

The background of the slide is a complex, abstract pattern in shades of teal and dark green. It features a dense, fractal-like structure with many fine, curved lines that create a sense of depth and movement, resembling a microscopic view of a mineral or a complex data visualization.

# Introduction

DAB 203

Business Analytics and Decision Making



## BOOK LINK

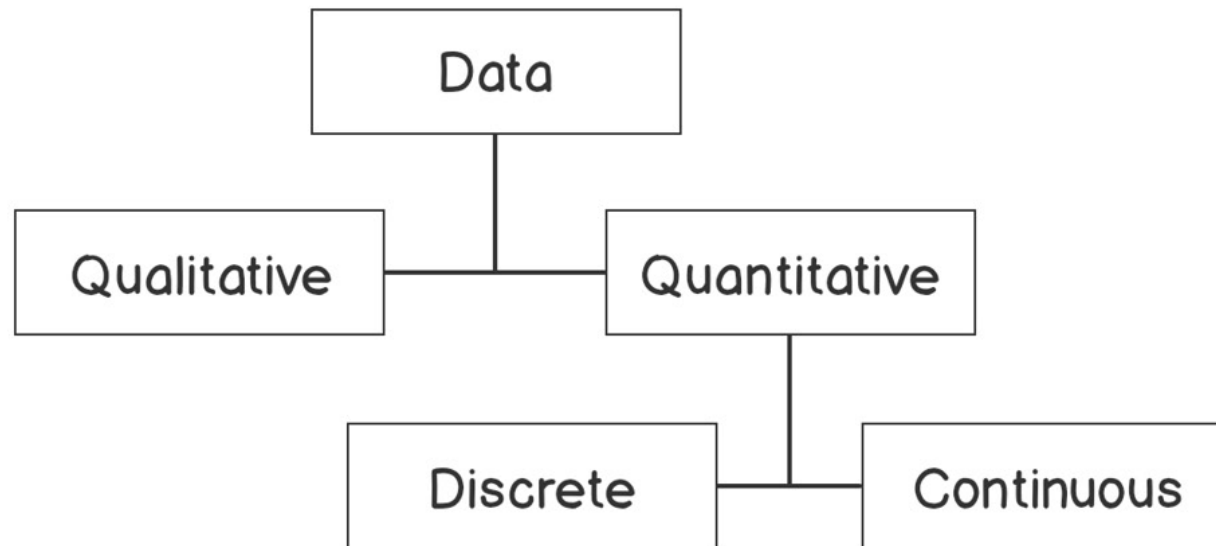
<https://nibmehub.com/opac-service/pdf/read/Data%20Analysis%20Using%20SQL%20and%20Excel-%202nd%20Edition.pdf>

# SQL

1. A mature and standardized language for accessing data
2. Multiple vendors including open source
3. Scalability over a very broad range of hardware
4. **Part of the SQL language used for data analysis purposes is the SELECT statement. The rest of the language is about getting the data into the database which is not the focus of this course.**
5. We will focus on what can we do with the data instead of how to do it

# What is Data?

---



# RELATIONAL DATABASES

- Relational databases, which were invented in the 1970s, are now the storehouse of mountains of data available to businesses.
- A relational database organizes data into tables which can be linked—or related—based on data common to each. This capability enables you to retrieve an entirely new table from data in one or more tables with a single query. It also allows you and your business to better understand the relationships among all available data and gain new insights for making better decisions or identifying new opportunities.

# Consolidated customer statement

Customer Table

Cust ID	Cust Name	Cust Address

Transaction Table


Trans date	Cust ID	Trans Amt	Payment Method



The popularity of relational databases rests on **ACID** properties of transactions:

Atomicity (transaction is completely executed or nothing is executed)

Consistency (Only valid data gets into the databases using business rules)



Isolation (If the transaction is not complete, data won't be visible to other processes until it is complete)

Durability (once the transaction is complete, the data should be retrieved irrespective of hardware failure)

