INTRODUCTION TO DATABASE SYSTEMS
IS211

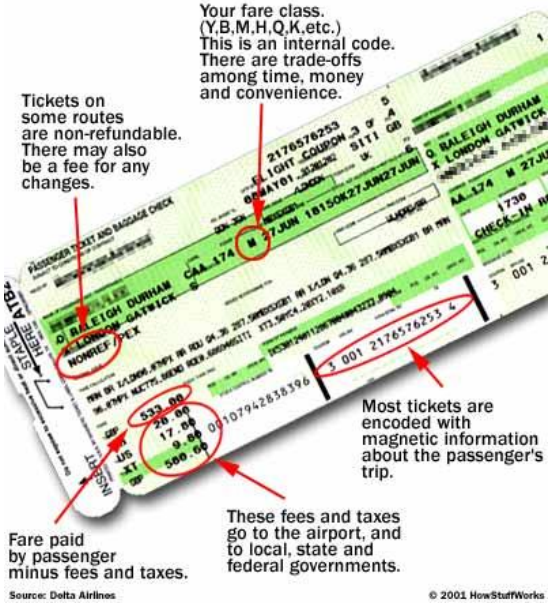
Dr. Noha Nagy

Lecture 2

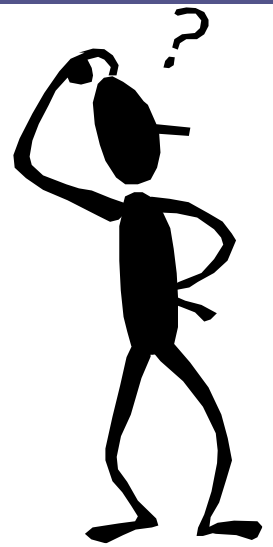
Relational Model

Databases Everywhere

amazon.com

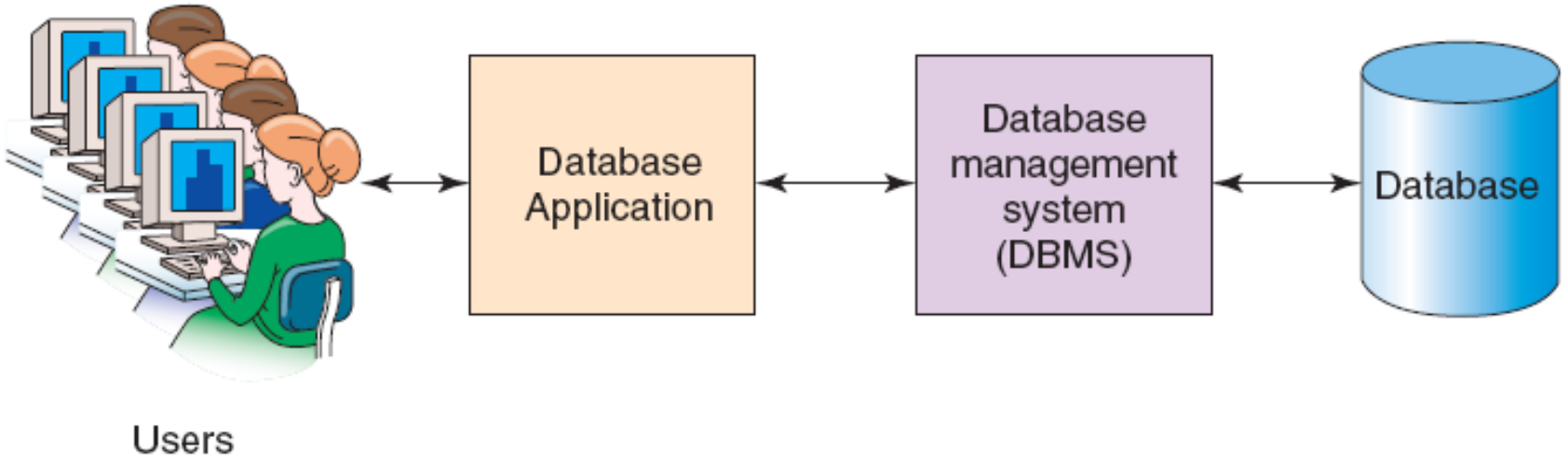


Definitions

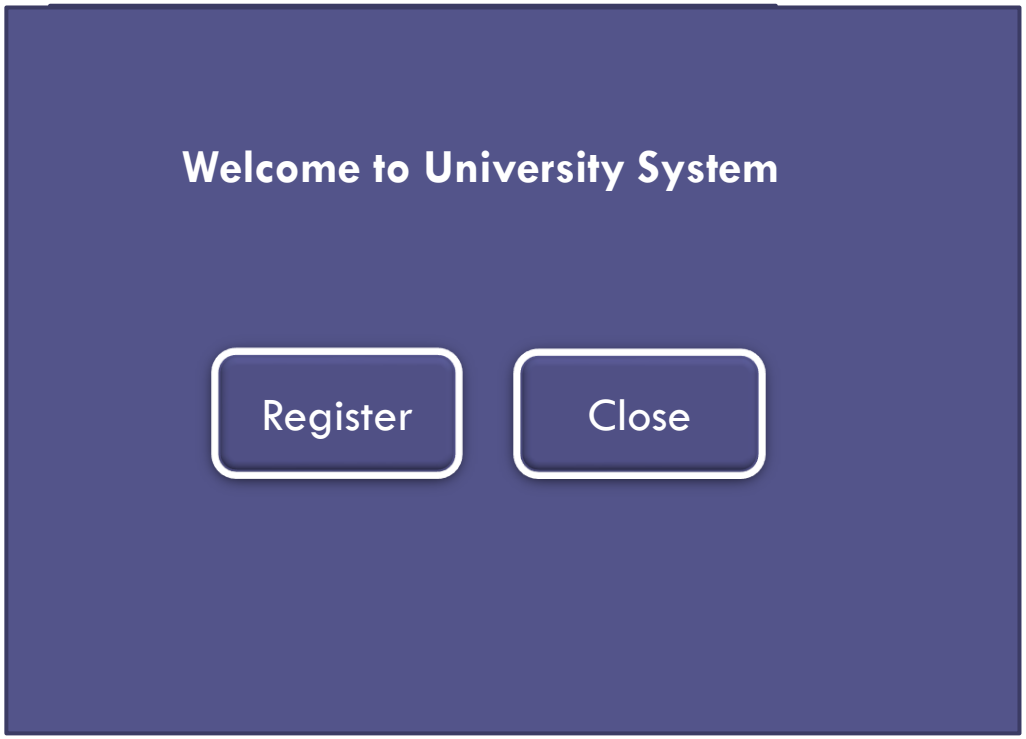


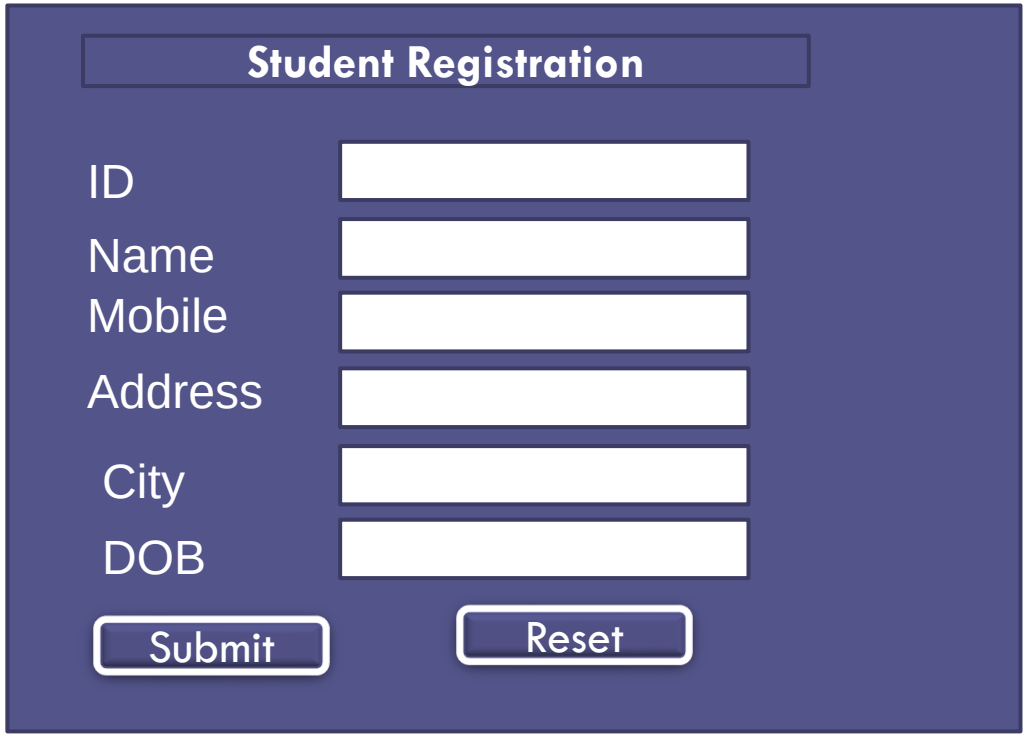
- **Database:**
  - Collection of related data.
  - A database contains a model of something!
- **A Database Management System (DBMS):** is a software system designed to store, manage and facilitate access to the database
- **Database/Application Program:** A computer program that interacts with the database through the DBMS

