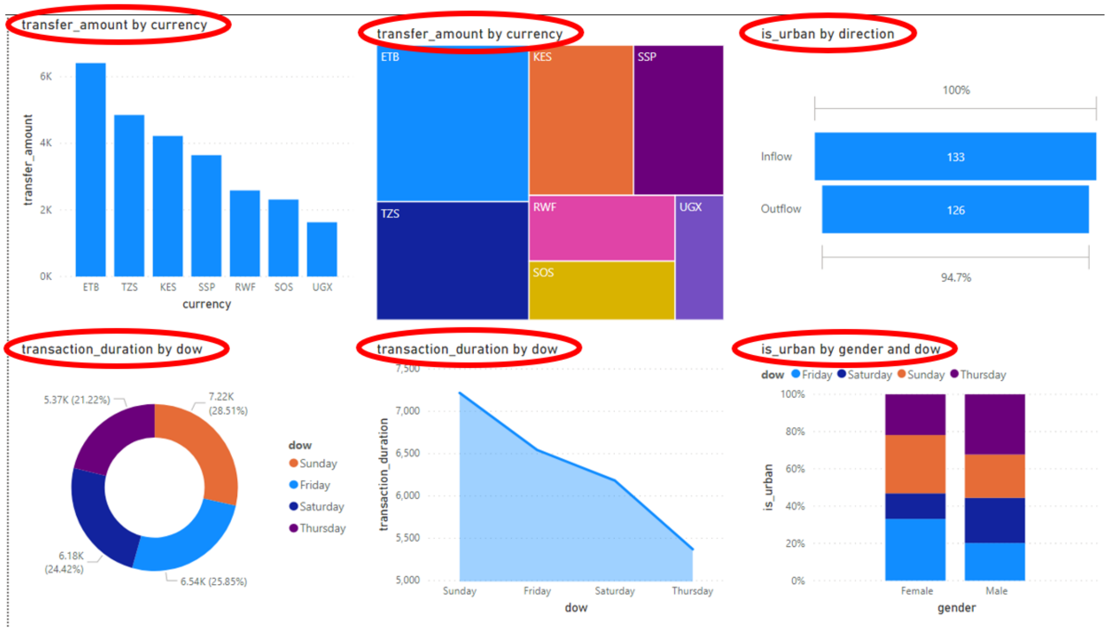
Current Attributes

Practice Exercise 2:

Use the different visualizations tools on ‘money transfer’ dataset for report

Answer

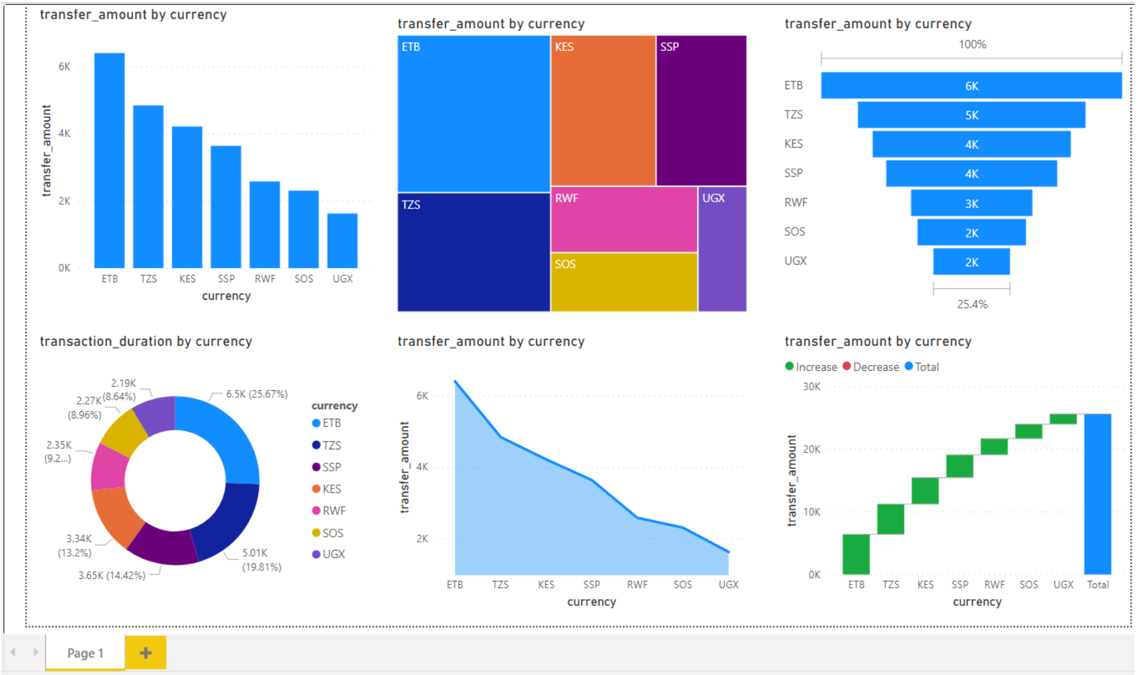
We have six tiles of different charts. The title of the charts is circled red. For example, the first chart by the upper-left is titled ‘transfer_amount by ‘currency’. It means that the two attributes ‘transfer_amount’ and ‘currency’ were placed against each other to see their impact. It is observed from the chart that ‘ETB’ currency was transferred most. Followed by ‘TZS’ while ‘UGX’’ was the least transferred. The last chart which is placed at the bottom-right corner was a stacked column chart by three attributes namely ‘is_urban’, ‘gender’ and ‘dow’.



