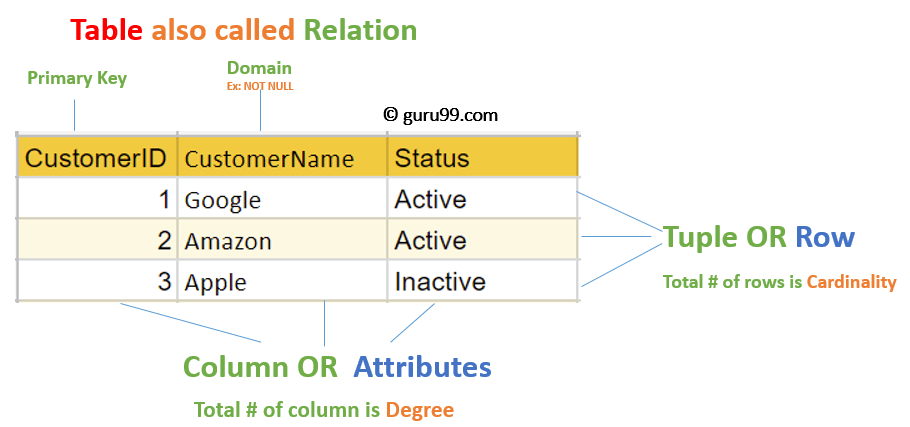
Types of Database

1. **Relational Model (RM)** represents the database as a collection of relations. A relation is nothing but a table of values. Every row in the table represents a collection of related data values. These rows in the table denote a real-world entity or relationship.

The table name and column names are helpful to interpret the meaning of values in each row. The data are represented as a set of relations. In the relational model, data are stored as tables. However, the physical storage of the data is independent of the way the data are logically organized.

## Relational Model Concepts

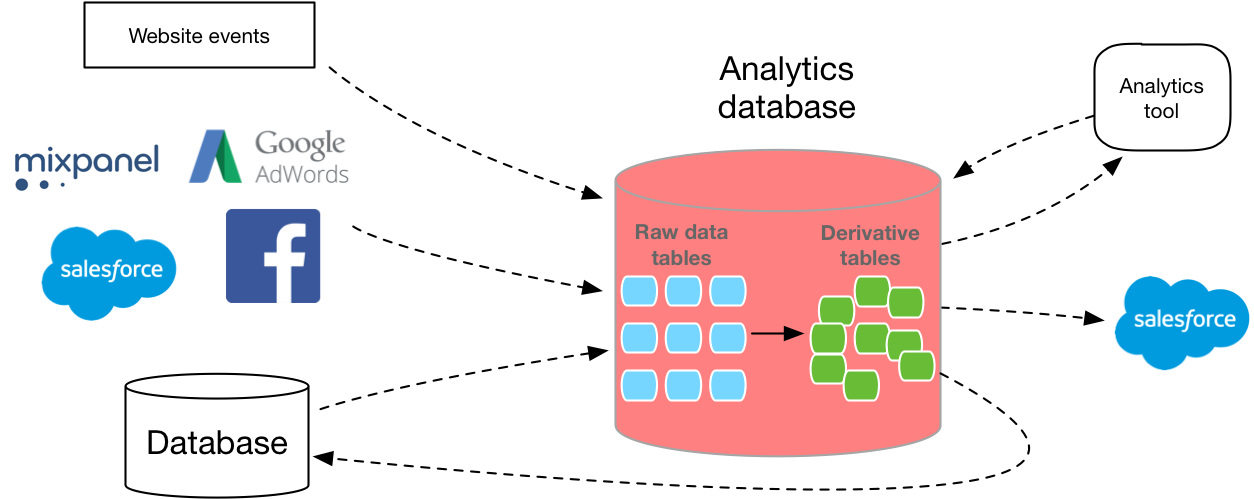
1. **Attribute:** Each column in a Table. Attributes are the properties which define a relation. e.g., Student\_Rollno, NAME,etc.
2. **Tables** – In the Relational model the, relations are saved in the table format. It is stored along with its entities. A table has two properties rows and columns. Rows represent records and columns represent attributes.
3. **Tuple** – It is nothing but a single row of a table, which contains a single record.
4. **Relation Schema:** A relation schema represents the name of the relation with its attributes.
5. **Degree:** The total number of attributes which in the relation is called the degree of the relation.
6. **Cardinality:**Total number of rows present in the Table.
7. **Column:** The column represents the set of values for a specific attribute.
8. **Relation instance** – Relation instance is a finite set of tuples in the RDBMS system. Relation instances never have duplicate tuples.
9. **Relation key** – Every row has one, two or multiple attributes, which is called relation key.
10. **Attribute domain** – Every attribute has some pre-defined value and scope which is known as attribute domain



