Database Concepts

Saraswathy S

Objective

- Need for a Database
- File – base systems
- Define DBMS
- Features of DBMS
- Use of DBMS
- RDBMS Basics

Data vs. information

What is data?

- Facts, statistics used for reference or analysis.
- Numbers, characters, symbols, images etc., which can be processed by a computer.
- Data must be interpreted, by a human or machine, to derive meaning
- Latin 'datum' meaning "that which is given"

What is information?

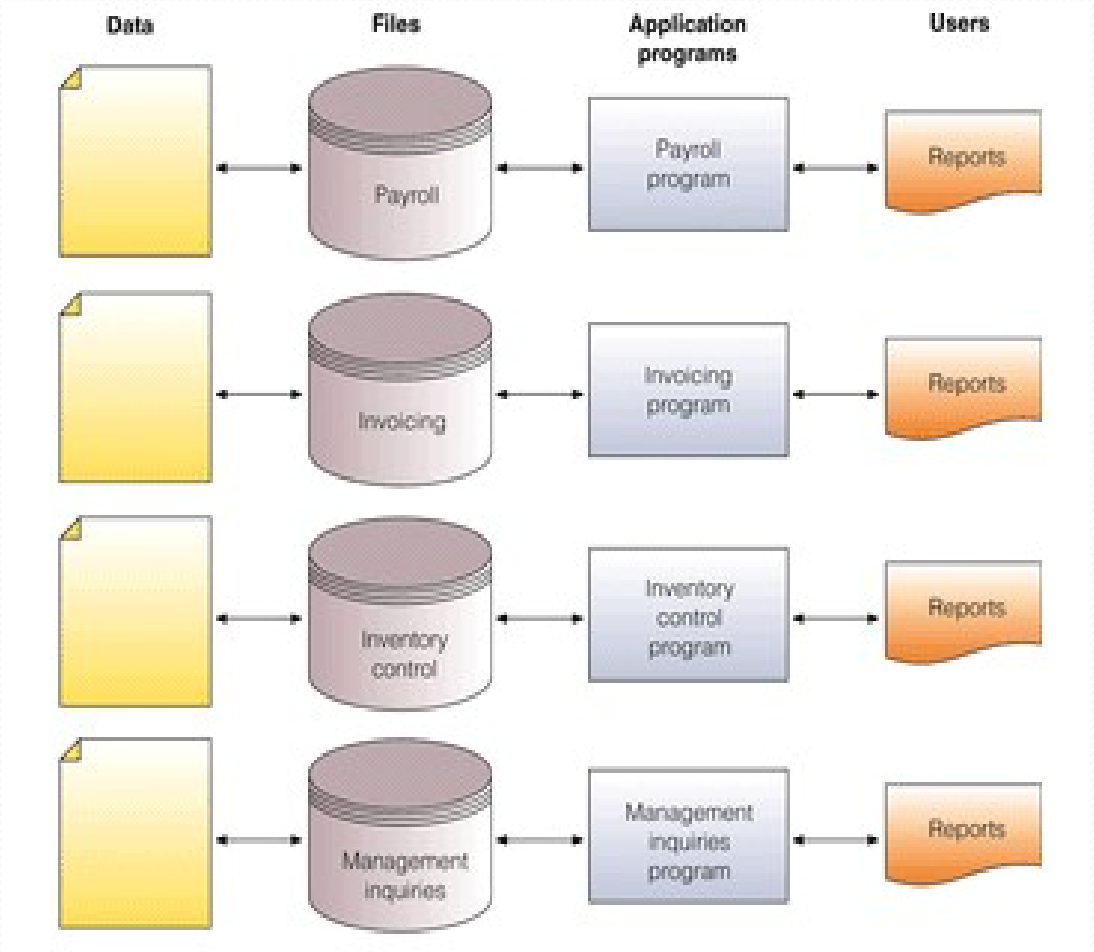
- Information is used to reveal the meaning of data.
- Knowledge derived from study, experience (by the senses), or instruction.
- Communication of intelligence.
- Information is interpreted data

Why do we need a database?

