## A little of mathematics

# Databases

- We will be particularly interested in relational databases.
- Data are stored in tables.
- Why mathematics?

  Relational databases are inspired of relations - mathematics.

# Tables

- Set of rows (no duplicates)
- Each row describes a different entity.
- Each column states a particular fact about each entity.
  - Has an associated domain.

| Id | Name | Address | Status |
|---|---|---|---|
| 1111 | John | 123 Main | Fresh |
| 2222 | Mary | 321 Oak | Soph |
| 1234 | Bob | 444 Pine | Soph |
| 9999 | Joan | 777 Grand | Senior |

# Sets

- **Sets** are the basic data structures of mathematics.
- Intuitively, any "well-defined" collection of mathematical objects can be grouped together to form a single object – a *set*.
- Well-known examples of sets from mathematics are the set of integers (N), the set of rational numbers (Q), and the set of real numbers (R). In computer science one often deals with sets of strings (also called "formal languages") or trees.

# Description of Sets

- A set with no elements is called an **empty set**. It is denoted $\emptyset$ or $\{\}$. It is **unique**.
- The number of element of a set $S$ is denoted $|S|$. We say also the **cardinal** of $S$.
- **Finite** sets can in principle be described by *listing* their elements.

  That is, we write
  $$\{x_1, \ldots, x_n\}$$
  to denote the set consisting of elements $x_1, \ldots, x_n$.
- A more general mechanism for describing a set (*finite or infinite*) is to characterize via a *logical formula* a condition (property) its elements have to satisfy:

  For every set $S$ and formula $P(x)$ there exists a set, denoted by
  $$\{x \in S \mid P(x)\},$$
  that consists of all elements of $S$ for which $P$ is true.

# Examples of Sets

- $\{\emptyset, 1, (4, 5), \text{"}bonjour\text{"}\}$
- The (finite) set of integers between −2 and 5:
  $$\{n \in \mathbf{Z} \mid -2 < n < 5\}$$
- The (open) interval of real numbers between −2 and 5:
  $$\{x \in \mathbf{R} \mid -2 < x < 5\}$$
- The (infinite) set of even integers:
  $$\{n \in \mathbf{Z} \mid \exists k\,(n = 2k)\}$$
- From a general description it may not always be obvious what the elements of the set are:
  - $$\{(x, y, z) \in \mathbf{N} \times \mathbf{N} \times \mathbf{N} \mid (z = x + y)\}$$
  - $$\{(x, y, z) \in \mathbf{N}\times\mathbf{N}\times\mathbf{N} \mid \exists n \in \mathbf{N}, (n > 2 \wedge x^n + y^n = z^n)\}$$

# Set Theory

- The basic concepts of sets theory are **sets** and the **elements-relationship**.
- The symbol $\in$ is commonly used to denote the **membership relation**, and one writes $x \in A$ to denote the proposition $x$ *is an element of* $A$ (which may be true or false).
- Intuitively, sets are **unordered** collections of objects, where the **multiplicities** of elements *don't matter*.
- **Examples:**
  $$\begin{aligned} \{1, 2\} &= \{2, 1\}? \\ \{1, 2\} &= \{1, 1, 2, 2, 2\}? \\ \{1, 2, 3\} &= \{1, 1, 1, 3\}? \end{aligned}$$

# Ordered Pairs and Tuples

- Sets are **unordered** collections of elements.
- **Pairs**, or more generally **tuples**, are **ordered** collections of elements.
- **Examples:** (1, 2) is a pair (a tuple of length 2). (1, 2, 4, 5) is a tuple of length 4.
- Tuples of different lengths are never equal.
- **Examples:**
  $$\begin{aligned} (1, 2) &\neq (2, 1) \\ \{1, 2, 3\} &= \{1, 3, 2\} \\ (1, 2, 3) &\neq (1, 3, 2) \\ \{1, 2\} &= \{1, 2, 2\} \\ (1, 2) &\neq (1, 2, 2) \end{aligned}$$

## Subsets ($\subseteq$)

- **Definition:**

  A set $A$ is a **subset** of another set $B$, written $A \subseteq B$, if, and only if, every element of $A$ is also an element of $B$.

