Database Systems

CO3201

Lecture 01 : Introduction

LECTURER : DILANI RANAWEERA

DEPARTMENT OF COMPUTER ENGINEERING

Course Information

1. Lectures

   Monday 8.00 -10.00 am

2. Laboratory session

   Friday 8.00am - 11.00am(CO students), Friday 1.00 pm - 4.00pm(EE students)

3. References

   - *Elmasri and Navathe.* ***Fundamentals of Database Systems****. Addison-Wesley, 6th Edition, 2011.*
   - *Thomas Connolly•Carolyn Begg,* ***Database Systems****, A Practical Approach to Design, Implementation, and Management, 4th Edition , 2005*
   - ***Modern Database Management****, Jeffrey A. Hoffer, V.Ramesh, Heikki Topi, 10th edition*

Course Evaluation

Continuous Assessment	40%
i. Tutorials/quizzes/assignments/Practical Exam	
Semester – End Theory Examination	60%

Eligibility criteria for Sitting for the End-Semester Examination

- Attendance for Lectures :- ≥ 80%
- Attendance for Practical Work :- All are Compulsory
- Total CA marks :- ≥40%

Course Outline

1. Database design process and ER model
2. Relational model
3. ER model to relational model
4. Normalization
5. Relational algebra
6. Query Language
7. Database applications

Introduction

Introduction

Traditional(manual) systems

- Most of the information that is stored and accessed is either textual or numeric
- data are stored as hand written forms and printed copies

Ex : Telephone directory , library systems , etc..

Disadvantages

- Time wasting
- more space
- More paper work

Contd…

- **Terms**
  - **Data** : a collection of raw facts and figures that can be processed by any computing machine
    - Ex :Perera, EMP001
  - **Information** : processed data or systematic/meaningful form of data
    - Ex : EMP001 is Perera
  - **Metadata** : data about data or Data that describe the properties or characteristics of other data.
    - Ex : Employee

Name	**Type**	**Length**	**Min**	**Max**	**Description**
EmpNo	Number	9			Employee No.
Name	Character	30			Employee Name
Dept	Character	10			Dept. No.
Salary	Number	8	5000	60000	Employee Salary

Contd…

Database :

- A **database** is a collection of related data.
- A database represents some aspect of the real world, sometimes called the **mini-world** or the **universe of discourse (UoD)**. Changes to the **mini-world** are reflected in the database.
- A database is a logically coherent collection of data with some inherent meaning.

Contd…

- Database Management System (**DBMS**)
  - collection of programs that enables users to **create and maintain a database**
  - general-purpose software system that facilitates the processes of definig, constructing, manupulating and sharing databases among various users and applications
    - **Defining** a database involves specifying the data types, structures and constraints of the data to be stored in the database.
      - The database definition or descriptive information is also stored by the DBMS in the form of a database catalog or dictionary ; called **meta-data**.
    - **Constructing** the database is the process of storing the data on some storage medium that is controlled by the DBMS.
    - **Manipulating** a database includes functions such as querying the database to retrieve specific data, updating the database to reflect changes in the miniworld, and generating reports from the data.
      - A **query** typically causes some data to be retrieved;
      - a **transaction** may cause some data to be read and some data to be written into the database.
    - **Sharing** a database allows multiple users and programs to access the database simultaneously.

Contd…

- An application program accesses the database by sending queries or requests for data to the DBMS.
- DBMS include **protecting the database and maintaining** it over a long period of time.
  - Protection includes system protection against hardware or software malfunction (or crashes) and security protection against unauthorized or malicious access

Contd...

- Database System
  - **database and DBMS software together** call as database system

  Ex :

  - Library systems
  - Airlines reservations systems
  - Payroll systems
  - University MIS
  - Hotel reservation systems etc..