Practice Exercise 3:

Represent attributes ‘tranfer_amount’ and ‘currency’ by different charts and name the charts

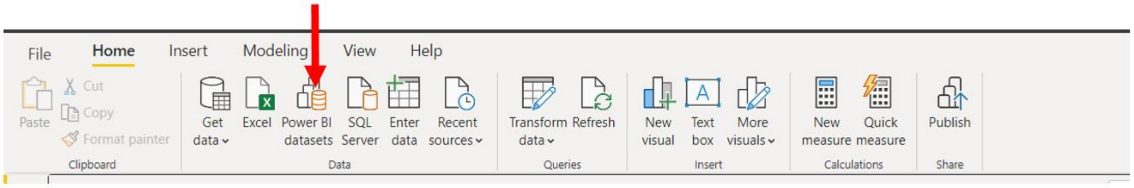
ANSWER: The six charts were stacked column chart, Donut chart, Treemap, Area chart, Funnel chart and waterfall chart.



TASK 3: Connect to 'Sales and Marketing Sample' dataset on Power BI dataset online. How many tables has the dataset? List them. How are the tables related? What is the cardinality of the tables?

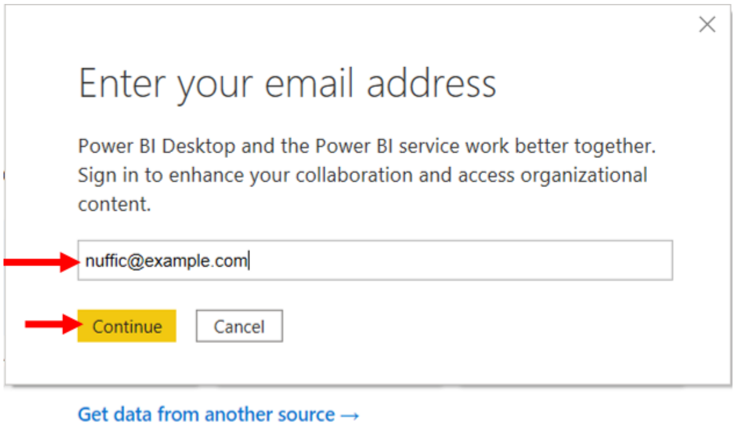
Answer:

Make sure you have internet connection. Click on the Power BI datasets' button



You enter your email address to sign in to Power BI service to connect to the dataset online.

NOTE: You can only have access to this dataset online. You cannot download

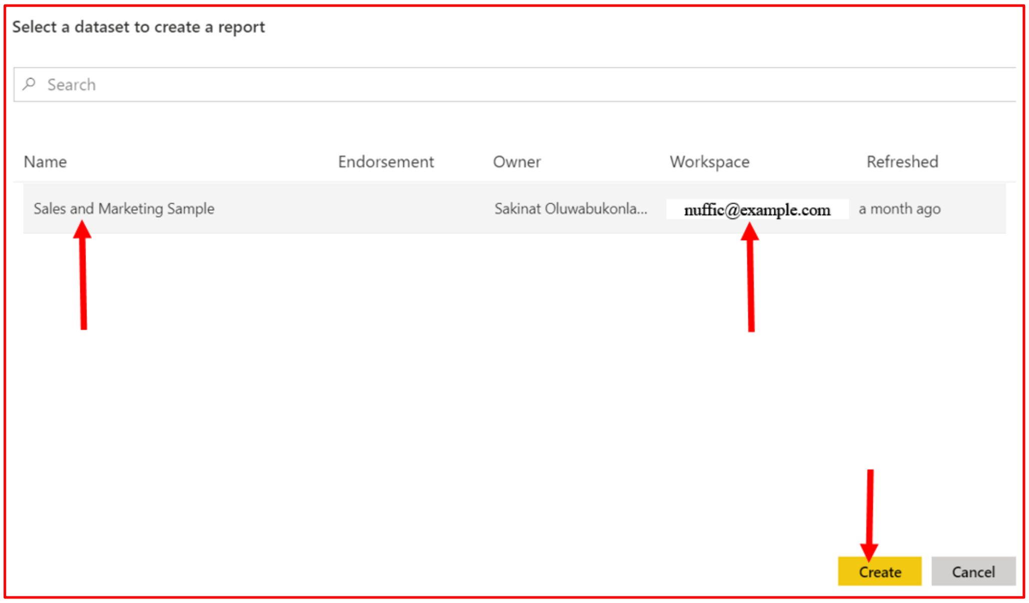


Enter your email address

Power BI Desktop and the Power BI service work better together. Sign in to enhance your collaboration and access organizational content.

nuffic@example.com

Confirm that the name of the dataset is what you seek, then click on the create button to create the dataset

