MODULE I
RELATIONAL DATABASES

- DBMS is a collection of interrelated data and a set of programs to access those data.
- The Database(collection of data) contains information relevant to an enterprise.
- The primary goal of a DBMS is to provide a way to store and retrieve database information that is both ***convenient*** **and** ***efficient***.

- Database systems are designed to manage large bodies of information.
- Management of data involves both *defining structures* for storage of information and *providing mechanisms for the manipulation* of information.

- The database system must ensure the safety of the information stored, despite system crashes or attempts at unauthorized access.
- If data are to be shared among several users, the system must avoid possible anomalous results.

Database systems are used to manage collections of data that:

- are highly valuable,
- are relatively large, and
- are accessed by multiple users and applications, often at the same time.

APPLICATIONS

Enterprise Information

- Sales:

  For customer, product, and purchase information.

- Accounting: For payments, receipts, account balances, assets, and other accounting information.
- Human resources: For information about employees, salaries, payroll taxes, and benefits, and for generation of paychecks.

Manufacturing:

For management of the supply chain and for tracking production of items in factories, inventories of items in warehouses and stores, and orders for items.

Banking and Finance

- Banking: For customer information, accounts, loans, and banking transactions.
- Credit card transactions: For purchases on credit cards and generation of monthly statements.

- Finance: For storing information about holdings, sales, and purchases of financial instruments such as stocks and bonds; also for storing real-time market data to enable online trading by customers and automated trading by the firm.

Universities: For student information, course registrations, and grades

Airlines: For reservations and schedule information. Airlines were among the first to use databases in a geographically distributed manner.

Telecommunication: For keeping records of calls, texts, and data usage, generating monthly bills, maintaining balances on prepaid calling cards, and storing information about the communication networks.

Web-based services

- Social-media: For keeping records of users, connections between users, posts made by users, rating/like information about posts, etc.

- Online retailers: For keeping records of sales data and orders as for any retailer, but also for tracking a user's product views, search terms, etc., for the purpose of identifying the best items to recommend to that user.
- Online advertisements: For keeping records of click history to enable targeted advertisements, product suggestions, news articles, etc. People access such databases every time they do a web search, make an online purchase, or access a social-networking site.

- **Document databases:** For maintaining collections of new articles, patents, published research papers, etc.
- **Navigation systems:** For maintaining the locations of varies places of interest along with the exact routes of roads, train systems, buses, etc.

Modes

- The first mode is to support online transaction processing, where a large number of users use the database, with each user retrieving relatively small amounts of data, and performing small updates.
- The second mode is to support data analytics, that is, the processing of data to draw conclusions, and infer rules or decision procedures, which are then used to drive business decisions.

PURPOSE OF DATABASE SYSTEMS

- Consider part of a university organization,keeps information about all instructors, students, departments and courses.
- One way to keep the information on a computer is to store it in operating-system files.
- To allow users to manipulate the information, the system has a number of application programs that manipulate the files, including programs to:

- Add new students, instructors, and courses.
- Register students for courses and generate class rosters.
- Assign grades to students, compute grade point averages (GPA), and generate transcripts.
- Programmers develop these application programs to meet the needs of the university.
- New application programs are added to the system as the need arises.

- This file-processing system is supported by a conventional operating system.
- The system stores permanent records in various files, and it needs different application programs to extract records from, and add records to, the appropriate files.
- Keeping organizational information in a file-processing system has a number of major disadvantages:

- **Data redundancy and inconsistency**. Since different programmers create the files and application programs over a long period, the various files are likely to have different structures, and the programs may be written in several programming languages.

- **Difficulty in accessing data.** Suppose that one of the university clerks needs to find out the names of all students who live within a particular postal-code area.

  The university clerk now has two choices: either obtain the list of all students and extract the needed information manually or ask a programmer to write the necessary application program.

- **Data isolation**. Because data are scattered in various files, and files may be in different formats, writing new application programs to retrieve the appropriate data is difficult.
- **Integrity problems.** The data values stored in the database must satisfy certain types of consistency constraints.
- **Atomicity problems.** The funds transfer must be *atomic—it must happen in its entirety or not at all. It* is difficult to ensure atomicity in a conventional file-processing system.