Contd…

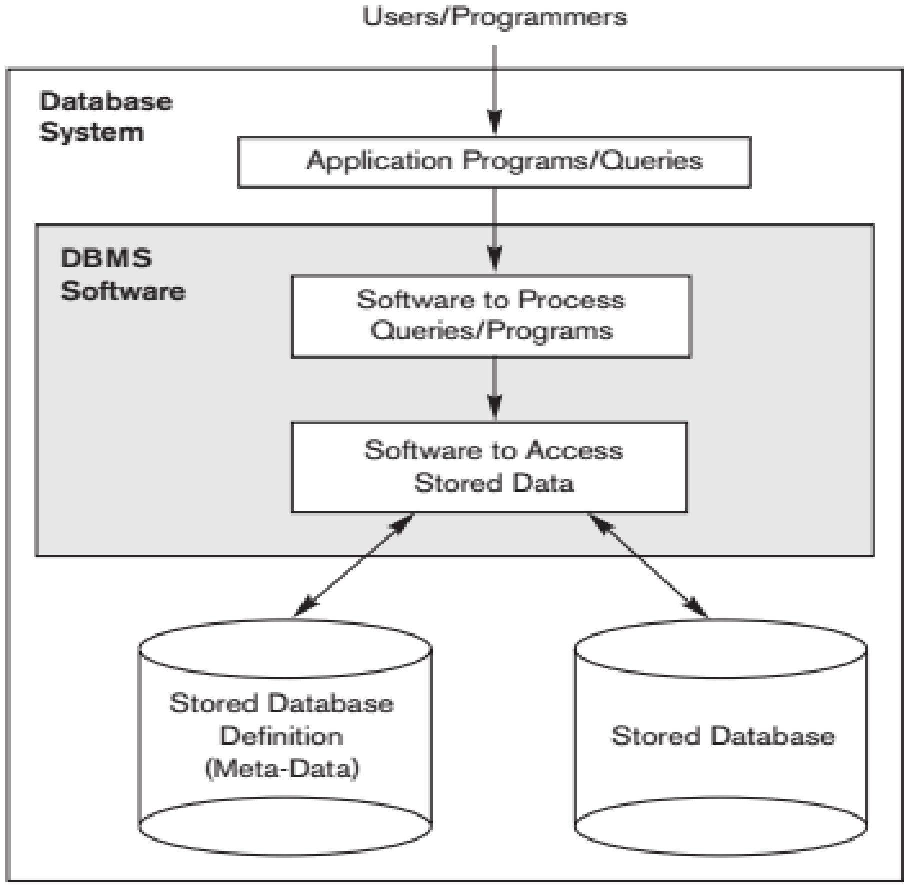
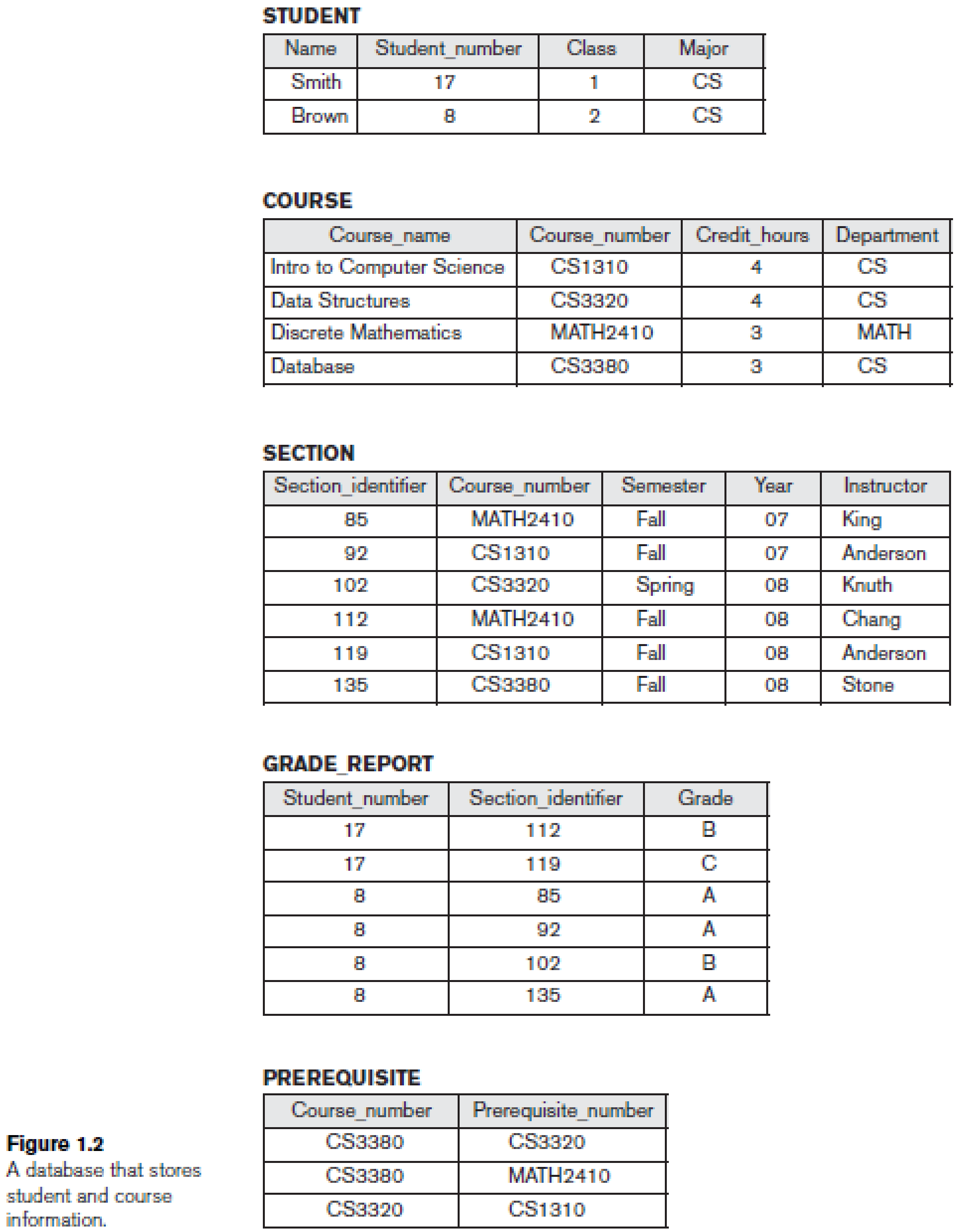