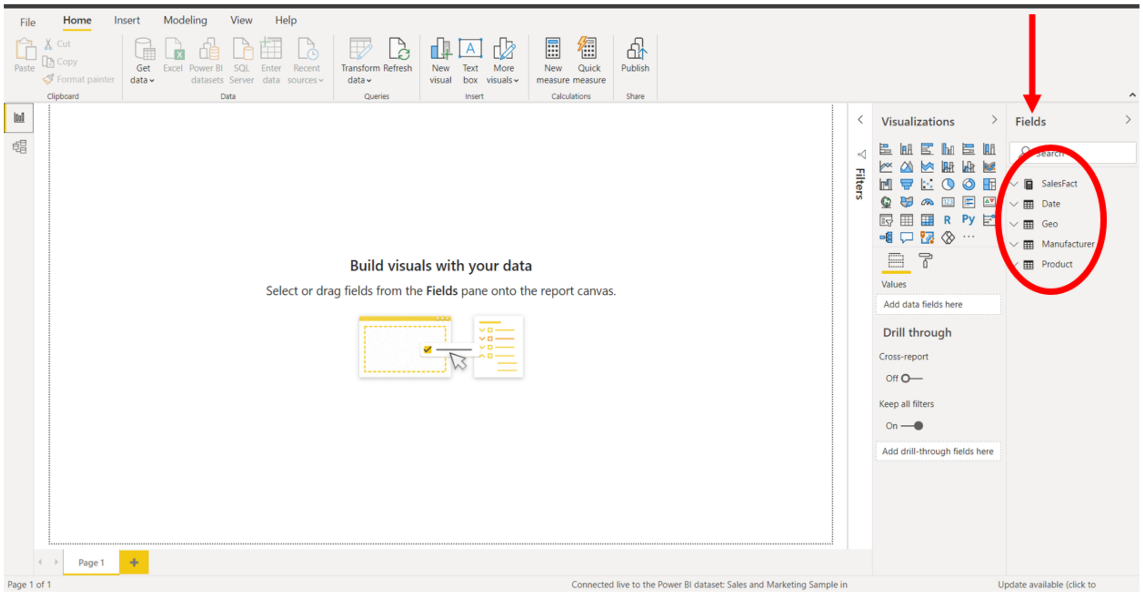


Select a dataset to create a report

Search

Name	Endorsement	Owner	Workspace	Refreshed
Sales and Marketing Sample		Sakinat Oluwabukonla...	nuffic@example.com	a month ago

Once the dataset is created, all tables in the dataset are shown on the ‘Field’ pane as shown in the diagram below



Build visuals with your data

Select or drag fields from the **Fields** pane onto the report canvas.

Fields

- SalesFact
- Date
- Geo
- Manufacturer
- Product

Page 1 of 1

Connected live to the Power BI dataset: Sales and Marketing Sample in

Update available (click to

Tables present in the datasets are

SalesFact

Date

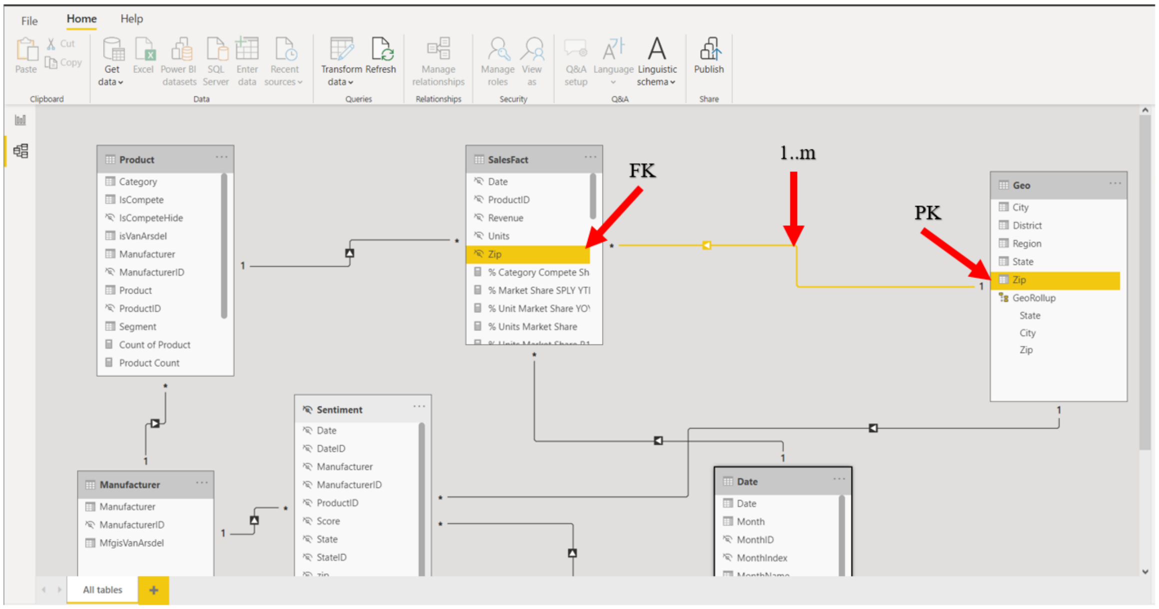
Geo

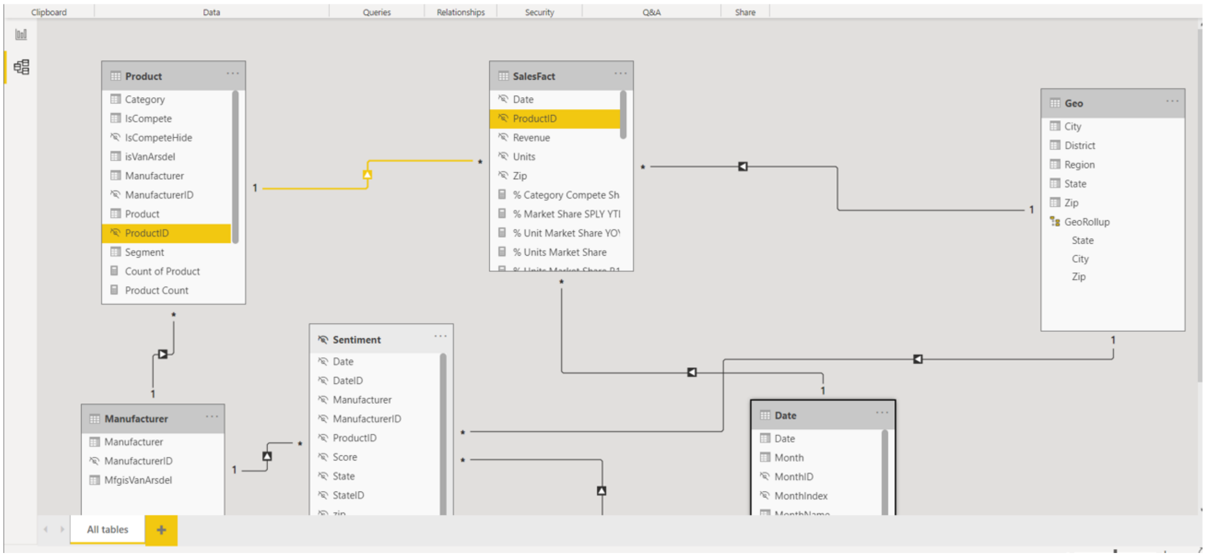
Manufacturer

Product

Sentiment

The tables are related by data modelling as shown by the diagram below. The six tables are connected by the yellow line with a one-to-many (1..m) cardinality relationship. All tables are connected by foreign keys (FK) using the crowfeet data modelling method.