Components of a Database System

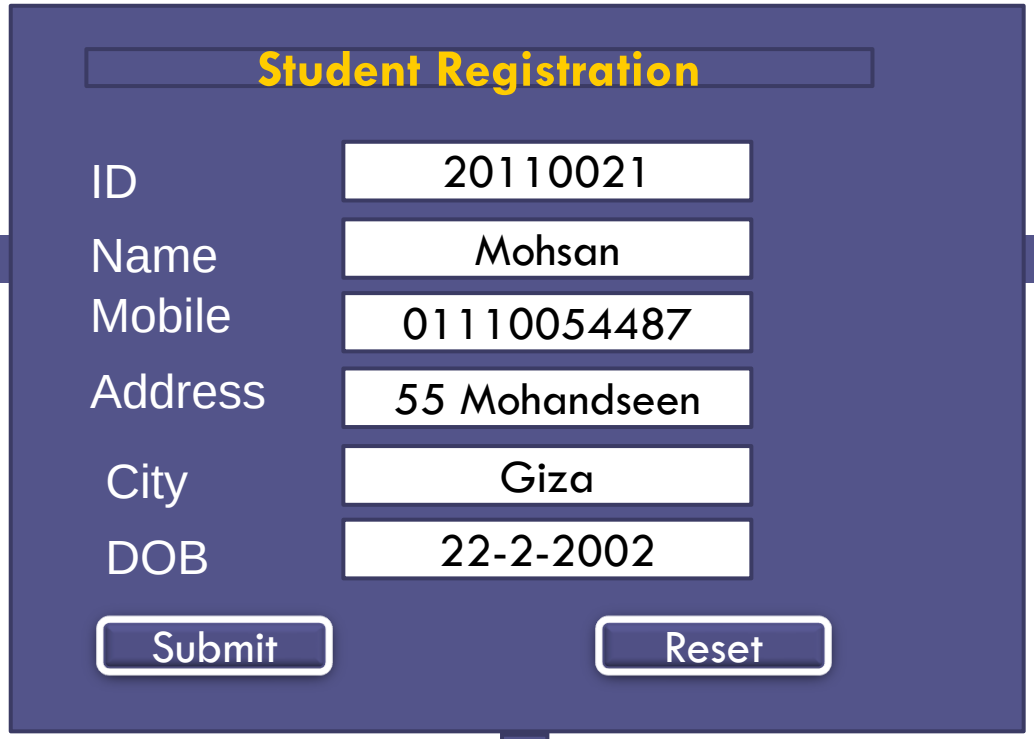


Example




Student Registration
ID
Name
Mobile
Address
City
DOB
Submit
Reset

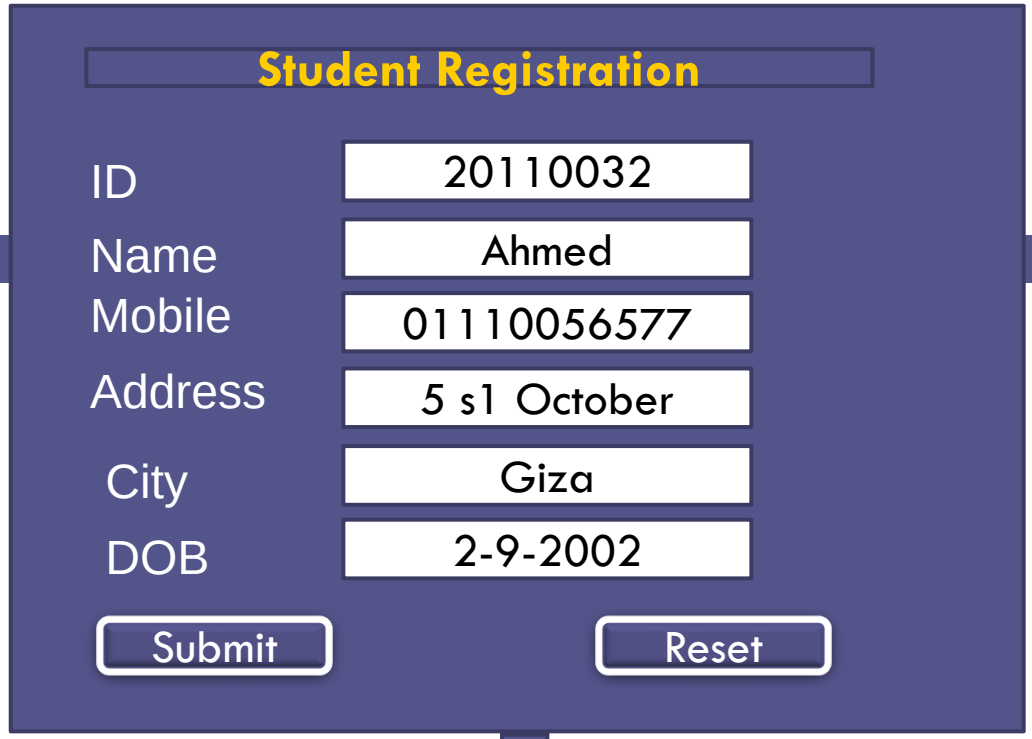
Application



DBMS

DB

ID	Name	Mobile	Address	City	DOB