Users/Programmers
Database System
Application Programs/Queries
DBMS Software
Software to Process Queries/Programs
Software to Access Stored Data
Stored Database Definition (Meta-Data)
Stored Database

Figure 1.1
A simplified database system environment.

Contd…

Advantages of using database approach

- Controlling Redundancy
- Restricting unauthorized access
- Provide backup and recovery
- providing multiple user interfaces



STUDENT

Name	Student_number	Class	Major
Smith	17	1	CS
Brown	8	2	CS

COURSE

Course_name	Course_number	Credit_hours	Department
Intro to Computer Science	CS1310	4	CS
Data Structures	CS3320	4	CS
Discrete Mathematics	MATH2410	3	MATH
Database	CS3380	3	CS

SECTION

Section_identifier	Course_number	Semester	Year	Instructor
85	MATH2410	Fall	07	King
92	CS1310	Fall	07	Anderson
102	CS3320	Spring	08	Knuth
112	MATH2410	Fall	08	Chang
119	CS1310	Fall	08	Anderson
135	CS3380	Fall	08	Stone

GRADE_REPORT

Student_number	Section_identifier	Grade
17	112	B
17	119	C
8	85	A
8	92	A
8	102	B
8	135	A

PREREQUISITE

Course_number	Prerequisite_number
CS3380	CS3320
CS3380	MATH2410
CS3320	CS1310

Figure 1.2
A database that stores student and course information.

Schemas, Instances, and Database State

Database schema - the description of a database , which is specified during database design and is not expected to change frequently