Practice Exercise 4:

Using the money_transfer_transactions dataset, answer the following questions

(a) How many attributes has the dataset? List them and their data types

(b) How many tuples?

(c) How is the attribute data represented?

(d) Based on the dataset, prepare a comprehensive report for the Branch Manager

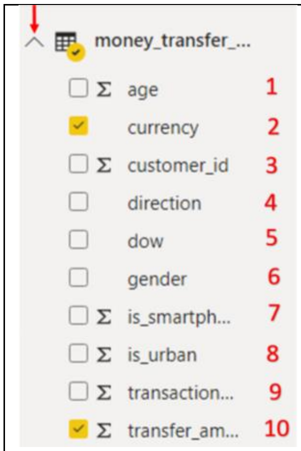
(e) Use a clustered chart to prepare the report for each transaction amount and dow. Use different colors to represent each day of the week

(f) Use the Treemap chat to represent the money transferred in different currencies. How many branches has the tree? Which branch/currency has the highest number of transactions? Which branch/currency has the lowest number of transaction?

(g) Use an area chart to report the units of quantity sold by each country. Find the average line and 75th percentile.

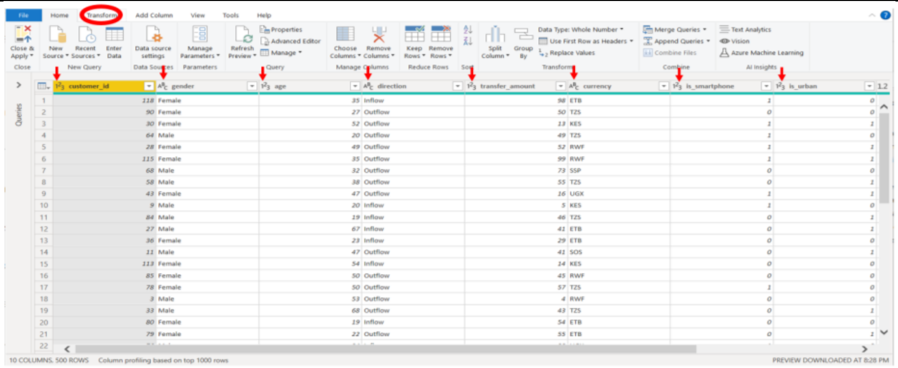
Answer:

(a) Click on the Λ button to expand the dataset. There are 10 attributes in the dataset



money_transfer_...

□	Σ	age	1
☑		currency	2
□	Σ	customer_id	3
□		direction	4
□		dow	5
□		gender	6
□	Σ	is_smartph...	7
□	Σ	is_urban	8
□	Σ	transaction...	9
☑	Σ	transfer_am...	10