- **Concurrent-access anomalies**. For the sake of overall performance of the system and faster response, many systems allow multiple users to update the data simultaneously.
- **Security problem**s. Not every user of the database system should be able to access all the data. For example, in a university, payroll personnel need to see only that part of the database that has financial information. They do not need access to information about academic records. But since application programs are added to the file-processing system in an ad hoc manner, enforcing such security constraints is difficult.

VIEW OF DATA

- A database system is a collection of interrelated data and a set of programs that allow users to access and modify these data.
- A major purpose of a database system is to provide users with an *abstract view of the data.*
- *The system hides certain details* of how the data are stored and maintained.

1.3.1 Data Models

1.3.2 Relational Data Model

1.3.3 Data Abstraction

1.3.1 Data Models

-underlying the structure of a database .

-collection of conceptual tools for describing data, data relationships, data semantics, and consistency constraints.

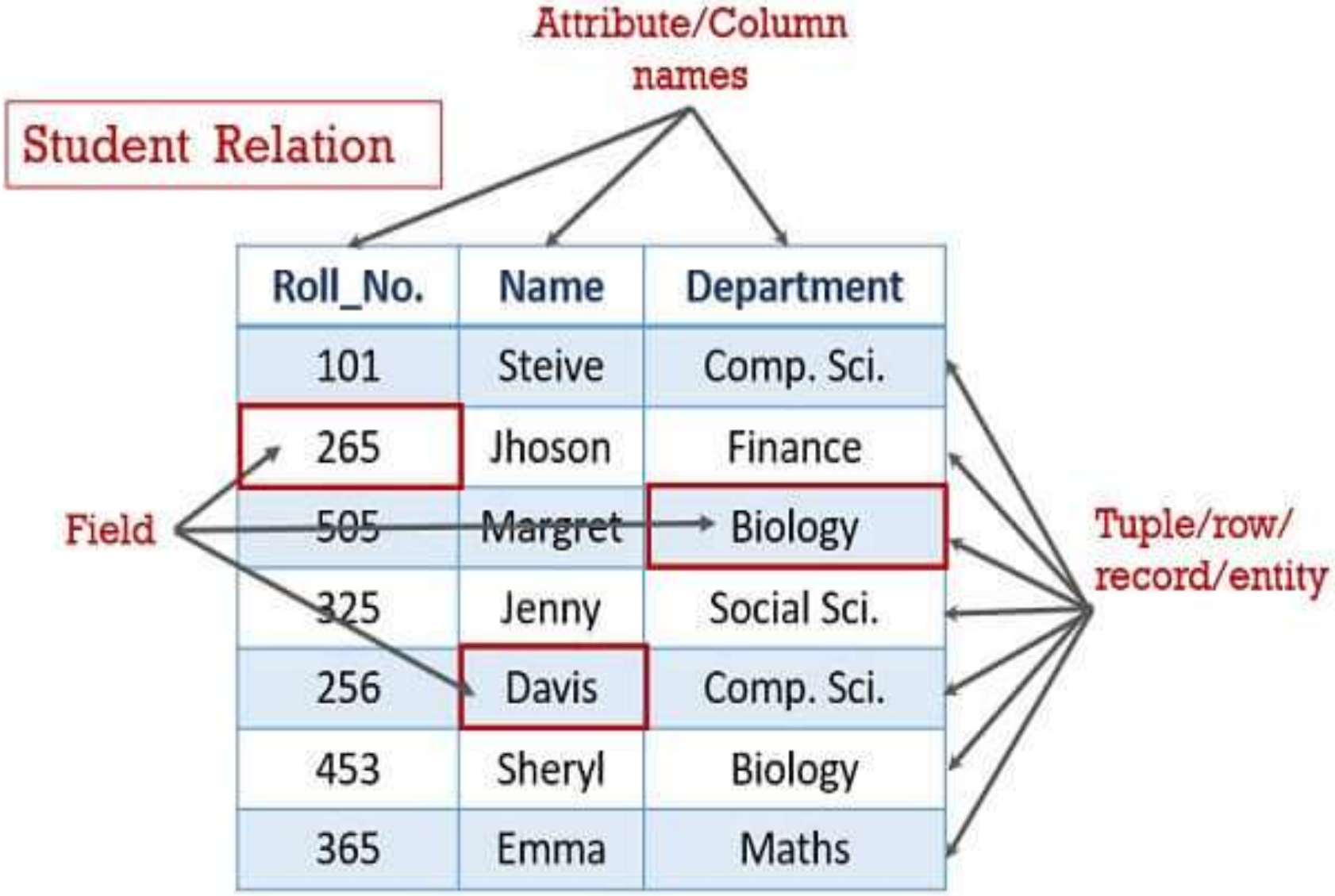
-It's classified into four different categories:

i. Relational Model.
ii. Entity-Relationship Model.
iii. Semi-structured Data Model
iv. Object-Based Data Model.

i.Relational Model.

- It uses a collection of tables to represent both data and the relationships among those data.
- Each table has multiple columns, and each column has a unique name.
- Tables are also known as relations.
- The relational model is an example of a record-based model which is structured in fixed-format records of several types

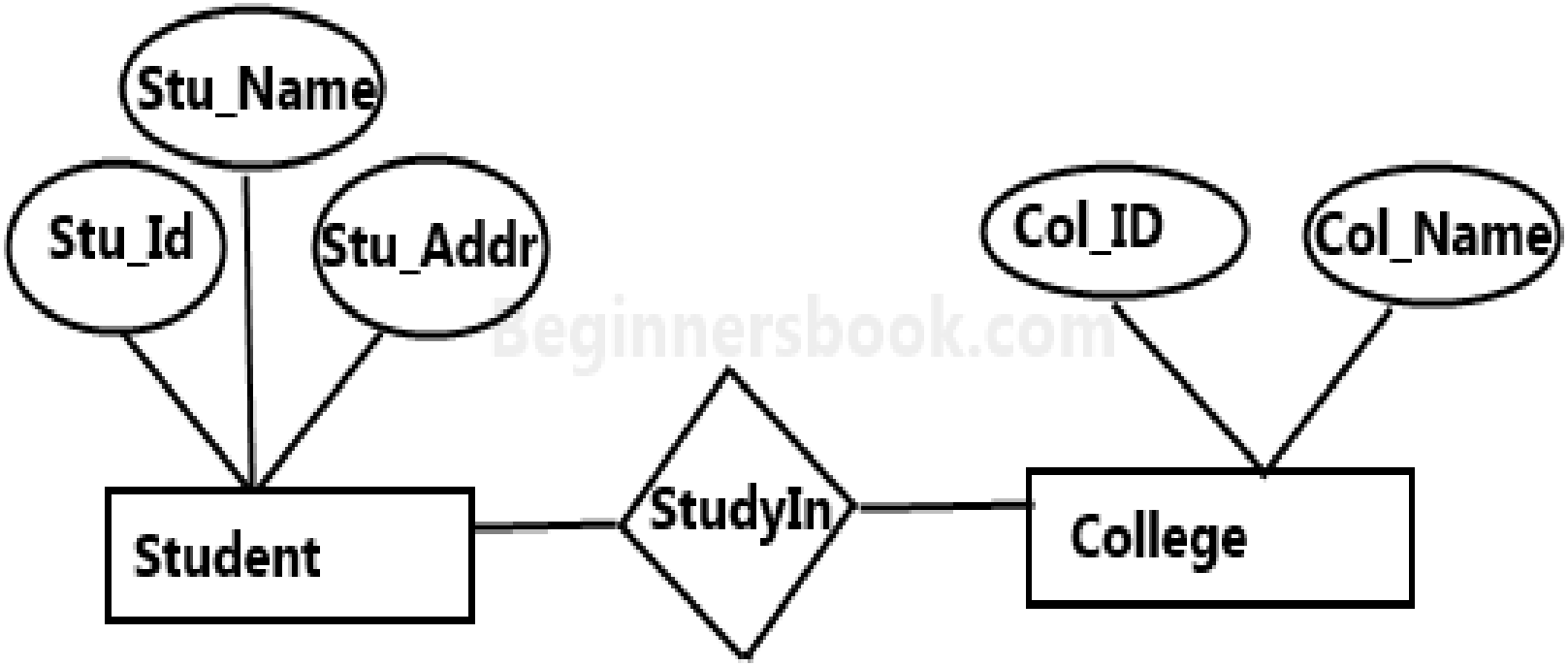
- Each table contains records of a particular type.
- Each record type defines a fixed number of fields, or attributes.
- The columns of the table correspond to the attributes of the record type.