- **Examples:**

$$\begin{aligned} \{1,2\} &\subseteq \{1,2,3\}? \\ \{1,1,2,2\} &\subseteq \{1,2\}? \\ \{1\} &\subseteq \{2,3,5,7\}? \end{aligned}$$

- The subset relation is often used to establish *equality* of sets, based on the following lemma.

  **Lemma:** If $A \subseteq B$ and $B \subseteq A$, then $A = B$.

## Proper subsets ($\subset$)

- **Definition:**

  $A$ is a **proper subset** of $B$, written $A \subset B$, if $A$ is a subset of $B$, but not equal to $B$.

- **Examples:**

$$\{1,2\} \subset \{1,2,3\}?$$

$$\{1,2\} \subset \{1,1,2,2\}?$$

## Membership and subset relations

- Be careful about the distinction between the element relation and the subset relation.

- **Examples:**

$$\begin{aligned} 2 &\in \{1,2,3\}? \\ \{2\} &\in \{1,2,3\}? \\ 2 &\subseteq \{1,2,3\}? \\ \{2\} &\subseteq \{1,2,3\}? \\ \{2\} &\subseteq \{\{1\},\{2\}\}? \\ \{2\} &\in \{\{1\},\{2\}\}? \end{aligned}$$

## Property of the Empty Set

- **Theorem:**

  If $\emptyset$ is an empty set, then $\emptyset \subseteq A$, for all sets $A$.

# Cartesian Products

- Pairs and tuples provide us with a way of constructing new sets from given ones.

- **Definition:**

  If $A$ and $B$ are sets, then there exists a set $A \times B$ (read "A cross B"), called the *Cartesian product* of $A$ and $B$, that consists of all ordered pairs $(a, b)$, where $a \in A$ and $b \in B$.

- Symbolically,

$$A \times B = \{(a, b) \mid a \in A \land b \in B\}.$$

- For example, if $A = \{1, 2\}$ and $B = \{4, 5\}$, then

$$A \times B = \{(1, 4), (1, 5), (2, 4), (2, 5)\}.$$

# More Set Operations

- Other operations for constructing sets include
  - *set union* ($\cup$)
  - *set intersection* ($\cap$)
  - *relative complementation* (or *set difference*) ($-$)
  - *complementation* ($^c$)

  They are defined as follows.

- Let $A$ and $B$ be subsets of some set $U$. We define:

$$\begin{aligned} A \cup B &= \{x \in U \mid x \in A \lor x \in B\} \\ A \cap B &= \{x \in U \mid x \in A \land x \in B\} \\ B - A &= \{x \in U \mid x \in B \land x \notin A\} \\ A^c &= \{x \in U \mid x \notin A\} \end{aligned}$$

  Note that set difference can also be defined as follows:

$$A - B = A \cap B^c.$$

- For example, let

  $R$ be the set of real numbers,
  $A$ the set $\{x \in \mathbf{R} \mid -1 < x \leq 0\}$,
  $B$ the set $\{x \in \mathbf{R} \mid 0 \leq x < 1\}$.

  What are $A \cup B$, $A \cap B$, $B - A$, and $A^c$?

# Properties of the Empty Set

- **Theorems:**

$$\begin{aligned} A \cup \emptyset &= A \\ A \cap \emptyset &= \emptyset \end{aligned}$$

# Set Identities

- 1. Set union and intersection are commutative.

  2. Set union and intersection are associative.

  3. Distributivity: $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$

  4. Double complement: $(A^c)^c = A$.

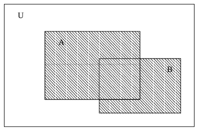
  5. Idempotency: $A \cap A = A \cup A = A$.

  6. De Morgan's Laws:
     $(A \cup B)^c = A^c \cap B^c$ and $(A \cap B)^c = A^c \cup B^c$.

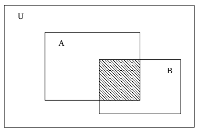
  7. Absorption: $A \cup (A \cap B) = A$ and $A \cap (A \cup B) = A$.