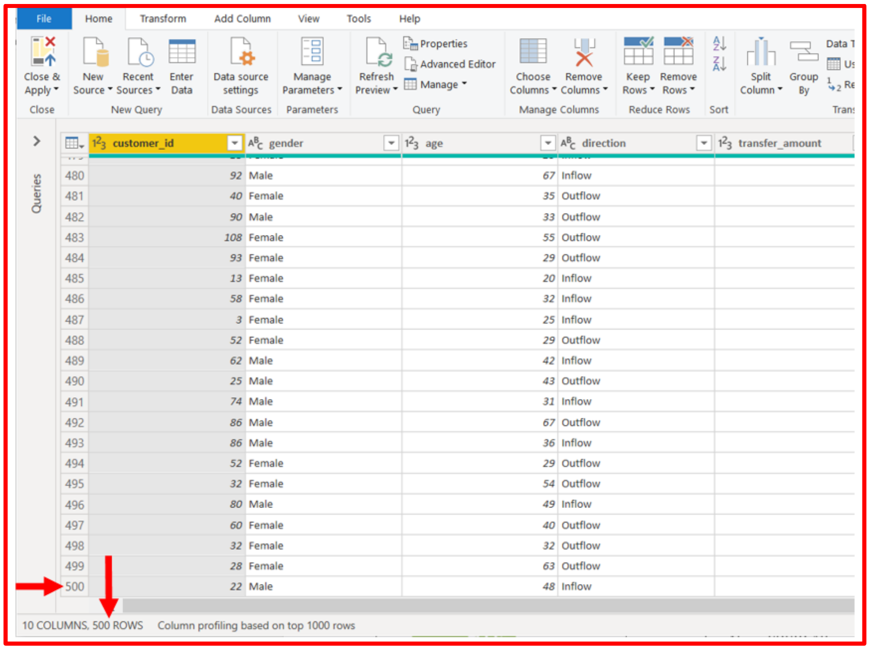


	customer_id	gender	age	direction	transfer_amount	currency	is_smartphone	is_urban
1	118	Female	35	Inflow	98	ETB	1	0
2	90	Female	27	Outflow	50	TZS	0	0
3	30	Female	52	Outflow	13	KES	1	1
4	64	Male	20	Outflow	49	TZS	1	0
5	28	Female	49	Outflow	52	RWF	1	1
6	115	Female	35	Outflow	99	RWF	1	1
7	68	Male	32	Outflow	73	SSP	0	0
8	58	Male	38	Outflow	55	TZS	0	1
9	43	Female	47	Outflow	16	UGX	1	1
10	9	Male	20	Inflow	5	KES	1	0
11	84	Male	19	Inflow	46	TZS	0	1
12	27	Male	67	Inflow	41	ETB	0	1
13	36	Female	23	Inflow	29	ETB	0	0
14	11	Male	47	Outflow	41	SOS	0	1
15	113	Female	54	Inflow	14	KES	0	0
16	85	Female	50	Outflow	45	RWF	0	0
17	78	Female	50	Outflow	57	TZS	1	0
18	3	Male	53	Outflow	4	RWF	0	0
19	33	Male	68	Outflow	43	TZS	0	1
20	80	Female	19	Inflow	54	ETB	0	0
21	79	Female	22	Outflow	55	ETB	0	1
22								

10 COLUMNS, 500 ROWS   Column profiling based on top 1000 rows   PREVIEW DOWNLOADED AT 8:28 PM

Click on the ‘Transform’ button to show the details on the dataset. If the icon beside the attribute’s name (The red arrow) is ‘123’, then the datatype is numeric. But if it is ‘ABC’, it is text.

(b) There are 500 tuples in the dataset. This can be seen where the red arrow points to ’10 COLUMNS 500 ROWS’ as tuples are represented by rows.

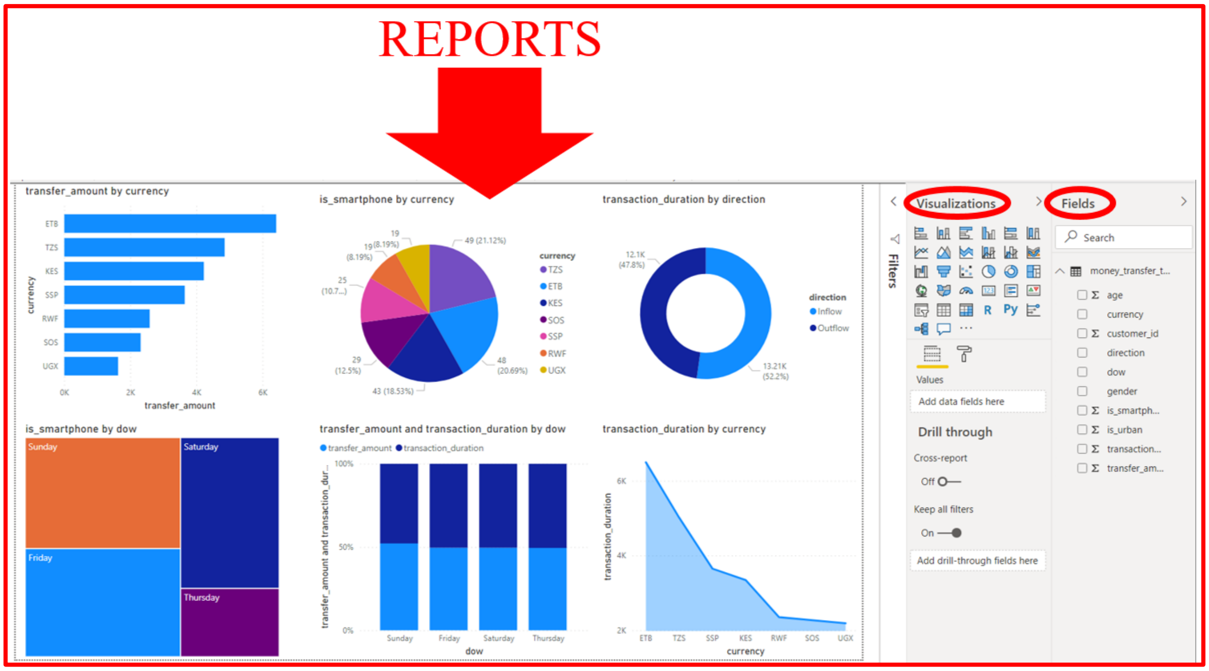


	customer_id	gender	age	direction	transfer_amount
480	92	Male	67	Inflow	
481	40	Female	35	Outflow	
482	90	Male	33	Outflow	
483	108	Female	55	Outflow	
484	93	Female	29	Outflow	
485	13	Female	20	Inflow	
486	58	Female	32	Inflow	
487	3	Female	25	Inflow	
488	52	Female	29	Outflow	
489	62	Male	42	Inflow	
490	25	Male	43	Outflow	
491	74	Male	31	Inflow	
492	86	Male	67	Outflow	
493	86	Male	36	Inflow	
494	52	Female	29	Outflow	
495	32	Female	54	Outflow	
496	80	Male	49	Inflow	
497	60	Female	40	Outflow	
498	32	Female	32	Outflow	
499	28	Female	63	Outflow	
500	22	Male	48	Inflow	

