### 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.

<table>
<thead>
<tr>
<th>Id</th>
<th>Name</th>
<th>Address</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>1111</td>
<td>John</td>
<td>123 Main</td>
<td>Fresh</td>
</tr>
<tr>
<td>2222</td>
<td>Mary</td>
<td>321 Oak</td>
<td>Soph</td>
</tr>
<tr>
<td>1234</td>
<td>Bob</td>
<td>444 Pine</td>
<td>Soph</td>
</tr>
<tr>
<td>9999</td>
<td>Joan</td>
<td>777 Grand</td>
<td>Senior</td>
</tr>
</tbody>
</table>
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.

---

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:**
  \[
  \begin{align*}
  \{1,2\} & \neq \{2,1\} \quad & \{1,2\} \neq \{1,1,2,2\} \quad & \{1,2,3\} \neq \{1,1,1,3\} \\
  \end{align*}
  \]

---

Examples of Sets

- \( \{\emptyset, 1, (4,5), \text{"bonjour"}\} \)

- The (finite) set of integers between \(-2\) and \(5\):
  \[
  \{n \in \mathbb{Z} \mid -2 < n < 5\}
  \]

- The (open) interval of real numbers between \(-2\) and \(5\):
  \[
  \{x \in \mathbb{R} \mid -2 < x < 5\}
  \]

- The (infinite) set of even integers:
  \[
  \{n \in \mathbb{Z} \mid \exists k (n = 2k)\}
  \]

- From a general description it may not always be obvious what the elements of the set are:
  \[
  \{(x,y,z) \in \mathbb{N} \times \mathbb{N} \times \mathbb{N} \mid z = x + y\}
  \]
  \[
  \{(x,y,z) \in \mathbb{N} \times \mathbb{N} \times \mathbb{N} \mid \exists n \in \mathbb{N}, (n > 2^n)\}
  \]

---

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:**
  \[
  (1,2) \neq (2,1) \quad (1,2,3) \neq (1,3,2) \quad (1,2,3) \neq (1,3,2) \quad (1,2) \neq (1,2,2) \quad (1,2) \neq (1,2,2)
  \]
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:**
  
  - $\{1, 2\} \subseteq \{1, 2, 3\}$?
  - $\{1, 1, 2, 2\} \subseteq \{1, 2\}$?
  - $\{1\} \subseteq \{2, 3, 5, 7\}$?

- **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 set $A$ 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:**

  - $2 \in \{1, 2, 3\}$?
  - $\{2\} \in \{1, 2, 3\}$?
  - $2 \subseteq \{1, 2, 3\}$?
  - $\{2\} \subseteq \{1, 2, 3\}$?
  - $\{2\} \subseteq \{\{1\}, \{2\}\}$?
  - $\{2\} \in \{\{1\}, \{2\}\}$?

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** ($'$)

  They are defined as follows.

- Let $A$ and $B$ be subsets of some set $U$. We define:
  $$A \cup B = \{x \in U \mid x \in A \lor x \in B\}$$  

  $$A \cap B = \{x \in U \mid x \in A \land x \in B\}$$  

  $$B - A = \{x \in U \mid x \in B \land x \notin A\}$$  

  $$A' = \{x \in U \mid x \notin A\}$$

  Note that set difference can also be defined as follows:
  $$A - B = A \cap B'.$$

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

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

---

Properties of the Empty Set

- **Theorems:**
  $$A \cup \emptyset = A$$  
  $$A \cap \emptyset = \emptyset$$
Venn Diagrams

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

![Venn Diagram]

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

![Venn Diagram]

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

![Venn Diagram]

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
  $$\mathcal{P}(A) = \{\emptyset, \{1\}, \{2\}, \{3\}, \{1, 2\}, \{1, 3\}, \{2, 3\}, \{1, 2, 3\}\}$$

  Do we have $1 \in \mathcal{P}(A)$, or $2 \in \mathcal{P}(A)$, or $3 \in \mathcal{P}(A)$?
  - No, because $1 \notin \{1\}$, 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 $y$” $\rightarrow$ (Morris, Steve), (Ria, Steve)
  - “$x$ is a number less than $y$” $\rightarrow$ (3, 42), (42, 43)
  - “Student number $x$ is named $y$ and majors in $z$” $\rightarrow$ (124324443, Mary, CSE), (563565426, Mary, PSY)
  - “$x$ is an even number” $\ldots$ (2)

  Essentially, a relation is the set of assignments which makes a predicate true.