Student Relation in Relational Model

ii. Entity-Relationship Model.

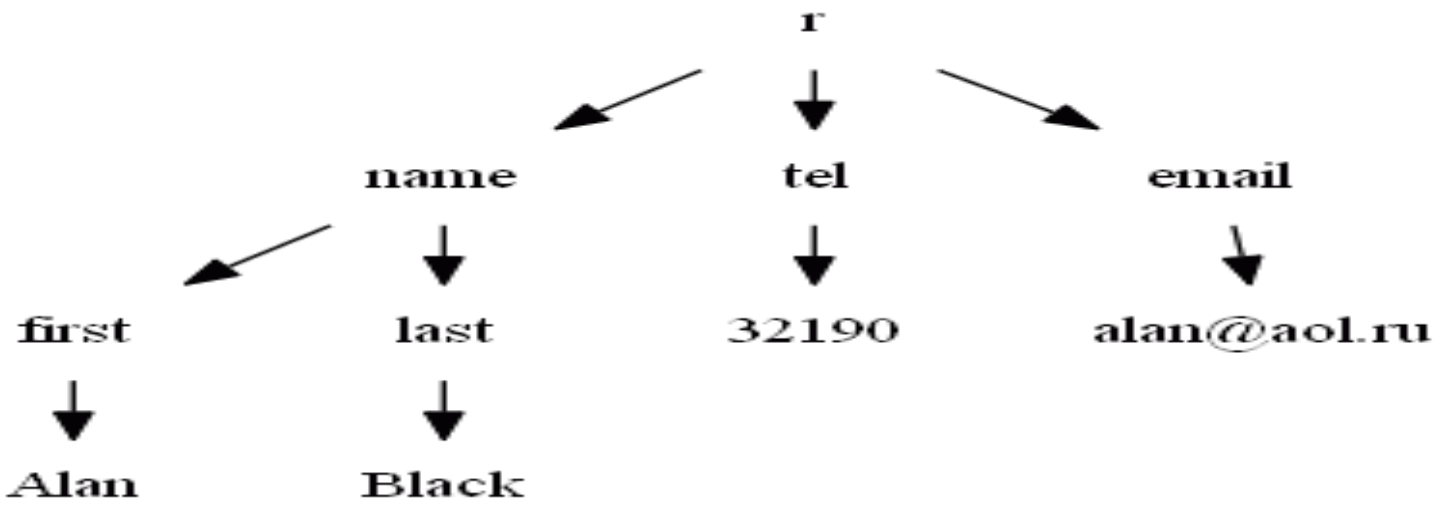
- It uses a collection of basic objects, called *entities, and relationships among these objects.*
- *An entity* is a “thing” or “object” in the real world that is distinguishable from other objects.
- The entity-relationship model is widely used in database design.