# What is Big Data?

## — Data Is Exploding

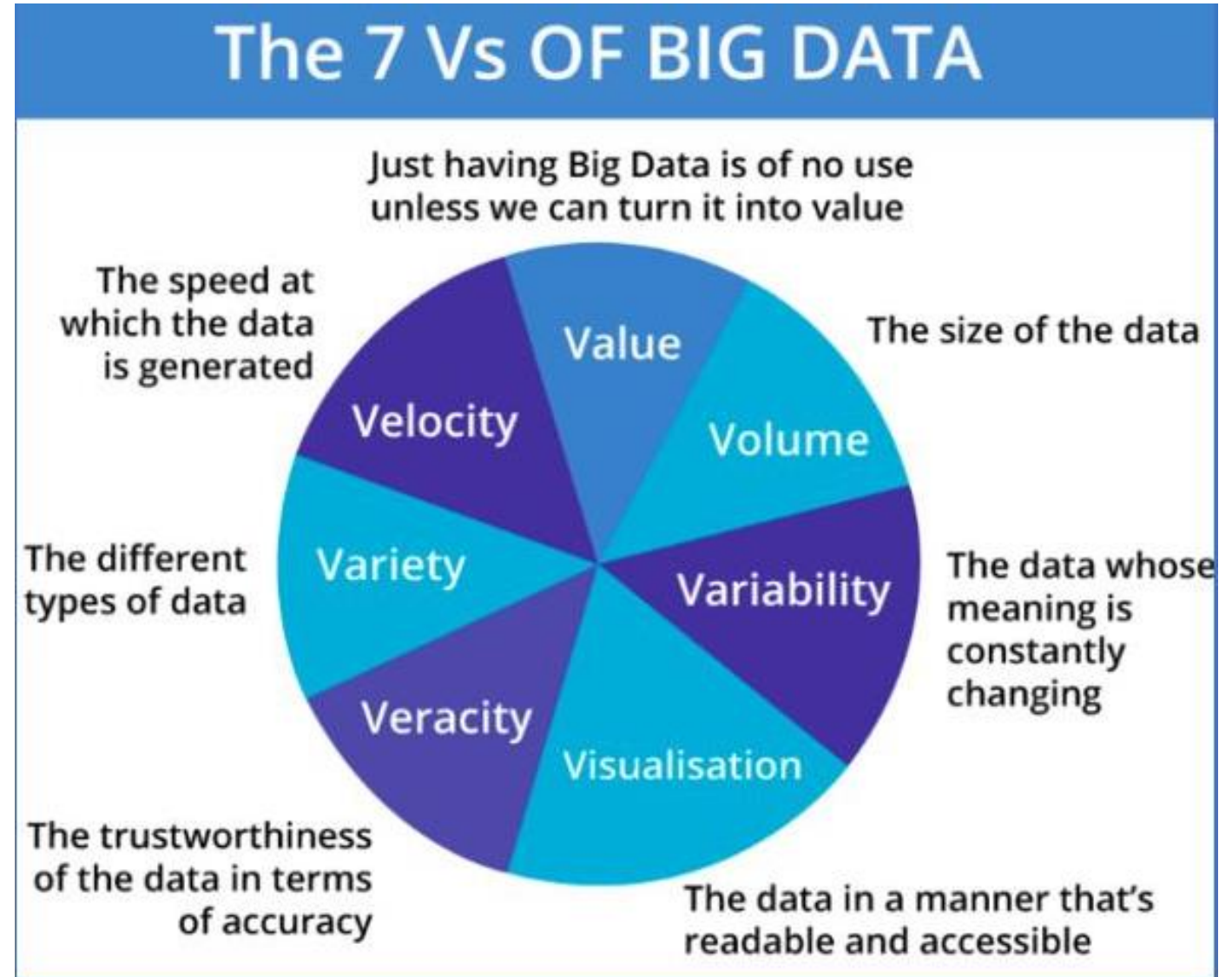
- Big data refers to the large volume of structured and unstructured data.
- The analysis of big data leads to better insights for business.
- By 2025, experts indicate that over 463 exabytes of data will be created each day, the equivalent of around 212,765,957 DVDs.
- Poor data quality can cost the US economy as much as \$3.1 trillion per year.
- The big data analytics market will reach a value of around \$103 billion by 2027
- 97.2% of organizations say they're now investing in AI and big data
- Around 95% of companies say their inability to understand and manage unstructured data is holding them back