# Venn Diagrams

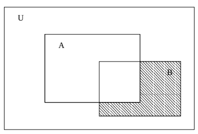
- Sets can often be conveniently represented by **Venn diagrams**.
- The union $A \cup B$ of $A$ and $B$ is represented by:




- The intersection $A \cap B$ is represented by:




- The set difference $B - A$ is represented by:




# Disjoint Sets

- Two sets $A$ and $B$ are said to be **disjoint** if they have no elements in common, i.e., $A \cap B = \emptyset$.
- **Examples:**

  Is $\{\emptyset, \{\emptyset\}\} \cap \{\emptyset\} = \emptyset$?
  *No*, $\{\emptyset, \{\emptyset\}\} \cap \{\emptyset\} = \{\emptyset\}$.
  Is $\{\emptyset, \{\emptyset\}\} \cap \emptyset = \emptyset$?
  *Yes*, because $A \cap \emptyset = \emptyset$.

- A **partition** of a set $A$ is a collection of pairwise disjoint sets $A_1, \ldots, A_n$, such that

$$A = A_1 \cup A_2 \cup \cdots \cup A_n.$$

- For example, at the end of the semester I will partition the class into subsets with grades of $A$, $A-$, etc. It will be a partition, since each student gets one, and only one, grade.

# Powersets ($\mathcal{P}$)

- **Powerset Axiom:**

  If $A$ is a set, then there exists a set, called the **powerset** of $A$ and denoted by the symbol $\mathcal{P}(A)$, whose elements are exactly all the subsets of $A$.

- **Example:**

  If $A$ is the set $\{1, 2, 3\}$, then

$$\begin{aligned}\mathcal{P}(A) \;=\; \{&\emptyset,\\ &\{1\}, \{2\}, \{3\},\\ &\{1,2\}, \{1,3\}, \{2,3\},\\ &\{1,2,3\}\\ \}&\end{aligned}$$

  Do we have $1 \in \mathcal{P}(A)$, or $2 \in \mathcal{P}(A)$, or $3 \in \mathcal{P}(A)$?

  *No*, because $1 \neq \{1\}$, etc.

- If $A$ has $n$ elements, how many elements are there in its powerset?

  Answer: $2^n$. Why?

# Relations

- **Relations** use ordered tuples to represent relationships among objects.
- **Examples:**
  - "x is a parent of y " – $(Morris, Steve)$, $(Ria, Steve)$
  - "x is a number less than y" – $(3, 42)$, $(42, 43)$
  - "Student number x is named y and majors in z" – $(124324443, Mary, CSE)$, $(563565426, Mary, PSY)$
  - "x is an even number" ... (2)
- Essentially, a relation is the set of assignments which makes a predicate true.
- **Examples:**
  - $IsParent = \{(Morris, Steve),\ (Ria, Steve)\}$
  - $LessThan = \{(3, 42), (42, 43)\}$
  - $MajorIn = \{(124324443, Mary, CSE),$ $(563565426, Mary, PSY)\}$
  - $IsEven = \{n \mid n = 2k\}$

# Binary Relations

- Binary relations have two blanks, relating two objects.
- More formally, suppose $A$ and $B$ are sets.
  A **binary relation** from $A$ to $B$ is a set $R \subseteq A \times B$.
- Thus $R$ is a set of ordered pairs $(a, b)$ where $a \in A$ and $b \in B$.
- **Notation:** If $(a, b) \in R$ then we sometimes write $aRb$.
- **Example:**
  $A = \{2, 6, 7\}$, $B = \{1, 2, 5\}$.
  $R_1$ is "x in $A$ is an integer multiple of y in $B$."
  so $R_1 = \{(2, 1), (2, 2), (6, 1), (6, 2), (7, 1)\}$