1. Analytical Model

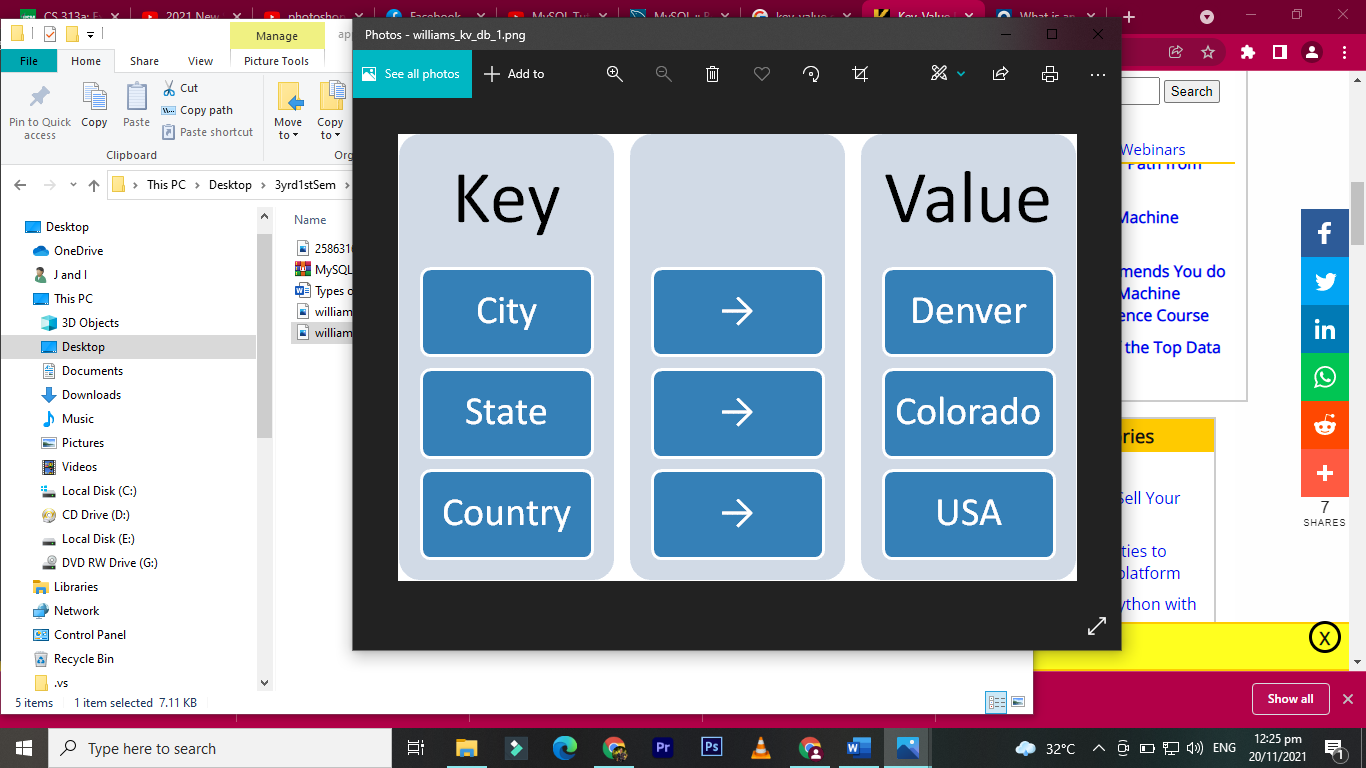
Analytical database software specializes in big data management for business applications and services. Analytical databases are optimized to provide quick query response times and advanced analytics. They are also more scalable than traditional databases and often times are columnar databases that can efficiently write and read data to and from hard disk storage in order to speed up the time it takes to return a query. Analytical database features include column-based storage, in-memory loading of compressed data and the ability to search data through multiple attributes.