Sample E-R Diagram

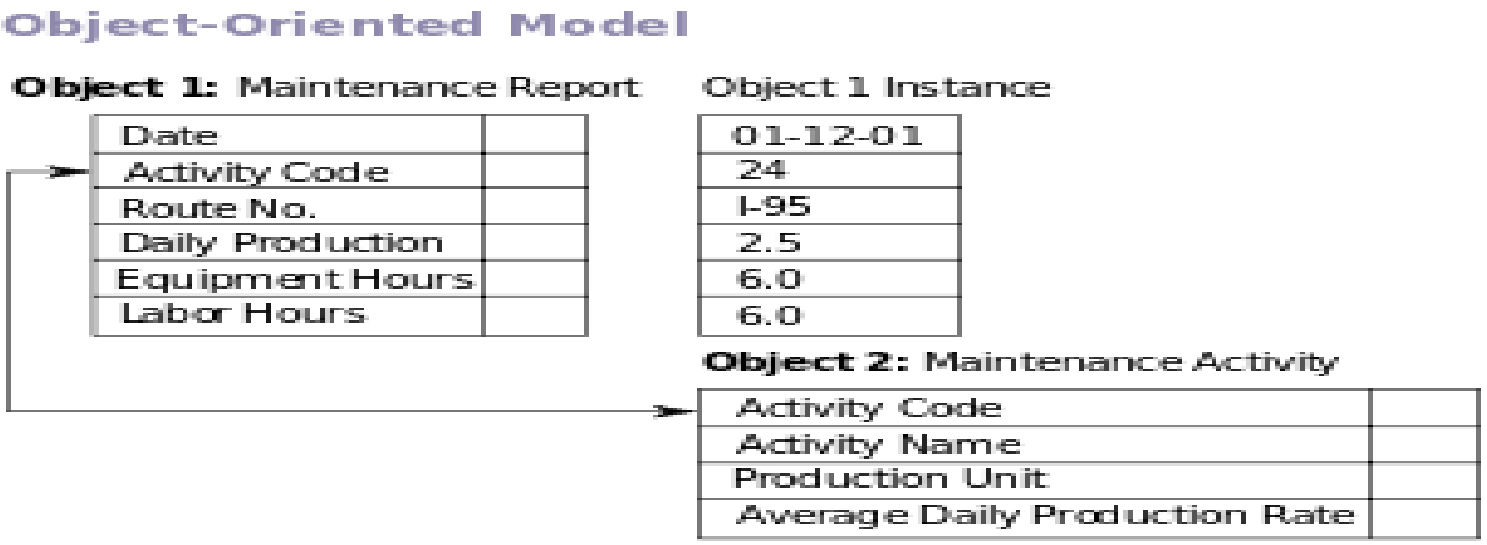
iii. Semi-structured Data Model.

- It permits the specification of data where individual data items of the same type may have different sets of attributes.
- Example: *JSON and Extensible Markup Language (XML)*



iv. Object-Based Data Model.

- Objects are well integrated into relational databases.Standards exist to store objects in relational tables.
- Database systems allow procedures to be stored in the database system and executed by the database system



1.3.2 Relational Data Model

- Data are represented in the form of tables.
- Each table has multiple columns, and each column has a unique name.
- Each row of the table represents one piece of information.

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

(a) The *instructor* table

dept_name	*building*	*budget*
Comp. Sci.	Taylor	100000
Biology	Watson	90000
Elec. Eng.	Taylor	85000
Music	Packard	80000
Finance	Painter	120000
History	Painter	50000
Physics	Watson	70000

(b) The *department* table

Figure 1.1 A sample relational database.

1.3.3 Data Abstraction

- The need for efficiency has led database system developers to use complex data structures to represent data in the database.
- There are several levels of data abstraction:

i. Physical level. The lowest level of abstraction describes *how the data are actually* stored. The physical level describes complex low-level data structures in detail.

- Logical level. The next-higher level of abstraction describes *what data are stored* in the database, and what relationships exist among those data.
- The simple structures at the logical level may involve complex physical-level structures, the user of the logical level does not need to be aware of this complexity. This is referred to as physical data independence.

iii. View level. The highest level of abstraction describes only part of the entire database.

- Many users of the database system do not need all this information; instead, they need to access only a part of the database.
- The view level of abstraction exists to simplify their interaction with the system.
- The system may provide many views for the same database.