# Presenting Binary Relations

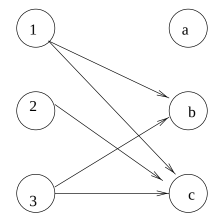
- Binary relations are particularly useful because they have two kinds of compact visual representation, **tables** and **graphs**.
- **Tables:**

| R | a | b | c |
|---|---|---|---|
| 1 | | * | * |
| 2 | | | * |
| 3 | | * | * |

or

| R | |
|---|---|
| 1 | b |
| 1 | c |
| 2 | c |
| 3 | b |
| 3 | c |

- **Graphs** are composed of **vertices** or **nodes** connected by **edges** or **arcs**.
  There is an arc from $a$ to $b$ iff $(a, b) \in R$

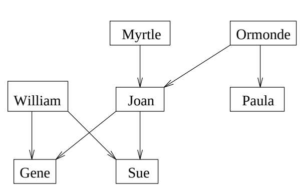



# The Parent-Of Relation

- The parent of relations, "x is a parent of y", is a binary relation between pairs of people.
- Table:

| R | Gene | Joan | William | Sue | Myrtle | Ormonde | Paula |
|---|---|---|---|---|---|---|---|
| Gene | | | | | | | |
| Joan | * | | | * | | | |
| William | * | | | * | | | |
| Sue | | | | | | | |
| Myrtle | | * | | | | | |
| Orm | | * | | | | | * |
| Paula | | | | | | | |

- Graph:




- Which representation is better for testing whether the pair $(x, y)$ is in the relation?
- Which representation is better for capturing the overall structure?

# General ($n$-ary) Relations

- Suppose $A_1$, $A_2$, ..., $A_n$ are sets. A relation of $A_1$, $A_2$, ..., $A_n$ is a set $R \subseteq A_1 \times A_2 \times \ldots \times A_n$.
- Thus $R$ is a set of ordered $n$-tuples $(a_1, a_2, \ldots, a_n)$ where $a_i \in A_i$.
- **Example:**
  $A_1 = N$, $A_2 = names$, $A_3 = majors$
  "Student number x is named y and majors in z"
  $(124324443, Mary, CSE), (563565426, Mary, PSY)$ are tuples of the relation.
- Such structures are modeled by **hypergraphs**, a graph structure where each "edge" represents a subset of more than two vertices.

# Relational Databases

## Overview

- The most important commercial database systems today employ the **relational** model, meaning that the data is stored as tables of tuples, i.e. relations.

  A relation is a mathematical entity corresponding to a table:

  – row - tuple

  – column attribute.

- A Shakespearian **killed** relation would be:

| Killer | Victim |
|---|---|
| Brutus | Caesar |
| Hamlet | Laertes |
| Hamlet | Polonius |
| Laertes | Hamlet |
| Brutus | Brutus |
| Cassius | Caesar |

- **Requests** for information from the database is made in a query language like **SQL** which is based on the notations of set theory and the predicate calculus.

- **Example 1: Who killed Caesar?**

  In SQL:

  **SELECT Killer from Killed where victim='Caesar'**

  This reads "select from relation 'killed' all tuples where the victim was Caesar, and report only the killer field from each.

- **Example 2: Who was both a killer and a victim?**

  In SQL:

  **(SELECT Killer from Killed) INTERSECT (SELECT Victim from Killed)**

- Much of the power of relational databases comes from the fact that we can **combine different relations**.

- For example, suppose we also have a **died-by** relation:

| Victim | Method |
|---|---|
| Caesar | Daggers |
| Hamlet | Sword |
| Laertes | Sword |
| Polonius | Sword |
| Brutus | Sword |

  We can combine the two tables with a **join** operation, which the tables based on **common fields**. For example, the join of **killed** and **died-by** is:

| Killer | Victim | Method |
|---|---|---|
| Brutus | Caesar | Daggers |
| Hamlet | Laertes | Sword |
| Hamlet | Polonius | Sword |
| Laertes | Hamlet | Sword |
| Brutus | Brutus | Sword |
| Cassius | Caesar | Daggers |

- **Example 3: Which killers used daggers?**

  In SQL:

  **SELECT Killer FROM Killed, Died_by**
  **WHERE Killed.victim=died_by.victim**
  **AND Method='Daggers'**

- Note that this database design assumes that each victim can only be killed by one weapon (sorry, Rasputin).