# Big Data: Seven V's:

According to the conventional definition, for data to be regarded as “big”, it should possess a number of key attributes – volume, velocity and variety

- ✓ **Volume**
- ✓ **Velocity**
- ✓ **Variety**
- ✓ **Variability**
- ✓ **Veracity**
- ✓ **Visualization**
- ✓ **Value**



# FUTURE OF AI AND BIG DATA

- <https://www.youtube.com/watch?v=uD8Dbozzod4>

# Hadoop

- **Hadoop is a framework that allows distributed processing of large datasets across clusters of computers using simple programming models.**
- Doug Cutting discovered Hadoop and named it after his son's yellow toy elephant.



<https://www.youtube.com/watch?v=aReuLtY0YMI>

## Why Hadoop?

### Challenges of Distributed Systems:

1. High Chances of system Failure
2. Limited Bandwidth
3. High Programming Complexity

**HADOOP is used to overcome these challenges!**



# Hadoop and Hive

- Hadoop is a platform for processing large amounts of data, particularly big data.
- Organizations with traditional data warehouses are based on SQL
- **HIVE provides SQL like intellect so the queries can interact with Hadoop systems**
- Users write SQL like queries that HIVE converts to MapReduce to query against Hadoop Databases

Hadoop	Hive
<b>Hadoop</b> is a framework to process/query the Big data	<b>Hive</b> is an SQL Based tool that builds over Hadoop to process the data.
<b>Hadoop</b> can understand Map Reduce only.	<b>Hive</b> process/query all the data using HQL (Hive Query Language) it's SQL-Like Language
Map Reduce is an integral part of <b>Hadoop</b>	<b>Hive's</b> query first get converted into Map Reduce than processed by Hadoop to query the data.

# Traditional Database Systems vs. Hadoop

Traditional Database Systems	Hadoop
Data is stored in a central location and sent to the processor at run time.	In Hadoop, the program goes to the data. It initially distributes the data to multiple systems and later runs the computation wherever the data is located.
Traditional Database Systems cannot be used to process and store a large amount of data (big data).	Hadoop works better when the data size is big. It can process and store a large amount of data easily and effectively.
Traditional RDBMS is used to manage only structured and semi-structured data. It cannot be used to manage unstructured data.	Hadoop has the ability to process and store a variety of data, whether it is structured or unstructured.

# NoSQL and Other Types of Databases

- Not only SQL (in between the unstructured HADOOP and structured traditional RDBMS)
  - Key-value pairs (web environment for managing data about online sessions)
  - Graph-based databases
  - Document databases
  - Geographic information systems
- These databases are complementary to the traditional databases

# What is a Data Model?

- The definitions of the tables, the columns, and the relationships among them constitute a data model for the database.
- Database has two data models
  - Logical data model (explains in terms that business users can understand)
  - Physical data model (explains how the database is actually implemented).



# What is a Table?

- A set of rows and columns that describe multiple instances of something (entity).
- Can have any number of rows and columns
- Each row represents one instance
- Each column contains one attribute for one instance
- Columns can take null values (not available or unknown)
- Metadata about what the values in column mean
- Each column in a database should have a datatype and a flag indicating whether NULL values are allowed