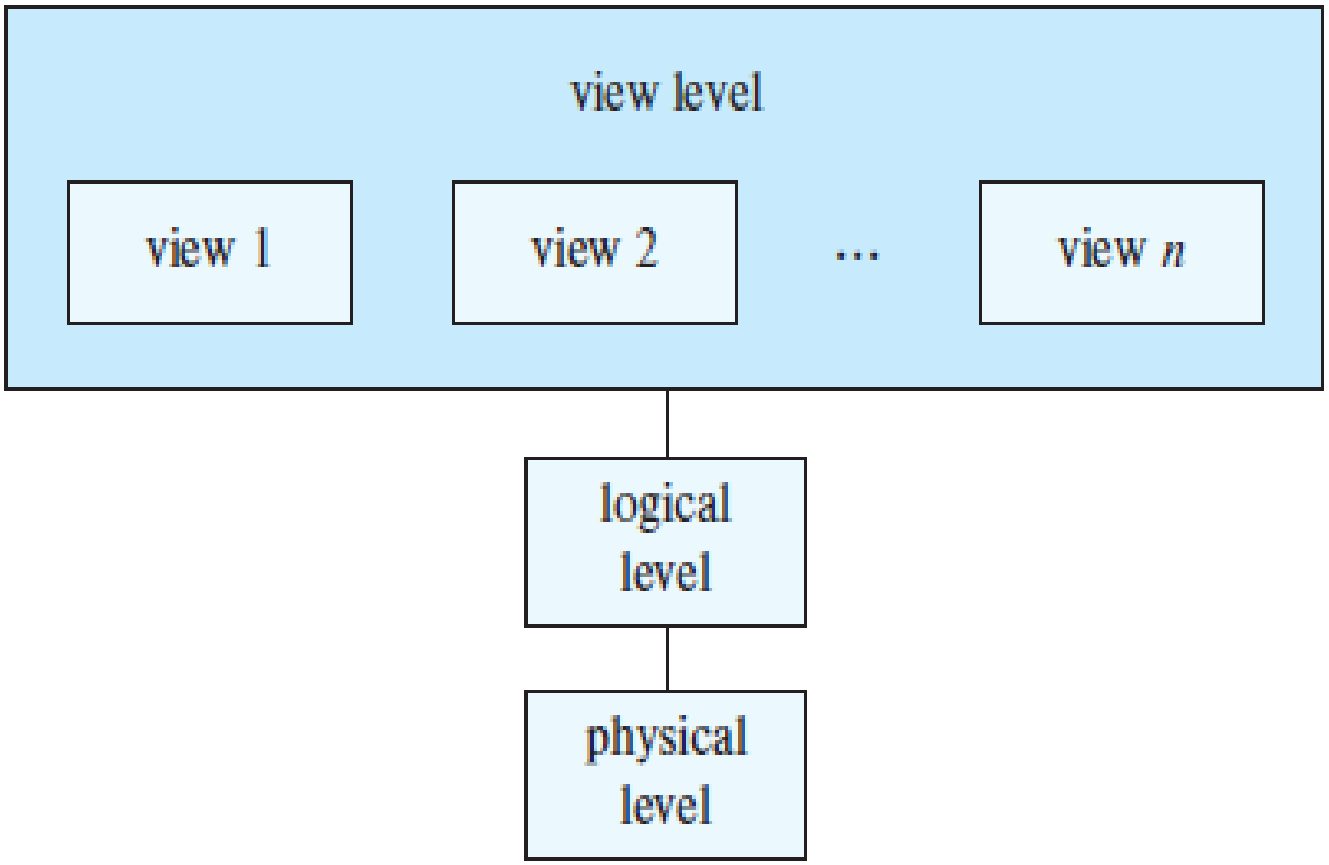


Figure 1.2 The three levels of data abstraction.

1.3.4 Instances and Schemas

- The collection of information stored in the database at a particular moment is called an instance of the database.
- The overall design of the database is called the database schema.
- A database schema corresponds to the variable declarations in a program.

- Each variable has a particular value at a given instant.
- The values of the variables in a program at a point in time correspond to an *instance of a database schema.*
- The physical schema describes the database design at the physical level, while the logical schema describes the database design at the logical level.
- A database may also have several schemas at the view level, sometimes called sub schemas, that describe different views of the database.