- Keep records of our:
  - Clients
  - Staff
  - Volunteers
- To keep a record of activities and interventions
- Keep sales records
- Develop reports
- Perform research
- Longitudinal tracking

File Based Approach

Every program stores and maintains data in separate files



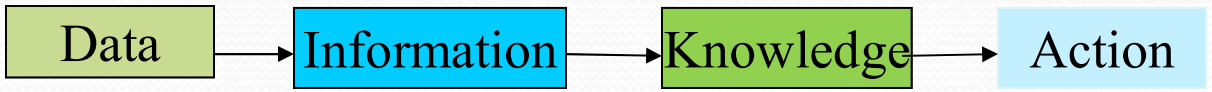
Limitations – File Based approach

- Separation and isolation of data
- Duplication of data
- Data dependence
- Incompatibility of files
- Fixed queries / proliferation of application programs

What is DBMS?

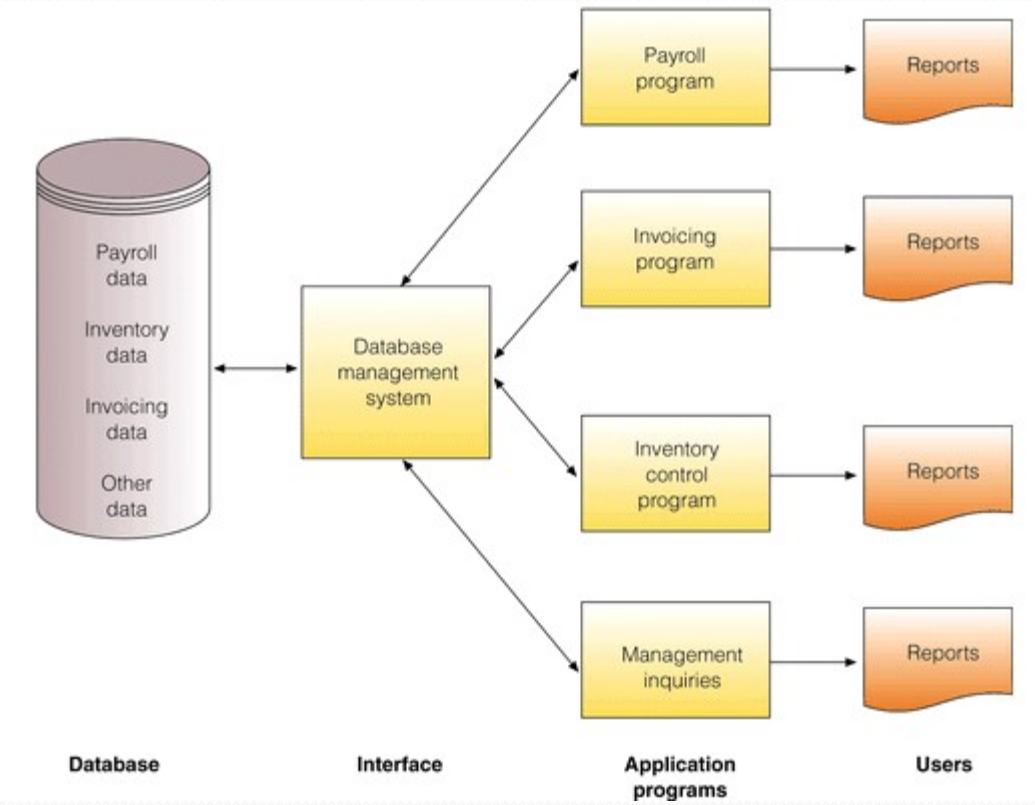
- A very large, integrated collection of data.
- Models real-world enterprise.
  - Entities (e.g., students, courses)
  - Relationships (e.g., John is taking CS662)
- A Database Management System (DBMS) is a software package designed to store and manage databases.

Purpose of a DBMS

- Is to transform 

- Data independence and efficient access.
- Data integrity and security.
- Uniform data administration.
- Concurrent access, recovery from crashes.
- Replication control
- Reduced application development time.