20110021	Mohsan	01110054487	55 Mohandseen	Giza	22-2-2002



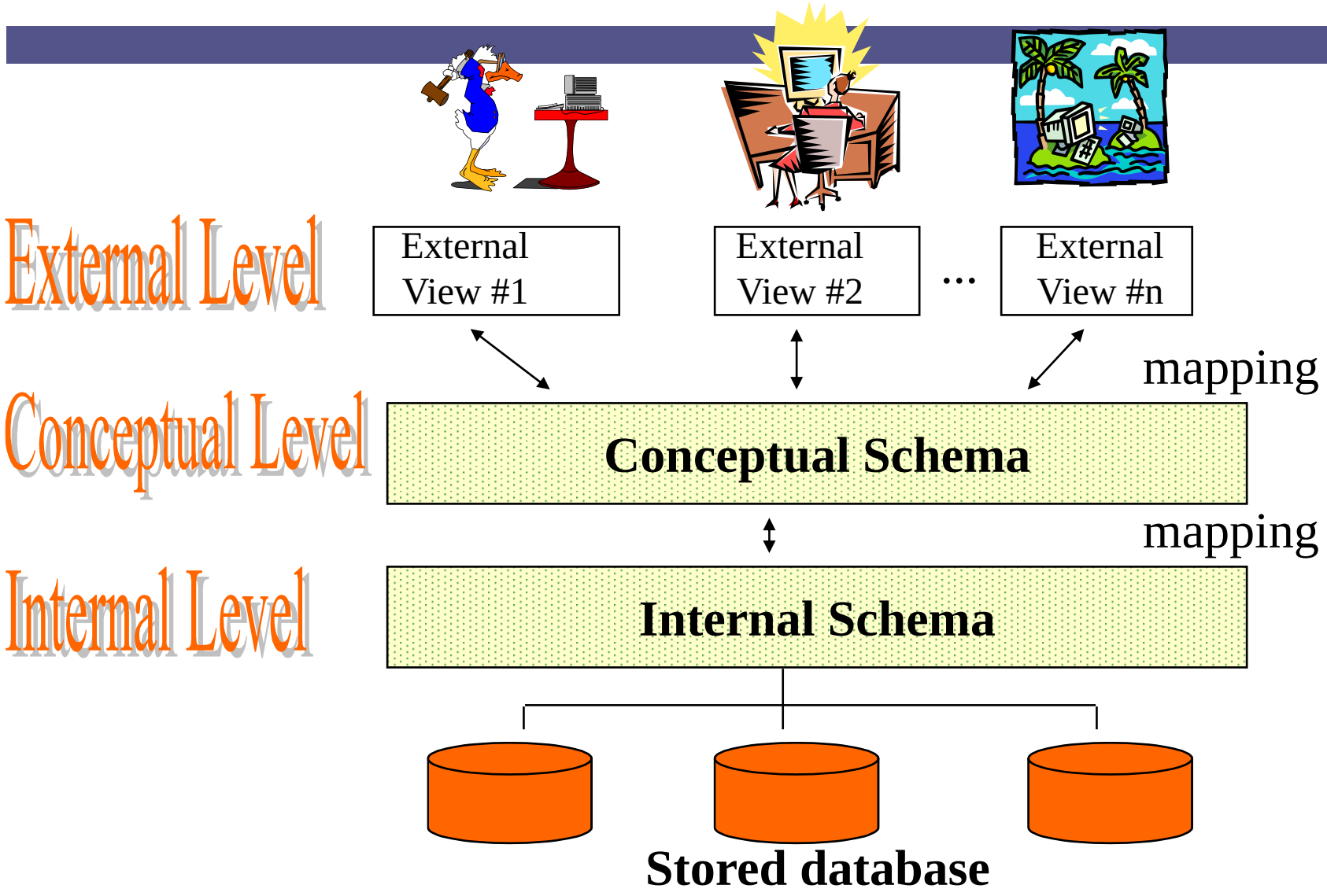
DBMS

ID	Name	Mobile	Address	City	DOB
20110021	Mohsan	01110054487	55 Mohandseen	Giza	22-2-2002
20110032	Ahmed	01110056577	5 s1 October	Giza	2-9-2002