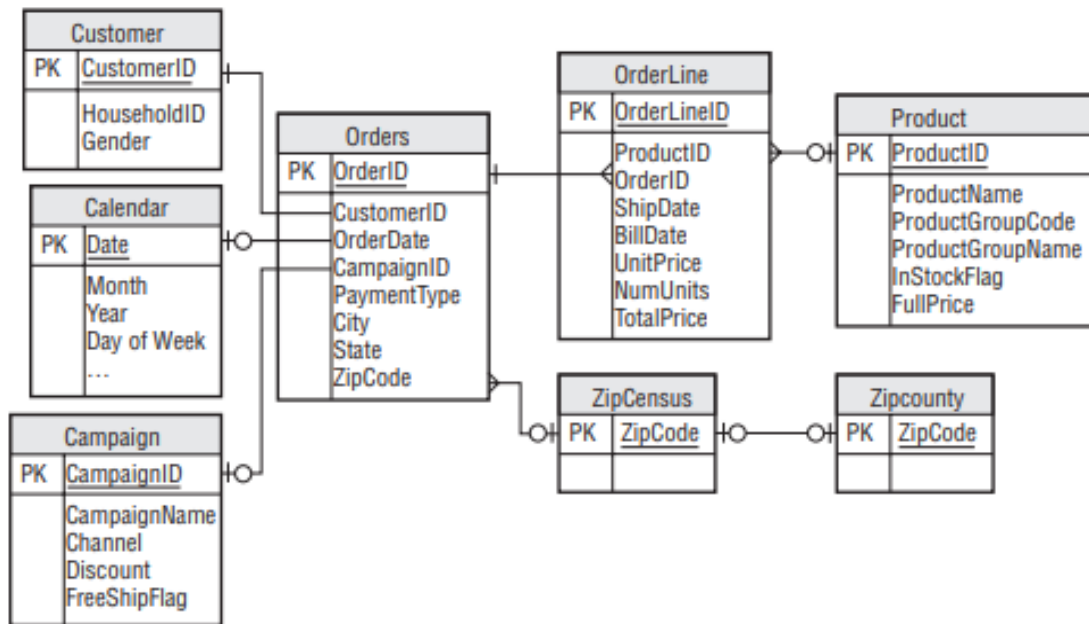
# Column Types

- Basically how the value in the particular column will be stored
- Primary Key columns uniquely identify each row in the table. No two rows can have the same value for the primary key and the primary is never NULL
- Numeric values support arithmetic and other mathematical operations (In SQL floating point numbers, integers and decimals)
- Dates and date-times
- Character string data

# Entity Relationship Diagram

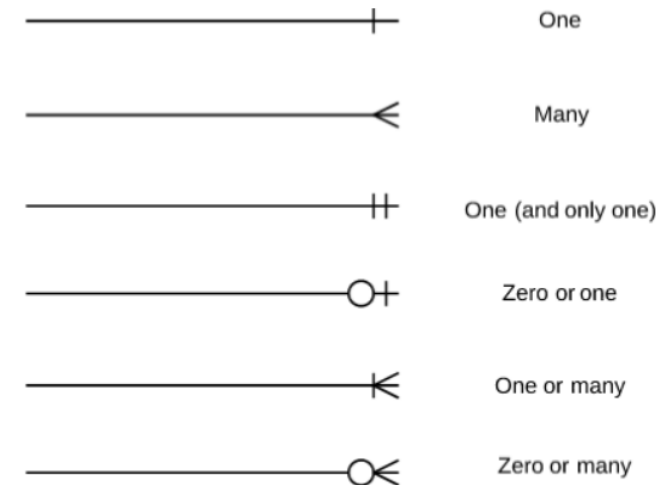
- Entity (Table)
  - Something that the users want to track (e.g. Order, Customer, Student, Item)
  - They have attributes (columns) that describe the characteristics (e.g. Order will have OrderId, OrderDate, tax, SubTotal)
  - They have an identifier (an attribute) whose value is associated with one and only one entity instance (e.g. OrderId) like a primary key
- Relationships
  - Entities have relationships to each other (e.g. Department, Supervisor, Student)
  - The relationship is through keys (column names) so common column in both the tables
  - Relationships have cardinality (no. of items related on each side)
    - 1:M (one to many)
    - M:M (many to many)

# Entity Relationship Diagram: Example



**Figure 1-1:** This entity-relationship diagram shows the relationship among entities in the purchase dataset. Each entity corresponds to one table.

Cardinality and ordinality are shown by the styling of a line and its endpoint, according to the chosen notation style.





# Primary and Foreign Key

- A foreign key is just a column whose contents are the primary key of another table (e.g. ZIPCODE in Orders is a foreign key; ZIPCODE in Zipcensus is a primary key)