**The Relational Model**

**Database Systems** – Powerful tools for **creating** and **managing** large amounts of data **efficiently** and **allowing** it to persist over long periods of time.

**Database**: a **collection** of **data** organized in a particular way and managed by a DBMS.

**DBMS**: Database Management System – a **collection** of software **programs** that manage a database.

**Data Independence:** Applications and users should be **insulated** from how data is **structured** and **stored**.

**Logical** independence: Protection from **changes** in **logical** structure of data (i.e. attributes/schema)

**Physical** independence: Protection from **changes** in **physical** structure of data (i.e. how it is stored)

**DBMS:**

A DBMS is a set of computer programs that support:

* At least one **data model** (a mathematical abstraction for representing data) to define a **database** and an associated **high-level language** to **create**, **manage** and **query** the database.
* Ability to **store** and **manage** large amounts of **persistent** data **efficiently**.
* **Transaction** management, **concurrency** control.
* **Access control** (limit access of certain data to certain users).
* **Resiliency** (ability to recover from crashes).

**Data Model:**

A **collection** of mathematical **concepts** for describing data – how it is **represented**, **organized** and **structured** in the **database**.

The **relational** **model** of data is the most widely used today.

Every data model has to come with a **data** **language** for defining, updating, manipulating, and retrieving data from its representation.

**Data Sublanguages:**

A data language has two parts (**sublanguages**):

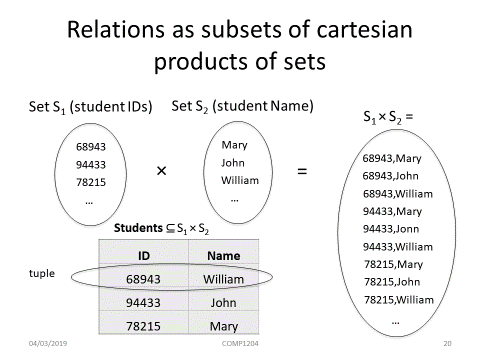
* Data **definition** language (DDL) – syntax for describing “**database** **templates**” in terms of the underlying data model.
* Data **manipulation** language (DML) – for insertion, deletion, update…

**Eg. DDL for XML – XML Schema; DML for XML – XPath/XQuery.**

**Relational Data Model:**

Formally: **a relation is a subset of a Cartesian product of sets.**

Informally: Tables.



**Basic Notions from Discrete Mathematics**

A k-tuple is an **ordered** **sequence** of k objects (need **NOT** be **distinct**) – (a, b, a, a, c) is a 5-tuple.

**If D1 = {0, 1} and D2 = {a, b, c, d} then |D1| \* |D2| = 8**

A **k-ary** **relation** R is a subset of a Cartesian product of k sets. K is called the **arity** of the relation.

* E.g. Unary: R = {0, 2, 4}
* Binary: T = {(a,b): a and b have the same b-day}
* Ternanry: S = {(m,n,s): s = m + n}

**The number of tuples is the number of rows.**

All rows are **distinct.**

The ordering of the attributes in **not significant.**