The Database Approach to Manage Data



Advantages of DBMS

- Control of data redundancy
- Data consistency
- Provides multiple views of the same data
- Sharing of data
- Improved data integrity
- Improved security
- Enforcement of standards
- Improved data accessibility and responsiveness
- Improved productivity & maintenance
- Improved backing and recovery services

Limitations – DBMS

- Relatively high cost for purchase, operation & maintaining DBMS
- Size (Substantial space in main memory & large disk space.
- Complexity
- Higher impact of a failure

Database Users

- Database administrators(DBAs):
- Database designers
- End Users
  - Casual
  - Parametric (or naïve)
  - Sophisticated
  - Stand-alone

Relational Database model

- Developed by E.F. Codd, C.J. Date (70s)
- Table = Entity = Relation
- Table row = tuple = instance
- Table column = attribute
- Table linkage by values
- Entity-Relationship Model

The Relational Model

- Each attribute has a unique name within an entity
- All entries in the column are examples of it
- Each row is unique
- Ordering of rows and columns is unimportant
- Each position (tuple) is limited to a single entry.

RDBMS Definitions

- Entity: Object, Concept or event (subject)
- Attribute: a Characteristic of an entity
- Row or Record: the specific characteristics of one entity
- Table: a collection of records
- Database: a collection of tables

RDBMS Basics

- Consider some information the University maintains:
  - Name
  - Address
  - SSN
  - Major
  - Courses Taken
  - Grades Received
  - Tuition Paid
  - Tuition Owed
  - Grants/Scholarships
- *HOW* is this information stored?

  You are an *entity* with *attributes* which *vary*. Within the University, *different areas* have different interests in you (i.e., the Registrar, the Bursar, etc.). Nonetheless, you are still part of the *University* as a whole.

- *HOW* does this relate to a database?

RDBMS Basics

Database Components:

You are an *entity* → Record

with *attributes* → Fields

which *vary* → Fields can contain characters, numbers, symbols, etc.

Within the University, *different areas*, have different interests in you → Files

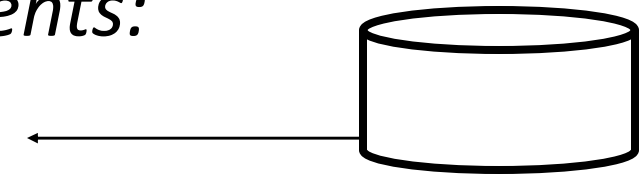
(i.e,. The Registrar, Bursar, etc.)

Nonetheless, you are still part of the *University* → Database

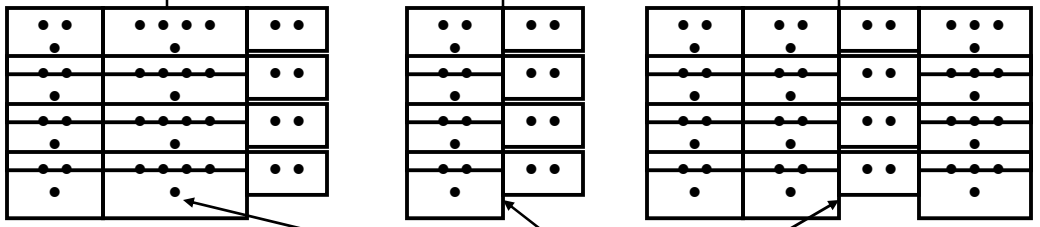
RDBMS Basics

Database Components:

A Database consists of



Files, which contain



Records, which contain

Hernandez, Juan	123456789	72	2.42
Jones, Mary	234567890	102	3.87

Fields, which may consist of a variety of data types

Notice that there should always be a Key (Unique) Field

RDBMS Basics

RDBMS Restrictions/Conventions:

- Each Relation *MUST* have a unique name

Table Student

StudentID	Name	Address	Major
123456789	Saenz, Lupe	123 Mesa	Finance
234567890	Chung, Mei	37 5th St.	INFOSYS
345678901	Adams, John	54B Hague	Accounting
456789012	Elam, Mary	123-22 E St.	Accounting
••••••	••••••	••••••	••••••

Table Balance

Student	Owed	Department
103456678	1,502.36	Marketing
123456789	COBA219	Finance
456789012	COBA232	Accounting
••••••	••••••	••••••