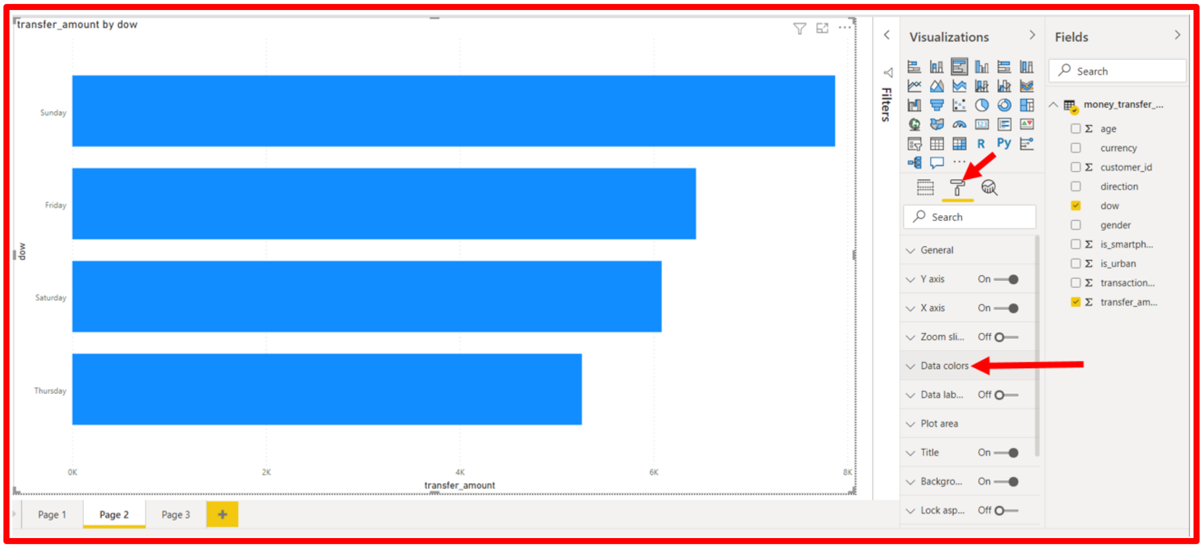
10 COLUMNS, 500 ROWS   Column profiling based on top 1000 rows

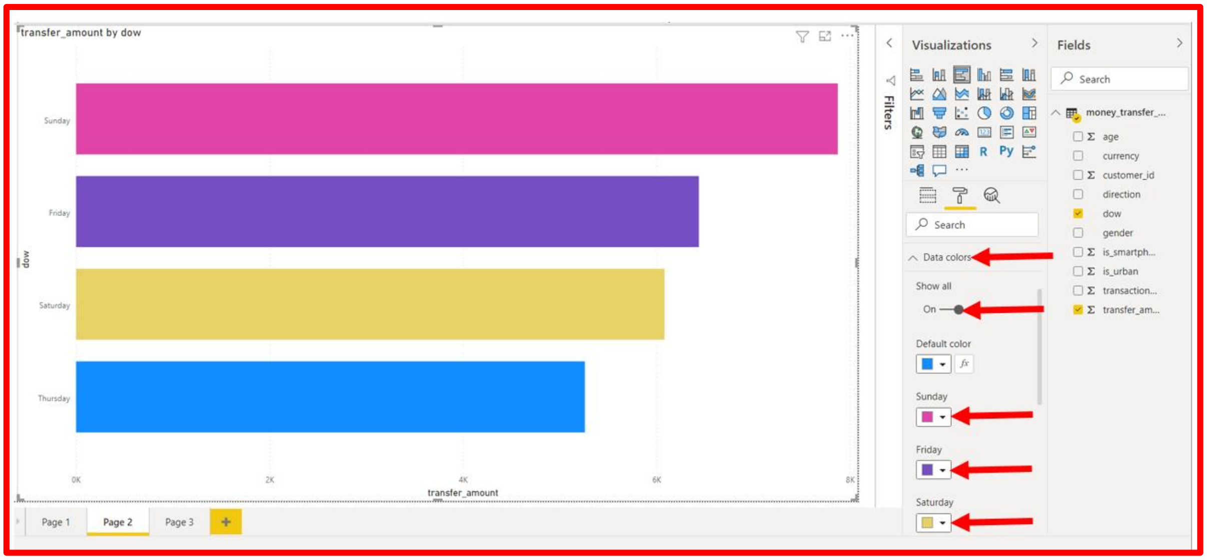
(c) The attributes are represented by columns in the relational table

(d) The reports are created using the visualization pane

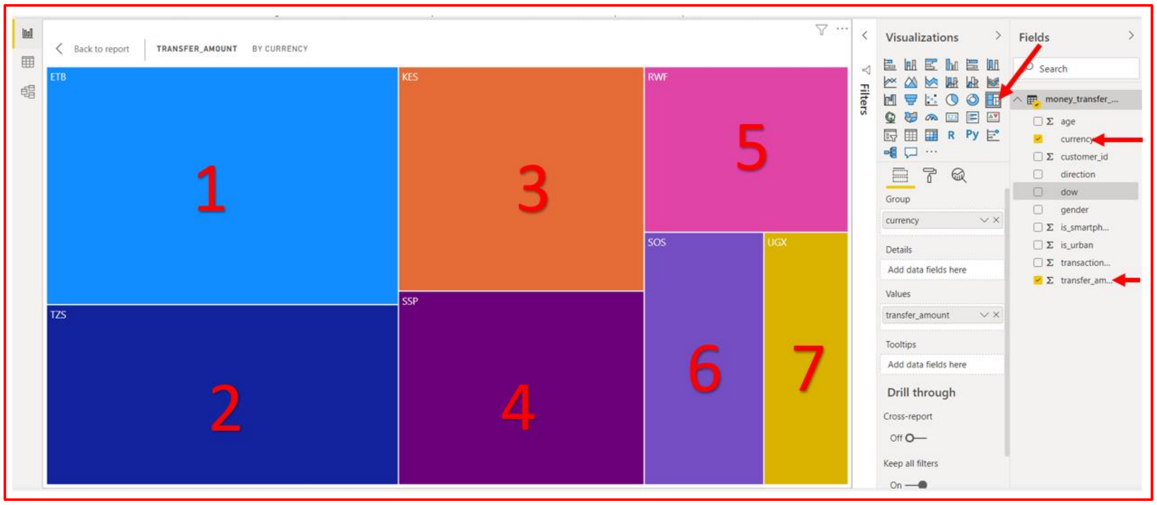


(e)