- A displayed schema is called a **schema diagram**

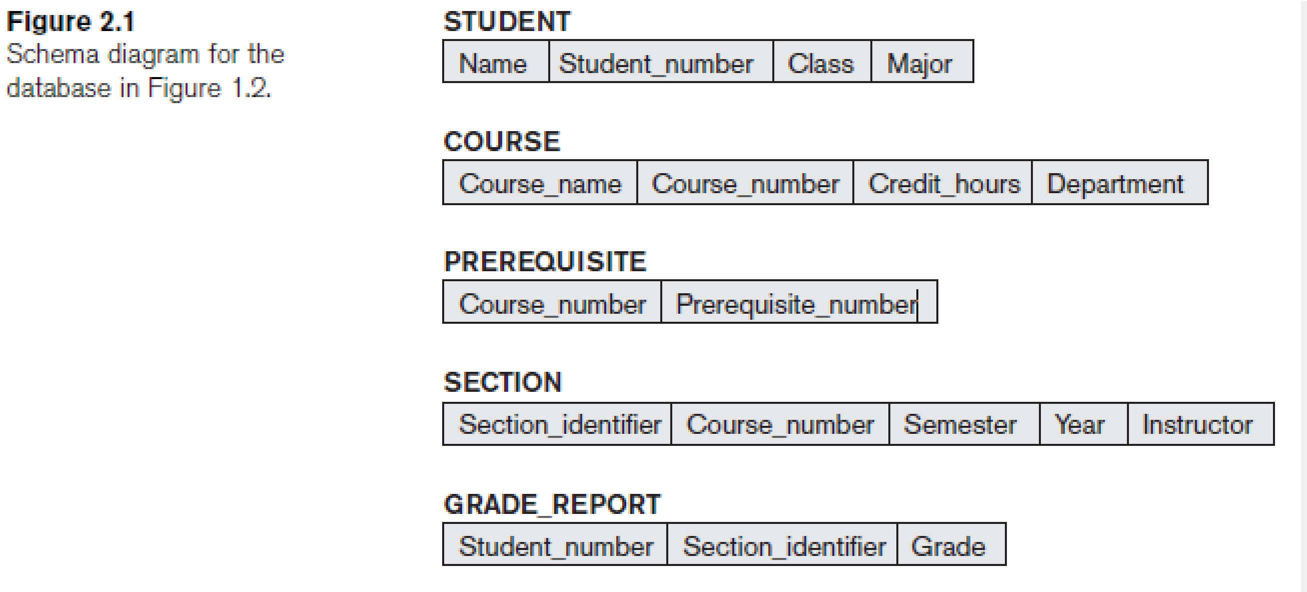


Figure 2.1
Schema diagram for the database in Figure 1.2.

STUDENT

Name	Student_number	Class	Major

COURSE

Course_name	Course_number	Credit_hours	Department

PREREQUISITE

Course_number	Prerequisite_number

SECTION

Section_identifier	Course_number	Semester	Year	Instructor

GRADE_REPORT

Student_number	Section_identifier	Grade

Schemas, Instances, and Database State

Instances - each schema construct has its own *current set* of instances;

- for example, the STUDENT construct will contain the set of individual student entities (records) as its instances

Database state or **snapshot -** The data in the database at a particular moment in time is called

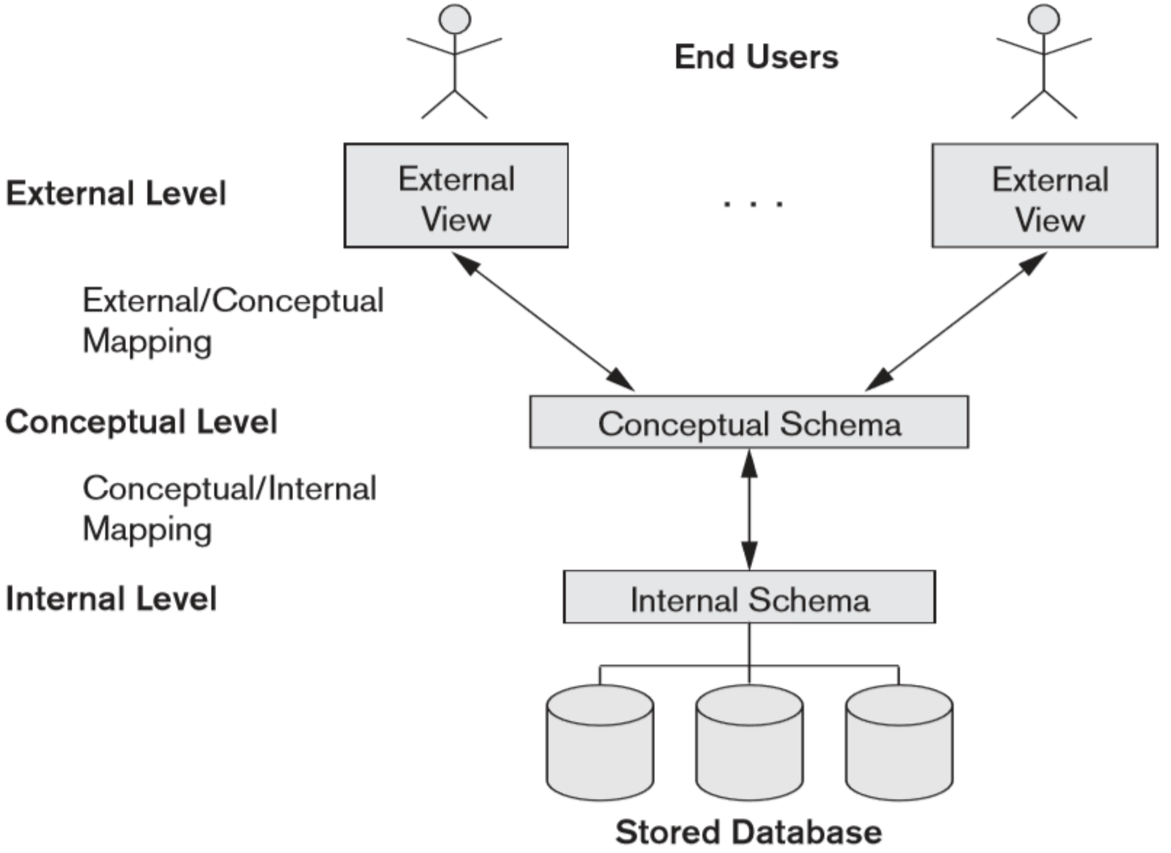
- It is also called the *current* set of **occurrences** or **instances** in the database
- **valid state**— satisfies the structure and constraints specified in the schema
- **define** a new database, we specify its database schema only to the DBMS . At this point, the corresponding database state is the ***empty state*** with no data.
- We get the ***initial state*** of the database when the database is first **populated** or **loaded** with the initial data.
- From then on, every time an update operation is applied to the database, we get another database state. At any point in time, the database has a ***current state***