Data Independence

- Does data have to be part of programs?
- Do we need to change one if the other changes?
- Three-tier architecture of databases

Three-Schema Architecture



Levels of Abstraction

- Views

  define how users see the data

  (use of data)

- Conceptual schema

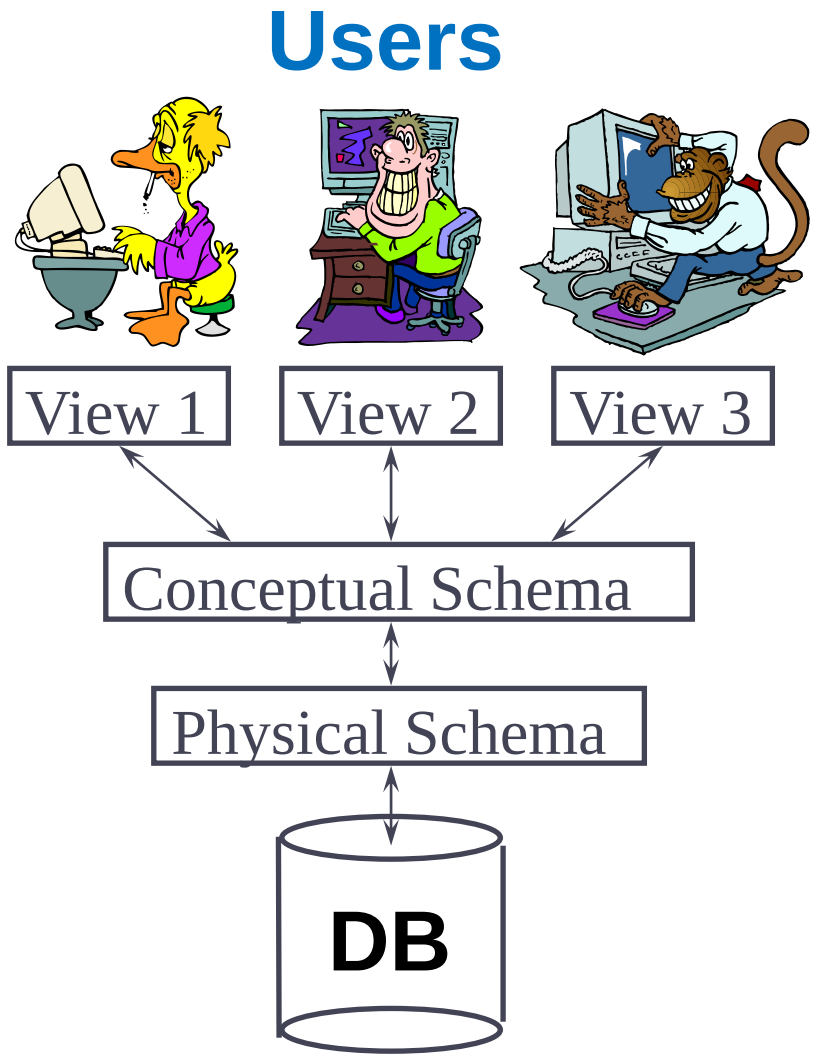
  defines logical structure

  (meaning of data)

- Physical schema

  describes the files and indexes used.

  (storage of data)



Three-Schema Architecture

- **Defines DBMS schemas at *three* levels:**
  - **Internal schema** is used to describe physical storage structures and access paths (e.g indexes).
    - Typically uses a physical data model.
  - **Conceptual schema** is used to describe the structure and constraints for the whole database for a community of users.
  - **External schemas** is used to describe the various user views.