NOT Allowed

Table Student

SID	••••••	Depart
987654321	••••••	Finance
876543210	••••••	INFOSYS
765432199	••••••	Accounting
••••••	••••••	••••••

RDBMS Basics

RDBMS Restrictions /Conventions :

- Each Relation *MUST* have a unique name
- All Columns (Tuples) *MUST* have Unique names

Table Student

StudentID	Name	Address	Address	Major
123456789	Saenz, Lupe	123 Mesa	Arlington	Finance
234567890	Chung, Mei	37 5th St.	New York	INFOSYS
345678901	Adams, John	54B Hague	Dallas	Accounting
456789012	Elam, Mary	123-22 E St.	Ft. Worth	INFOSYS

One of the names *MUST* be changed

* NOTE: The same field names *CAN* be used in different Relations

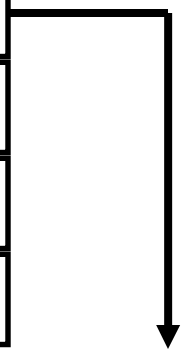
RDBMS Basics

RDBMS Restrictions /Conventions :

- Each Relation *MUST* have a unique name
- All Columns (Tuples) *MUST* have Unique names
- All Column Elements *MUST* be of the same data type

Table Student

StudentID	Name	Address	Major
123456789	Saenz, Lupe	123 Mesa	24.34
234567890	Chung, Mei	37 5th St.	INFOSYS
345678901	Adams, John	54B Hague	Accounting
456789012	Elam, Mary	123-22 E St.	INFOSYS

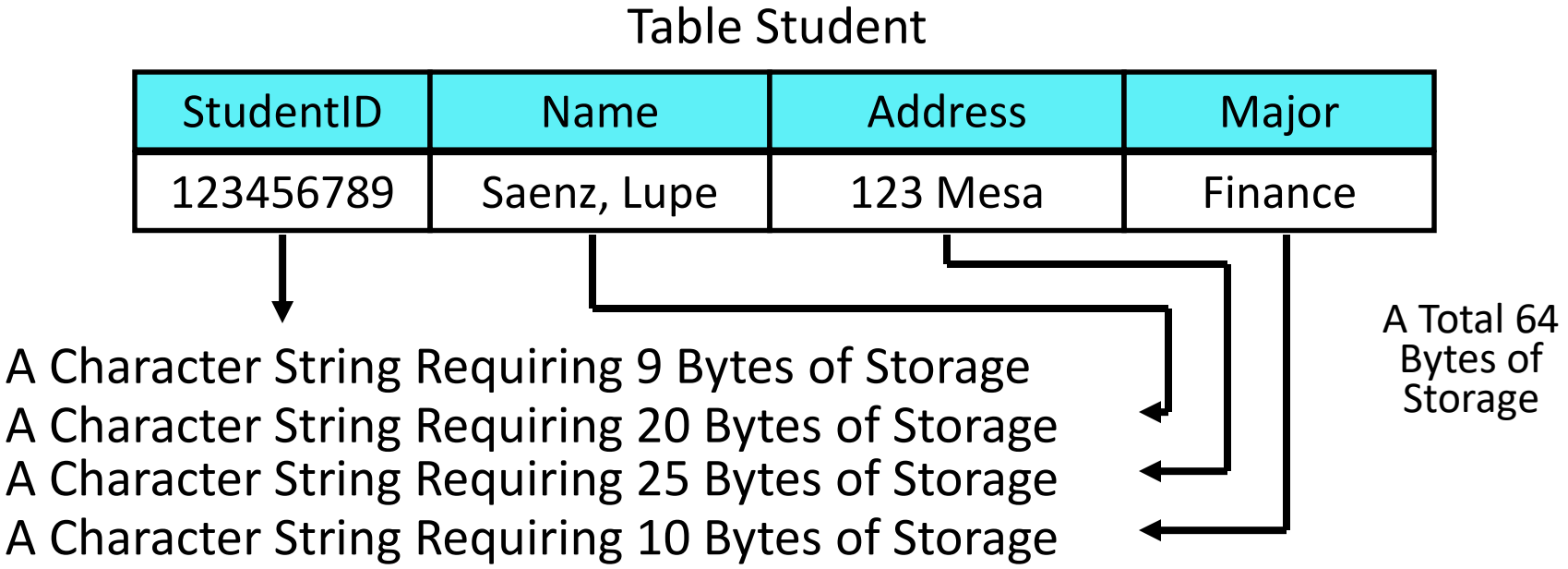


Unless this is stored as Character String “24.34” (and *NOT* as the real number 24.34), it *MUST* be changed