Three-Schema Architecture

- The **goal of the three-schema** architecture is to **separate the user applications from the physical database.**
  - DBA should be able to change **database storage structures** without affecting the user's views.
  - DBA should be able to change **conceptual structure** of database without affecting all users.

Three-Schema Architecture

Figure 2.2
The three-schema architecture.



Three-Schema Architecture

1. Internal level
   - has an internal schema, which describes the **physical storage structure of the database**
   - Use to achieve optimal runtime performance and storage space utilization
   - Covers **data structures and file organizations** used to store data on the storage device
   - Storage space allocation for data and indexes

2. Conceptual level
   - has a conceptual schema, which describes the **structure of the whole database for a community of users**
   - **hides the details of physical storage structures** and **concentrates on describing entities, data types, relationships, user operations, and constraints**

Three-Schema Architecture

3. External level
   - Includes a number of **external schemas or user views**.
   - Describes the part of the database that is relevant to the user
   - Different views have different representations of the same data
   - External Views Allow to
     - hide unauthorized data •e.g. salary, dob
     - provide user view •e.g. view employee name, designation, department data taken from employee and department files
     - derive new attributes •e.g. age derived from dob or nid
     - change unit of measurement •e.g. show age in years or months
     - define security levels •e.g. update access to employee file read-only to department file

How to design a database ?

1. Identify Entities, relationships, attributes and keys
2. Design ER diagram(Conceptual design)
3. Convert ER design into relational schema
4. Normalization
5. Convert normalized relational schema into tables
6. Create tables in the DB

Entity and Entity sets

Entity

- a thing in the real world with an independent existence
- it may be an object with a **conceptual existence**
  - Ex: For university database --> courses
- may be an object with a **physical existence**
  - Ex : For university database --> students, lecturers