(f)

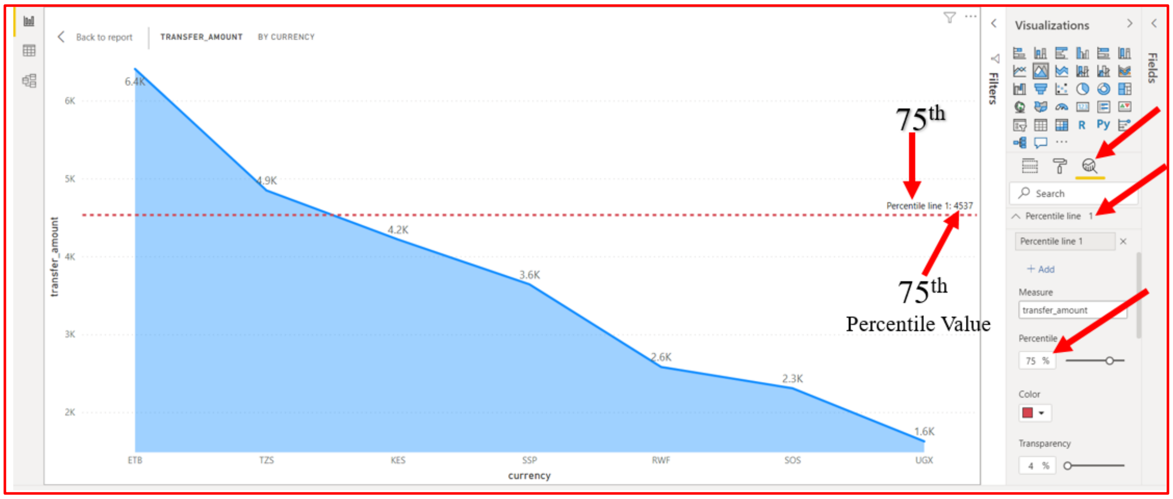


There are seven (7) branches (currencies) in the tree.

ETB has the highest number of transactions because it has the biggest size.

UGX has the lowest number of transactions because it has the biggest size.

(g) Please, note all the arrow heads. The red-dashed line shows the 75th percentile of the money tranfer.



Percentile line 1

Percentile line 1

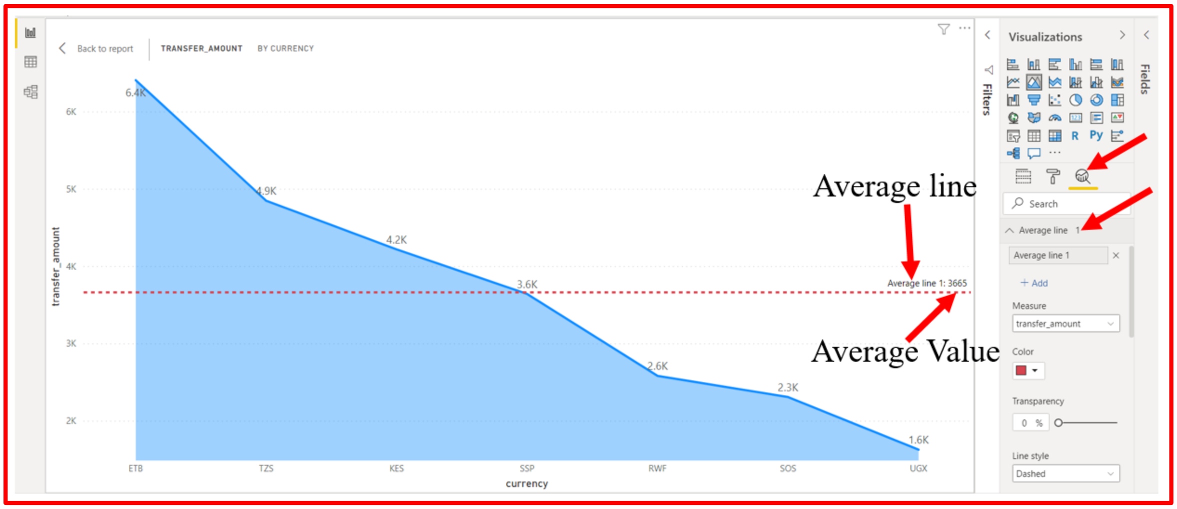
+ Add

Measure: transfer_amount

Percentile: 75 %

Color

Transparency: 4 %



Average line 1

Average line 1

+ Add

Measure: transfer_amount

Color

Transparency: 0 %

Line style: Dashed

 Learning Activity	Create simple reports with the following datsets a. Financial Sample b. Africa COVID-19 c. Fatal-police-shootings-data

Assessments – TMA

Tutor Marked Assignment I

Using the Financial Sample dataset, answer the following questions

(a) How many attributes has the dataset? List them and their data types

(b) How many tuples?

(c) How is the attribute data represented?

(d) Based on the dataset, prepare a comprehensive report for the CEO of the company

(e) Use a clustered chart to prepare the report for each country and the sales made. Use different colours to represent each country

(f) Use the Treemap chat to represent the profits made by each country and fine the minimum and maximum profits

(g) Use an area chart to report the units of quantity sold by each country. Find the average line, 75th, 90th and 21st percentiles.

(h) What is the maximum of unit sold by segment of the country

(i) Describe the profit made by each segment of each country

Self Review Questions and Answers

1. What is the name of pane to see an uploaded data?

Answer: Field

2. Name the three views present in Power BI.

Answer: Model, Data and Report

3. What type of modeling notation is adopted by Power BI?

Answer: E-R crow's feet

4. What does the percentile in the Analytics of Power BI does?

Answer: It is a measure of central tendency that divides data into 100%.

