Databases

sql || no-sql

what is a database?

- it is a **collection of data**, typically describing the activities of one (or more related) application(s)

- the goal is to organize data in a way that facilitates **efficient *retrieval* and *modification***

- **note:** the data maintained by a system are much more important/valuable than the system itself

- A **database management system** (DBMS) is a software program to assist in maintaining and utilizing large databases

advantages of using a dbms

- data independence
- efficient data access
- data integrity and security
- data administration
- concurrent access and crash recovery
- reduced application development time

more on data independence

- **Idea:** application programs are isolated from changes in the way the data is structured & stored.
  - Indirect access supports:
    - advanced data structures
    - data restructuring
    - distribution and load balancing,
    - …
    - all without changes to applications
  - **Note:** A very important advantage of using a DBMS!

more on data independence

- **Logical**: applications immune from changes in the logical structure of the data.
  - Example:
    - Student (name: string, major: string, DOB: integer)
    - …
    - …
- **Physical**: applications immune from physical storage details.
  - Such as the file structure and the choice of indexes

more on relational model

Idea. *All information is organized in flat relations.*

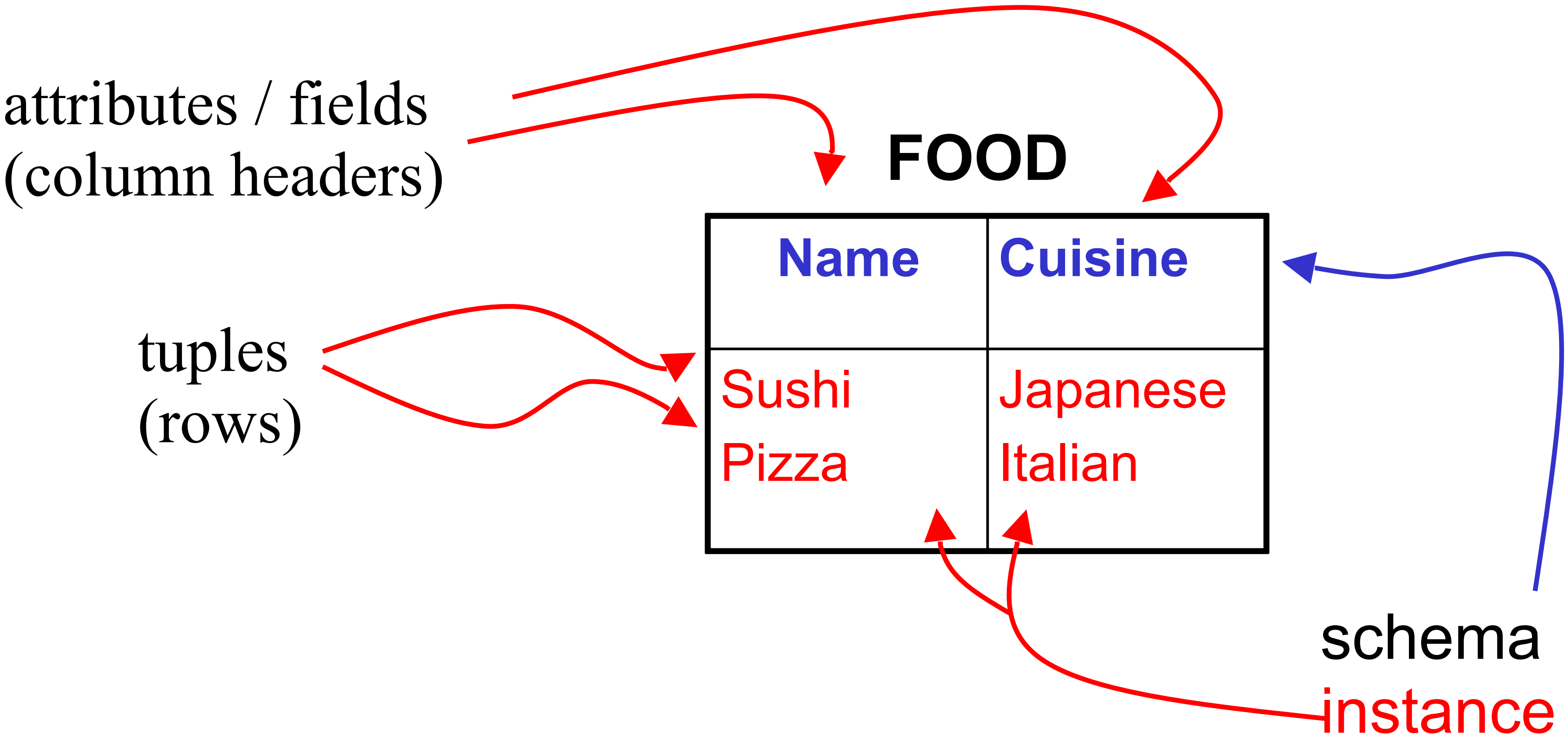
- Features:
  - very simple and clean data model
  - *often* matches how we think about data
  - abstract model that underlies SQL, the most popular database language
  - powerful and *declarative* query/update languages
  - semantic integrity constraints

transaction

A **transaction** is any ***one execution*** of a process in a DBMS, which is seen as a series of **actions**—such as *reads* and *writes*, followed by a *commit* or an *abort*.

- Properties of transactions: **(ACID**)
  - **Atomic:** either all actions or nothing are carried out.
  - **Consistency:** must preserve the DB constraints.
  - **Isolation:** understandable without considering other transactions.
  - **Durability:** once committed, the changes made are permanent.

a relation is a table



attributes / fields
(column headers)

FOOD

Name	Cuisine
Sushi	Japanese
Pizza	Italian

tuples
(rows)

schema

instance

more tabular form

FOOD

Name	Cuisine
Pizza	Italian
Stroganoff	Russian
Poutine	Canadian

STUDENT

ID	Name	Major
1022083920	Adam	Math
901183280	Saniya	CS

LIKES

Student	Food
1022083920	Pizza
1022083920	Poutine
901183280	Pizza

that’s why relations are often called “tables”.

SQL examples

-
  ```
  INSERT INTO food VALUES ( “Pizza”, “Canadian” );
  ```

-
  ```
  UPDATE food SET cuisine = “Italian”
     WHERE name = “Pizza”;
  ```

-
  ```
  SELECT name FROM food
     WHERE cuisine = “Russian”;
  ```

-
  ```
  SELECT cuisine, COUNT(*) AS “count”
     FROM food
     GROUP BY cuisine;
  ```

-
  ```
  SELECT DISTINCT cuisine
     FROM food,
              (SELECT food as name FROM likes,student
               WHERE  major=“CS”) csLikes
     WHERE food.name=csLikes.name;
  ```