RDBMS Basics

RDBMS Restrictions /Conventions :

- Each Relation *MUST* have a unique name
- All Columns (Tuples) *MUST* have Unique names
- All Column Elements *MUST* be of the same data type
  - Notice that this means each record requires the *SAME* number of Bytes of Storage



RDBMS Basics

RDBMS Restrictions /Conventions :

- Each Relation *MUST* have a unique name
- All Columns (Tuples) *MUST* have Unique names
- All Column Elements *MUST* be of the same data type
- The order of Rows is *NOT* important

StudentID	Name	Address	Major
123456789	Saenz, Lupe	123 Mesa	Finance
234567890	Chung, Mei	37 5th St.	INFOSYS
345678901	Adams, John	54B Hague	Accounting

Is the Same as

StudentID	Name	Address	Major
123456789	Saenz, Lupe	123 Mesa	Finace
345678901	Adams, John	54B Hague	Accounting
234567890	Chung, Mei	37 5th St.	INFOSYS

RDBMS Basics

RDBMS Restrictions /Conventions :

- Each Relation *MUST* have a unique name
- All Columns (Tuples) *MUST* have Unique names
- All Column Elements *MUST* be of the same data type
- The order of Rows is *NOT* important
- The Number of Bytes/Record Must be the same

StudentID	Name	Address	Major
123456789	Saenz, Lupe	123 Mesa	Finance
234567890	Chung, Mei	37 5th St.	INFOSYS
345678901	Adams, John	54B Hague	Accounting

Each contains
64 Bytes

RDBMS Basics

RDBMS Keys:

- Purpose
  - Define entity relationships
  - Determination
    - Knowing the value of a key field means you also know (Determine) the values of the other fields
    - E.g., knowing StudentID means you know StudentName, StudentAddress, etc.

StudentID $\longrightarrow$ StudentName, StudentAddress

(StudentID Determines StudentName and StudentAddress)

RDBMS Basics

RDBMS Keys:

- Purpose
  - Define entity relationships
  - Determination
  - Functional Dependence
    - An attribute is functionally dependent on another if can be determined by that attribute

StudentID ⟶ StudentAddress

(StudentID Determines Student Address)

- NOTE:

StudentAddress ↛ StudentID

(Two Students *MAY* live at the same address)

RDBMS Basics

RDBMS Keys:

- Each Relation *MUST* have a unique identifier or *PRIMARY KEY*

Table Student

StudentID	Name	Address	Major
123456789	Saenz, Lupe	123 Mesa	Finance
234567890	Chung, Mei	37 5th St.	INFOSYS
345678901	Adams, John	54B Hague	Accounting
456789012	Elam, Mary	123-22 E St.	INFOSYS
••••••	••••••	••••••	••••••

No two students can have the same StudentID

RDBMS Basics

RDBMS Keys:

- Each Relation *MUST* have a unique identifier or *PRIMARY KEY*
- A *COMPOSITE KEY* is a combination of keys (Multi-key attributes) used to produce uniqueness

StudentName,StudentAddress $\longrightarrow$ StudentMajor

(StudentName AND StudentAddress Determines StudentMajor)

- NOTE: If Student Major is functionally dependent upon StudentName AND StudentAddress , BUT not on either StudentName or StudentName it is FULLY FUNCTIONAL DEPENDENT on the Concatenated key

(Attributes are fully functionally dependent on *PRIMARY KEYS*)

RDBMS Basics

RDBMS Keys:

- Each Relation *MUST* have a unique identifier or *PRIMARY KEY*

- A is *COMPOSITE KEY* is a combination of keys (Multi-key attributes) used to produce uniqueness
- A is *SUPER KEY* is either a *PRIMARY* or *COMPOSITE KEY* that uniquely identifies an entity