- **Examples:**
  - $\text{IsParent} = \{(\text{Morris, Steve}), (\text{Ria, Steve})\}$
  - $\text{LessThan} = \{(3, 42), (42, 43)\}$
  - $\text{MajorIn} = \{(124324443, \text{Mary, CSE}), (563565426, \text{Mary, PSY})\}$
  - $\text{IsEven} = \{n \mid n = 2k\}$
**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:**
\[
A = \{2, 6, 7\}, \quad B = \{1, 2, 5\}.
\]
\( R_1 \) is “\( x \) in \( A \) is an integer multiple of \( y \) in \( B \).”
so \( R_1 = \{(2, 1), (2, 2), (6, 1), (6, 2), (7, 1)\} \)

---

**The Parent-Of Relation**

- The parent of relations, “\( x \) is a parent of \( y \)”, is a binary relation between pairs of people.

**Table:**

<table>
<thead>
<tr>
<th></th>
<th>Gene</th>
<th>Joan</th>
<th>William</th>
<th>Sue</th>
<th>Myrtle</th>
<th>Ormonde</th>
<th>Paula</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gene</td>
<td></td>
<td>***</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Joan</td>
<td>***</td>
<td>**</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>William</td>
<td></td>
<td></td>
<td></td>
<td>**</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sue</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>**</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Myrtle</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>**</td>
</tr>
<tr>
<td>Ormonde</td>
<td></td>
<td></td>
<td></td>
<td>**</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Paula</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>**</td>
<td></td>
</tr>
</tbody>
</table>

**Graph:**

```
William -- Myrtle         Ormonde
           v               v
            v               v
                v               v
                Sue            Paula
```

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

---

**Presenting Binary Relations**

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

**Tables:**

<table>
<thead>
<tr>
<th>R</th>
<th>a</th>
<th>b</th>
<th>c</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td>2</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td>3</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
</tbody>
</table>

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

```
1  2  3  4

1---2
|   |
|   |
|   |
```

**General (n-ary) Relations**

- Suppose \( A_1, A_2, \ldots, A_n \) are sets. A relation of \( A_1, A_2, \ldots, A_n \) 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:**
\( A_1 = N, A_2 = \text{name}, A_3 = \text{majors} \)
“Student number \( x \) is named \( y \) and majors in \( z \)”
(1243244434, 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.
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:

<table>
<thead>
<tr>
<th>Killer</th>
<th>Victim</th>
</tr>
</thead>
<tbody>
<tr>
<td>Brutus</td>
<td>Caesar</td>
</tr>
<tr>
<td>Hamlet</td>
<td>Laertes</td>
</tr>
<tr>
<td>Hamlet</td>
<td>Polonius</td>
</tr>
<tr>
<td>Laertes</td>
<td>Hamlet</td>
</tr>
<tr>
<td>Brutus</td>
<td>Brutus</td>
</tr>
<tr>
<td>Cassius</td>
<td>Caesar</td>
</tr>
</tbody>
</table>

- 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:

<table>
<thead>
<tr>
<th>Victim</th>
<th>Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>Caesar</td>
<td>Daggers</td>
</tr>
<tr>
<td>Hamlet</td>
<td>Sword</td>
</tr>
<tr>
<td>Laertes</td>
<td>Sword</td>
</tr>
<tr>
<td>Polonius</td>
<td>Sword</td>
</tr>
<tr>
<td>Brutus</td>
<td>Sword</td>
</tr>
</tbody>
</table>

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:

<table>
<thead>
<tr>
<th>Killer</th>
<th>Victim</th>
<th>Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>Brutus</td>
<td>Caesar</td>
<td>Daggers</td>
</tr>
<tr>
<td>Hamlet</td>
<td>Laertes</td>
<td>Sword</td>
</tr>
<tr>
<td>Hamlet</td>
<td>Polonius</td>
<td>Sword</td>
</tr>
<tr>
<td>Laertes</td>
<td>Hamlet</td>
<td>Sword</td>
</tr>
<tr>
<td>Brutus</td>
<td>Brutus</td>
<td>Sword</td>
</tr>
<tr>
<td>Cassius</td>
<td>Caesar</td>
<td>Daggers</td>
</tr>
</tbody>
</table>

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).