Data Models

A Map is a Model of Reality

Data Models

- A collection of tools for describing
  - data
  - data relationships
  - data constraints
- Models:
  - Relational Model
  - object-oriented model
  - semi-structured data models
  - Older models: network model and hierarchical model

History of Relational Model

- Introduced by Ted Codd in 1970 in a classic paper
- Ted Codd was an IBM Researcher
- Many database concepts & products based on this model

Relational Model

Relations

- A relational database is a set of **relations**
- Relations are basically tables of data
- Each row represents a record in the relation
- Each relation has a unique name in the database
- Each row in the table specifies a relationship between the values in that row

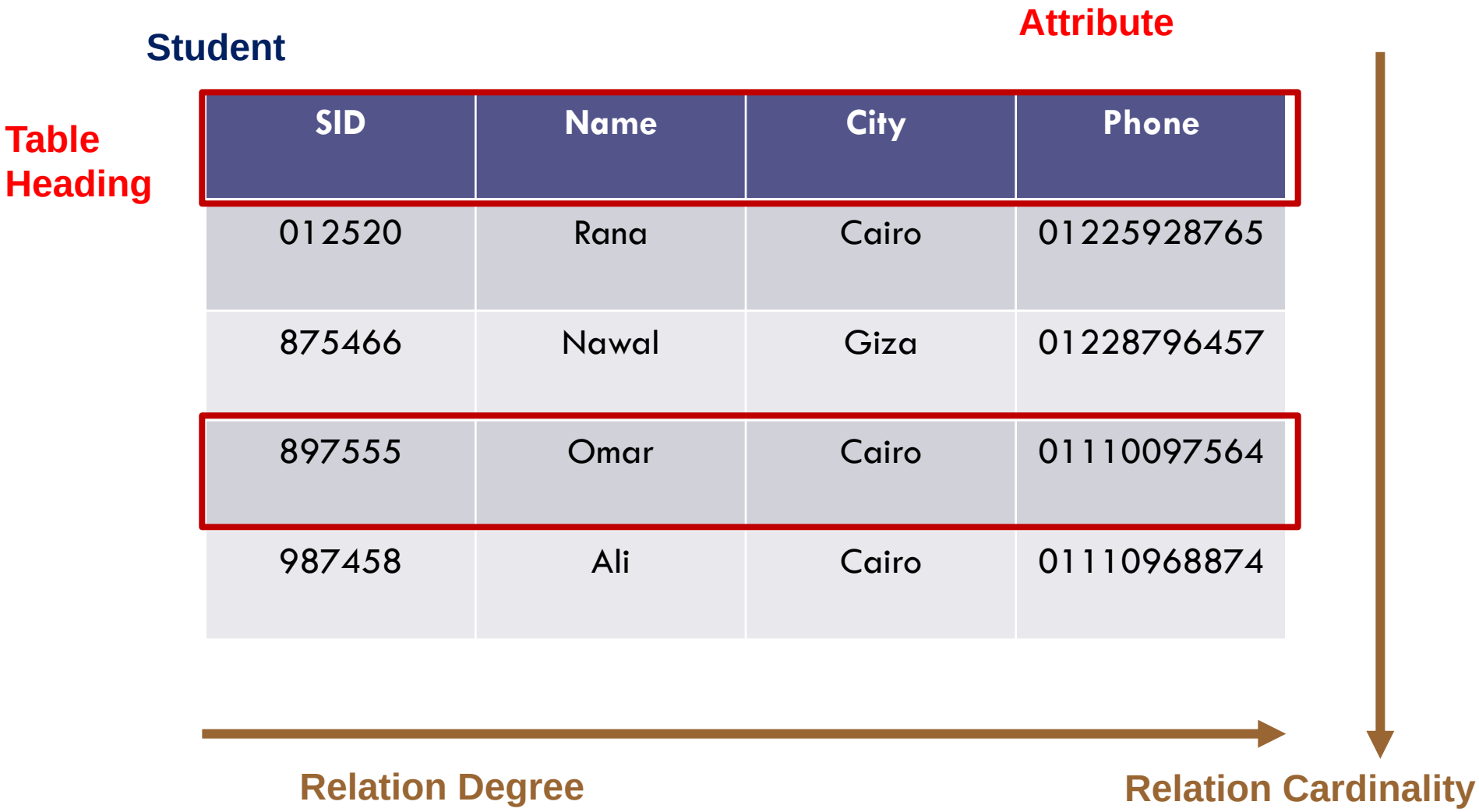
acct_id	branch_name	balance
A-301	New York	350
A-307	Seattle	275
A-318	Los Angeles	550
…	…	…

The *account* relation

- Example :

The account ID “A-307”, branch name “Seattle”, and balance “275” are all related to each other