Examples of Analytical Databases

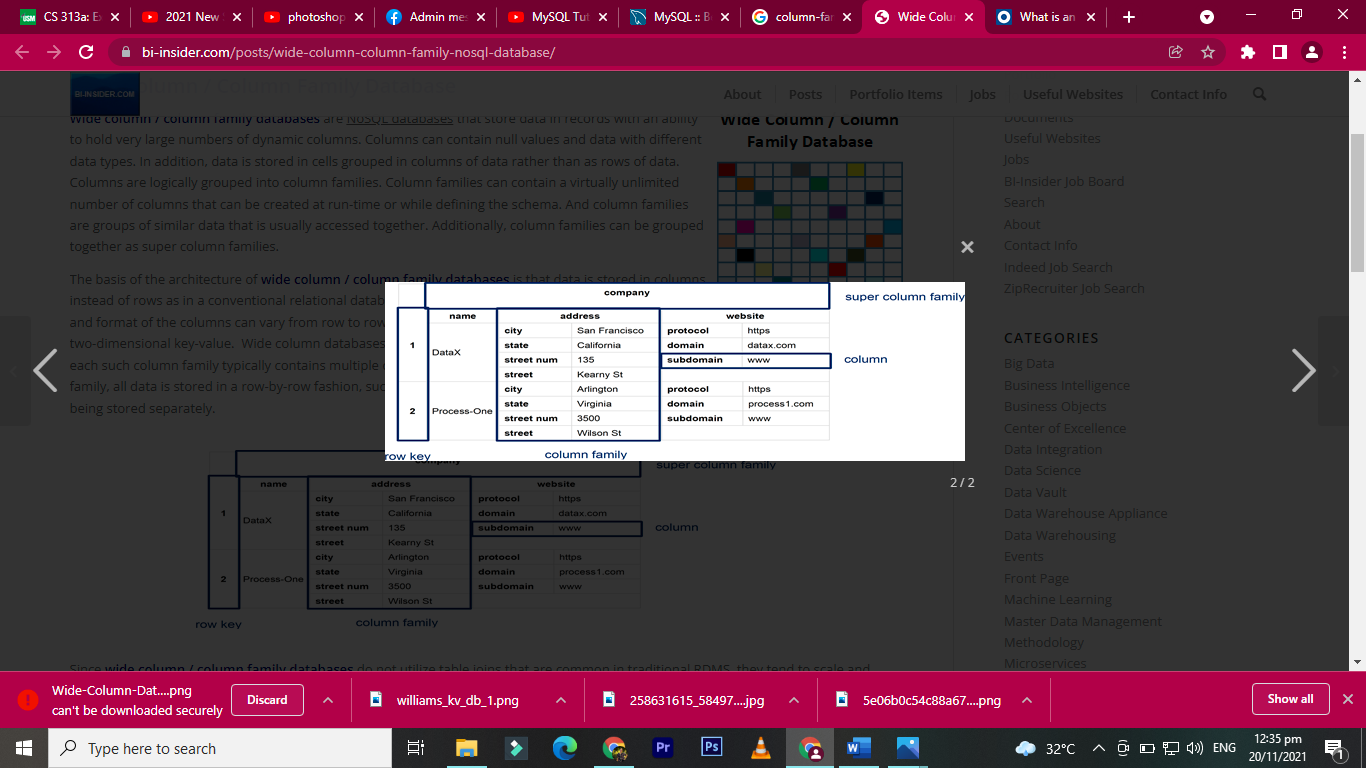
* Market data — Historical price and volume data for financial markets for testing trading strategies.
* Transactional data — Historical transactions that can include purchasing patterns for improved marketing.
* Sensor data — Historical data from sensors that monitor situations like the weather.
* Natural language data — Study of social media posts for research purposes.
* Process data — Study of processes to better understand logistics and find bottlenecks.
* Machine data — Software and hardware-generated data from products to improve efficiency.

1. Key-value model

Key-value stores are actually pretty straightforward. A value, which can be basically any piece of data or information, is stored with a key that identifies its location. In fact, this is a design concept that exists in pretty much every piece of programming as an array or map object. The difference here is that it’s stored persistently in a database management system.



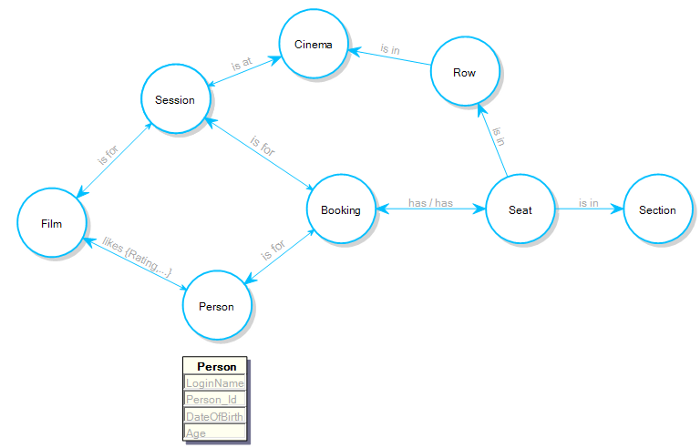
1. **Wide column / column family databases** are NoSQL databases that store data in records with an ability to hold very large numbers of dynamic columns. Columns can contain null values and data with different data types. In addition, data is stored in cells grouped in columns of data rather than as rows of data. Columns are logically grouped into column families. Column families can contain a virtually unlimited number of columns that can be created at run-time or while defining the schema. And column families are groups of similar data that is usually accessed together. Additionally, column families can be grouped together as super column families.



## Graph Databases and Graphs

Graph databases are described as databases that operate over [graphs](https://en.wikipedia.org/wiki/Graph_theory) and where relationships between things matter. A graph is a type of structure and the underlying graph of a graph database maps the structure, or schema, of the data stored in the database.

The picture below is a graph model for a seat booking database solution for a cinema. We would use such a schema to book seats to watch a film in a particular session at that cinema. I believe it can be readily said that graph schemas are straight forward to look at.



## Document-Oriented Database

A document-oriented database is a specific kind of database that works on the principle of dealing with 'documents’ rather than strictly defined tables of information.

The document-oriented database plays an important role is aggregating data from documents and getting them into a searchable, organized form.