5. How can you find the minimum and maximum values of a dataset?

Answer: Through the min() and max() in the analytics pane.

6. Mention the relationship in the data model presented by Power BI?

Answer: 1-1, 1-*, *-1 and *-*

7. How can you detect the number and type of attributes and tuples present in a dataset?

Answer: Through the 'transform data' button present in the home menu

8. How can you represent each bar in the clustered column with different colors and labels?

Answer: Through the 'format' pane, use data colors and Data labels

9. How many tiles are present by default in the report pane of Power BI?

Answer: 8

10. Multidimensional view of data attributes is possible in POWERBI? True/False.

Answer: True

11. How can I change the attribute's datatype. Eg from character to integer?

Answer: Click on the attribute from the 'transform data' mode

12. How do I sort data based on an attribute?

Answer: Click on the attribute with the integer or float or numeric datatype and the ⇧ arrow in the 'transform data' mode. Choose either ascending or descending order

13. Mention 5 different data file format present in Power BI.

Answer: Excel, CSV, XML, JSON, PDF, and SharePoint formats and databases such as SQL, Access, SQL Server Analysis Services, Oracle, IBM, MySQL, and much more.

14. How to import the data in Power BI desktop?

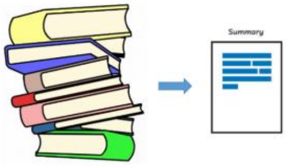
Answer: Go to getting data Sources and click on your required sources (Excel, SQL, CSV) then Load it. Click on the Data view to view that data. To choose the table click on the fields and you can pick a visualization to generate a report.

References

1. SQL Analytics: the Foundation of Business Intelligence l Sisense
2. SQL Tutorial - w3resource
3. Basic queries (lgatto.github.io)
4. SQL Exercises, Practice, Solution - w3resource
5. Codd, E. F. (1970). "A Relational Model of Data for Large Shared Data Banks". *Communications of the ACM*. **13** (6): 377–387. doi:10.1145/362384.362685.
6. Ramakrishan, R., Gehrke, J. “Database Management Systems” Second Edition. McGraw Hill Higher Education.
7. https://mindmajix.com/power-bi-visualization-types
8. Dedić, N., & Stanier, C. (2016). Measuring the success of changes to existing business intelligence solutions to improve business intelligence reporting. In International conference on research and practical issues of enterprise information systems (pp. 225-236). Springer, Cham.
9. Rud, O. P. (2009). Business intelligence success factors: tools for aligning your business in the global economy (Vol. 18). John Wiley & Sons.
10. https://www.sisense.com/glossary/descriptive-analytics/
11. https://www.dataversity.net/dia-webinar-descriptive-prescriptive-predictive-analytics/
12. https://www.ibm.com/cloud/learn/data-warehouse
13. https://datawarehouseinfo.com/data-warehouse-star-schema-vs-snowflake-schema/
14. Runtuwene, J. P. A., Tangkawarow, I. R. H. T., Manoppo, C. T. M., Salak, R. J. (2018). A Comparative Analysis of Extract, Transformation and Loading (ETL) Process. IOP Conf. Series: Materials Science and Engineering. 306: 012066 doi:10.1088/1757-899X/306/1/012066
15. https://olap.com/learn-bi-olap/olap-bi-definitions/business-intelligence/
16. https://olap.com/olap-definition/

17. https://www.guru99.com/online-analytical-processing.html
18. https://forums.bsdinsight.com/threads/descriptive-predictive-and-prescriptive-analytics-explained.41558/
19. https://optilon.com/optilon-academy/advanced-analytics/
20. https://www.izenda.com/business-intelligence-reporting-tools/#:~:text=What%20are%20Business%20Intelligence%20Reporting,actionable%20insights%20into%20business%20trends.
21. https://www.datapine.com/blog/business-intelligence-reporting/
22. https://www.tableau.com/learn/articles/business-intelligence/choosing-bi-platforms
23. https://towardsdatascience.com/best-7-business-intelligence-tools-2020-round-one-fight-3afd4185fd59

Summary of Study Unit 4



SUMMARY

- BI is a set of processes, architectures, and technologies that convert raw data into meaningful information that drives profitable business actions.
- BI systems help businesses to identify market trends and spot business problems that need to be addressed.
- BI technology can be used by Data analyst, IT people, business users and head of the company.
- BI system helps organization to improve visibility, productivity and fix accountability

The draw-backs of BI is that it is time-consuming costly and very complex process.

- Data modeling is the process of developing data model for the data to be stored in a Database.
- Data Models ensure consistency in naming conventions, default values, semantics, security while ensuring quality of the data.
- Data Model structure helps to define the relational tables, primary and foreign keys and stored procedures.
- There are three types of conceptual, logical, and physical.
- The main aim of conceptual model is to establish the entities, their attributes, and their relationships.
- Logical data model defines the structure of the data elements and set the relationships between them.
- A Physical Data Model describes the database specific implementation of the data model.
- The main goal of a designing data model is to make certain that data objects offered by the functional team are represented accurately.
- The biggest drawback is that even smaller change made in structure require modification in the entire application.

Reading this Data Modeling tutorial, you will learn from the basic concepts such as What is Data Model? Introduction to different types of Data Model, advantages, disadvantages, and data model example

OLAP is a technology that enables analysts to extract and view business data from different points of view.
At the core of the OLAP concept, is an OLAP Cube.
Various business applications and other data operations require the use of OLAP Cube.
There are primary five types of analytical operations in OLAP 1) Roll-up 2) Drill-down 3) Slice 4) Dice and 5) Pivot