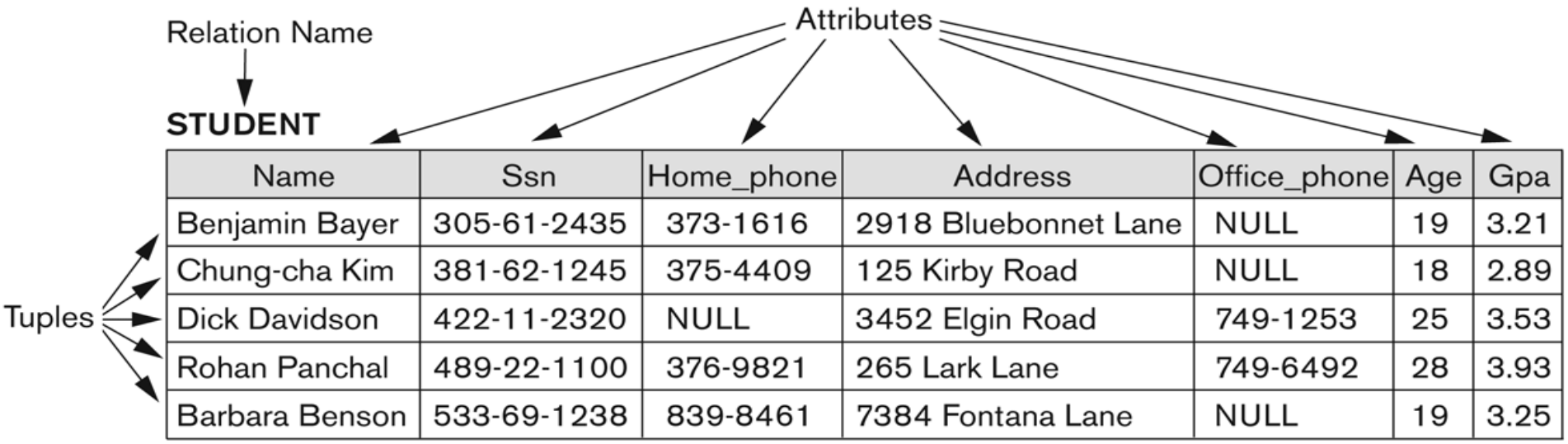
Relation in RDB



Definition Summary

Informal Terms		Formal Terms
Table		Relation
Column Header		Attribute
All possible Column Values		Domain
Row		Tuple
Table Definition		Schema of a Relation
Populated Table		State of the Relation

Relation Example



Name	Ssn	Home_phone	Address	Office_phone	Age	Gpa
Benjamin Bayer	305-61-2435	373-1616	2918 Bluebonnet Lane	NULL	19	3.21
Chung-cha Kim	381-62-1245	375-4409	125 Kirby Road	NULL	18	2.89
Dick Davidson	422-11-2320	NULL	3452 Elgin Road	749-1253	25	3.53
Rohan Panchal	489-22-1100	376-9821	265 Lark Lane	749-6492	28	3.93
Barbara Benson	533-69-1238	839-8461	7384 Fontana Lane	NULL	19	3.25

The attributes and tuples of a relation STUDENT.

Degree =7
Cardinality=5

Relations and Attributes

- Each relation has some number of attributes Sometimes called "columns"
- Each attribute has a domain Specifies the set of valid values for the attribute

acct_id	branch_name	balance
A-301	New York	350
A-307	Seattle	275
A-318	Los Angeles	550
…	…	…

The *account* relation

- The *account* relation:
  - 3 attributes
  - Domain of *balance* is the set of nonnegative integers
  - Domain of *branch_name* is the set of all valid branch names in the bank

DOMAINS

- A domain D is a set (pool) of values, from which one or more attributes takes their values.
- Example

CITY = {London, Paris, Doha, Cairo, Athens, Rome, Dobai, Madrid}

CITY is a pool of cities from which The attributes Supplier.City, Customer.City take their own values.

DATE = (DAY, MONTH, YEAR)

Where:

DAY = {1..31}, MONTH = {1..12}, YEAR = {1990..2100}

CITY is a simple domain, but DATE is a **composite** Domain

Tuples and Attributes

- Each row is called a tuple
  - A fixed-size, ordered set of name-value pairs
- Each attribute in the tuple has a unique name

acct_id	branch_name	balance
A-301	New York	350
A-307	Seattle	275
A-318	Los Angeles	550
…	…	…

The *account* relation

Tuples and Relations

- A relation is a set of tuples
- Each tuple appears **exactly once**
- The order of tuples in a relation is not relevant

Instructor

ID	*name*	*dept_name*	*salary*
22222	Einstein	Physics	95000
12121	Wu	Finance	90000
32343	El Said	History	60000
45565	Katz	Comp. Sci.	75000
98345	Kim	Elec. Eng.	80000
76766	Crick	Biology	72000
10101	Srinivasan	Comp. Sci.	65000
58583	Califieri	History	62000
83821	Brandt	Comp. Sci.	92000
15151	Mozart	Music	40000
33456	Gold	Physics	87000
76543	Singh	Finance	80000