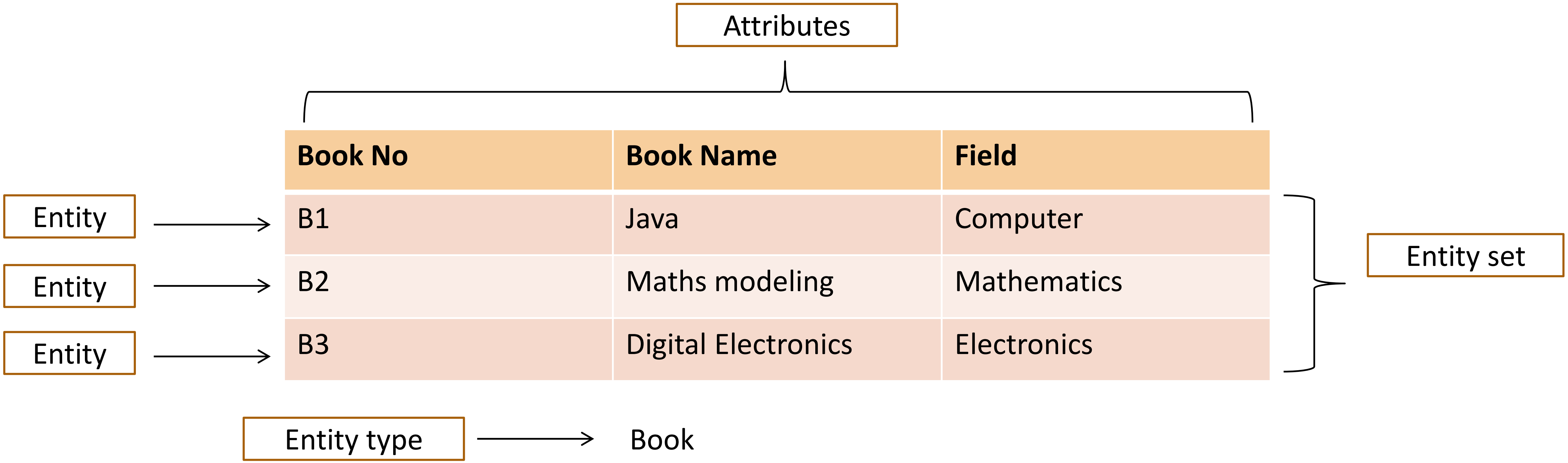
Entity sets

- The collection of all entities of a particular entity type in the database at any point in time
  - Ex : Student set may contain all the students in a university

Attributes

- the information about the entity that needs to be stored
- represents properties of an entity, that describes entity
- Each entity has **attributes**
  - Ex: Student entity may be described by the student’s name, registration number, DOB

Entity and attributes



Attributes

Book No	Book Name	Field
B1	Java	Computer
B2	Maths modeling	Mathematics
B3	Digital Electronics	Electronics

Entity

Entity

Entity

Entity set

Entity type → Book

Domain

- A set of permitted values
  - Ex :

    Academic_department_codes → ‘CO’,‘EE’ and ‘ME’.

    Age → 16 <= age <= 55

Types of entities

– categorized based on their characteristics

1. Strong entity -Entity exists independently of other entity
   - Ex : Student , lecturer, department
2. Weak entity - Entity that existence depend on some entity type
   - Ex : Section of a dept. is depend on Department entity
3. Composite entity - known as a “bridge” entity. This is used to handle the many-to-many relationships
   - EX : students are enrolled to courses

Types of entities

4. Recursive entity - entity which is one in a relation can exist between occurrences of the same entity set
   - Ex: employee who is also a manager, and manager who is also an employee, details are saved in employee entity

Types of attributes

1. Simple attribute – atomic values which cannot be divided further
   - Ex : phone number
2. composite attribute – made of more than one simple attribute
   - Ex : student name → first_name and last_name
3. Derived attribute - values are derived by other attributes presents in the database and this is not exist in physically
   - Ex : **age** is derived by **DOB**

Types of attributes

4. Single value attribute – contains single value
   - Ex : NIC
5. multi valued attribute – contains more than one values
   - Ex: person can have more than one phone number, email address,erc..

Relationship and Relationship set

Relationship : Association among entities and it holds association information

- Ex : students enroll in courses

Relationship set : a set of relationships of similar type

Relationship connectivity

-Describes relationship classification

1. 1:1 relationship

   ex : A student has only one ID card

2. 1:N relationship

   ex : A student can be enrolled to one department, but department can have many students

3. M:N relationship

   ex : One student can be assigned to many courses, also a course can be followed by many students

Relationship Cardinality

-Expresses the specific number of entity occurrences (min and max) associated with one occurrence of the related entity (provide boundary conditions like 1:3, define the storage requirements/limitations)

- Ex : each team has at least 12 at most 22 players and player may play in only one team

Degree of Relationship

– number of entity types that participate in it

1. Unary/ recursive (connects 1 entity type with itself)
   - Both participants in the relationships are the same entity
   - Ex: employee who is also a manager, and manager who is also an employee, details are saved in employee entity
2. Binary (connects 2 entity types)
   - When two entities participate and it is the most common relationship degree
   - Ex : teacher teaches student
3. Ternary(connects 3 entity types)
   - When three entities participate in the relationship.
   - Ex : Books are purchased by Employees and Students

Activity 01 (non graded)

1. Describes following actors involved in large databases.
   - i. Database Administrators
   - ii. Database Designers
   - iii. End Users(answer should be included types of end users)
   - iv. System Analysts and Application Programmers (Software Engineers)

Further Reading

Read Chapter 01 and Chapter 02

- Elmasri and Navathe. **Fundamentals of Database Systems**. Addison-Wesley, 6thEdition, 2011.

Assignment 01

1. Briefly describes following architectures used in DBMSs.
   i. Centralized DBMS architecture
   ii. Basic Client/Server Architecture
   iii. Two-Tier Client/Server Architecture
   iv. Three-Tier and n-Tier Architectures for Web Application
2. List down Database Management Systems (DBMS) used in database development and importance of its work.

(Submit on or before 21st **of February, 2022 to the LMS submission link**)

Maximum number of pages : 03