StudentID ⟶ StudentAddress

AND

StudentName,StudentAddress ⟶ StudentMajor

Are BOTH Superkeys

RDBMS Basics

RDBMS Keys:

- Each Relation *MUST* have a unique identifier or *PRIMARY KEY*
- A is *COMPOSITE KEY* is a combination of keys (Multi-key attributes) used to produce uniqueness
- A is *SUPER KEY* is either a *PRIMARY* or *COMPOSITE KEY* that uniquely identifies an entity
- A is *CANDIDATE KEY* is any key or group of keys that could become a *SUPER KEY*
  - StudentID is a candidate key
  - StudentName,StudentAddress is a candidate key
  - StudentID,StudentAddress is *NOT* a candidate key

(StudentID by itself is a *CANDIDATE KEY*)

RDBMS Basics

RDBMS Keys:

- Each Relation *MUST* have a unique identifier or *PRIMARY KEY*
- A is *COMPOSITE KEY* is a combination of keys (Multi-key attributes) used to produce uniqueness
- A is *SUPER KEY* is either a *PRIMARY* or *COMPOSITE KEY* that uniquely identifies an entity
- A is *CANDIDATE KEY* is any key or group of keys that could become a *SUPER KEY*
- A is *SECONDARY KEY* is any field, or combination of fields, which does NOT yield a unique value
  - Used for retrieval/Narrowing purposes only
  - StudentName,StudentZip may yield several records

RDBMS Basics

RDBMS Keys:

- In Order to relate two (or more) tables *FOREIGN KEYS* must be used

Table Student

StudentID	Name	Address	Major
123456789	Saenz, Lupe	123 Mesa	Finance
234567890	Chung, Mei	37 5th St.	INFOSYS
345678901	Adams, John	54B Hague	Accounting

A *FOREIGN KEY* in one table is a *PRIMARY/SUPER KEY* In another

Table Department

DeptName	Chairman	Telephone
INFOSYS	Sircar, Sumit	555-1234
Marketing	McDaniel, C.	555-2345
Accounting	Courtney, H.	555-3456

RDBMS Basics

Database Integrity:

- Maintaining wholeness and Unity
- Entity Integrity:
  - All Entries MUST be Unique
  - No NULL values in primary key fields

Table Student

StudentID	Name	Address	Major
123456789	Saenz, Lupe	123 Mesa	Finance
234567890	Chung, Mei	37 5th St.	INFOSYS
123456789	Adams, John	54B Hague	Accounting
456789012	Elam, Mary	123-22 E St.	INFOSYS
	Bush, G.W.	555 Austin	Marketing

Illegal Entries

RDBMS Basics

Database Integrity:

- Maintaining wholeness and Unity
- Entity Integrity
- Relational Integrity:
  - Foreign Key MUST have a valid entry in the corresponding table (or be NULL)

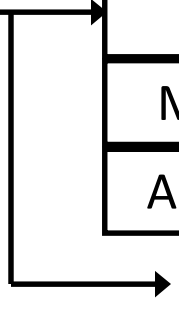
Table Student

StudentID	Major
123456789	Scamming
234567890	INFOSYS
345678901	

???

Table Department

DeptName	Chairman	Telephone
INFOSYS	Sircar, Sumit	555-1234
Marketing	McDaniel, C.	555-2345
Accounting	Courtney, H.	555-3456



RDBMS Basics

Database Integrity:

- Maintaining wholeness and Unity
- Entity Integrity
- Relational Integrity:
  - Foreign Key MUST have a valid entry in the corresponding table (or be NULL)
  - Primary Key entry CAN NOT be deleted if a foreign key refers to it

Table Student

StudentID	Major
123456789	Accounting
234567890	INFOSYS
345678901	Finance

Table Department

DeptName	Chairman	Telephone
INFOSYS	Sircar, Sumit	555-1234
Marketing	McDaniel, C.	555-2345

INFOSYS can NOT be deleted

Summary

- Data - one of the most valuable resources a firm possesses.
- DBMS - a group of programs used as an interface between a database and application programs.
- The database approach to data management provides significant advantages over the traditional file-based approach.