Schema VS Instance

- the name of the relation and the set of attributes is called the **schema**
- the current values contained in the relation represent an **instance**

Relation Schemas

- Every relation has a schema
- A relation schema includes:
  - an ordered set of attributes
  - the domain of each attribute

acct_id	branch_name	balance
A-301	New York	350
A-307	Seattle	275
A-318	Los Angeles	550
…	…	…

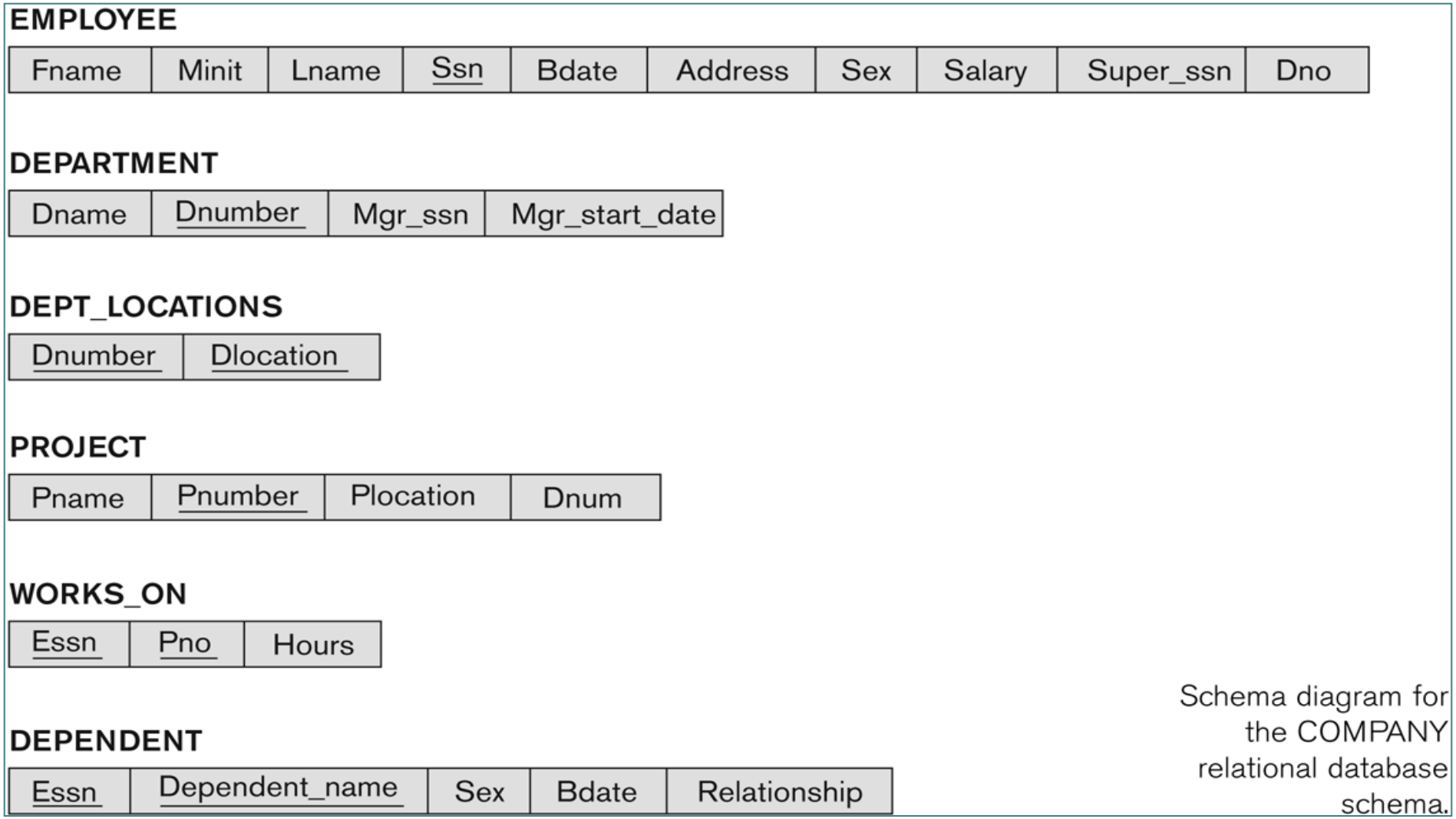
The *account* relation

- The relation schema of *account* is:

Account_schema = (*acct_id*, *branch_name*, *balance*)

- Database schema is a collection of relation schemas

COMPANY Database Schema



Schema diagram for the COMPANY relational database schema.