COL 362 & COL 632

Organization and Introduction

3 Jan 2023

Overview

- **Instructor:** Srikanta Bedathur
- **Office hours:** Tuesdays 5pm – 6pm.
- **TAs:** Anjali, Rohan, Harsh, Nikhil, Neeraj, Sahil, Rakshita, Aditya, Tushar and Mohit.
- **Course website:** https://www.cse.iitd.ac.in/~srikanta/course/col362-632-2023/
- **Piazza:** https://piazza.com/iitd.ac.in/winter2023/col362632
  - **Accesscode (note this down, will not be shared again):** **7grc9e09sz3** (open until the early add/drop deadline)

What is this course about?

- We are living in a world overflowing with data
- It is by far the most important resource for success

Data is the new oil. It's a valuable resource that can be mined, refined, and used to fuel the economy and drive innovation.
Satya Nadella, CEO Microsoft, The Verge 2017

Data is the new electricity.
Sundar Pichai, CEO, Alphabet/Google, Economist 2015

Data is the new soil. It's the new foundation for business, for digital transformation, for innovation, for growth.
Bernard Marr

Data is the new currency. It's valuable, but it's also a liability. If it's stolen or mishandled, it can create a huge problem.
Mark Barrenechea

How do we store, manage, generate insights from data?

DataBase Management Systems

Course Contents

We learn about DBMS

- **What** is it? How is different from data structures?
- **How** do we **use** it? (SQL, App Development)
- What goes on **inside**, and how do we make it **better**? (Indexes, Query Planning/Optimization, Views)
- How do we **trust** the data stored in it? (Transactions, Concurrency, Recovery)
- Is there some **theory** behind it, or is it just a whole pile of hack? (Datalog, Algebra)

And more …

Textbook(s) and study material

- **Database System Concepts (7 ed.)**
  by Silberschatz, Korth and Sudarshan
  resource website: https://www.db-book.com

- **Database Systems: The Complete Book**
  by Garcia-Molina, Ullman, Widom.

- There are other books which are perhaps equally good (and better for some topics) – but we will not use them
  - If needed, I may post some excerpts on Piazza
- Course slides are adapted and extended from previous course offerings + textbook slides and some new material

Evaluation

	Individual weightage	Comp	Audit
Assignments (individual) x 3	10% each	30	≥ 15, all submissions
Project (teams upto 3)	25%	25	≥ 10
Minor1	10%	10	must
Minor2	10%	10	must
Major	25%	25	must

- **Specific instructions will be given for each assignment**
  - **No exceptions. if not followed, they will not be graded**
- **All assignments will be posted on moodle, and submissions will be through moodle**
- **Details about project will be posted soon.**

Assignments

- **Assignments will be posted on Moodle**
- Will involve heavy amount of work in short periods of time
  - Start early – do not take it lightly
  - Follow the submission guidelines **exactly**
  - Many evaluation steps are going to be automated
- **Absolutely no extensions to deadlines –** even if there are
  - Health issues, personal issues, Computer crashes, …
  - Errors in the assignment text
  - COVID, other natural disasters…

One pre-minor1,
One in between minor1-minor2
One after minor2

Academic Honesty

- Every submission goes through **automatic plagiarism check**
- A **random 10% will be manually screened and subjected to viva**
- *You are expected to solve all assignments on your own*
  - Do not **show or give your work**
  - Do not borrow / copy / "steal" **anyone's** work
    "**anyone**" can be, but not limited to, your classmate, senior, a web resource, chatgpt, textbook, etc.
  - Legitimate submissions are **thought, typed & tested by you alone**
- **Impersonating someone else is considered cheating**
- Discuss problems with each other, explore solution directions
- **If caught**
  - **Assignments:** You will receive -ve of the weightage for the assignment
  - **Exam:** -ve of the total exam marks
  - **Multiple offence** will result in grade reduction(s) – i.e., for 3 offences grade reduces twice

Contact

- All contact only through **Piazza**
  - NO to emails, no phones/messages, personal visits, telepathy, …
  - It is your responsibility to enroll on Piazza
- Give reasonable time for response
  - Especially as the deadline approaches
  - Don’t expect TAs and instructor to be always available
- We will **stop** responding to assignment questions 10 hours before the deadline

Resources Required

- **A computer / laptop:** at least 1 GHz CPU, 2GB RAM, 10GB Diskspace
- **PostgreSQL 8.3.23 (binaries as well as compiled source code)**
  - For using:
    - **CPU architectures**: x86, x86_64, IA64, PowerPC, PowerPC 64, S/390, S/390x, Sparc, Sparc 64, ARM, MIPS, MIPSEL, and PA-RISC.
    - **Operating systems**: Linux (all recent distributions), Windows (XP and later), FreeBSD, OpenBSD, NetBSD, macOS, AIX, HP/UX, and Solaris.
  - For building from the source – you will need this for assignments (check: https://www.postgresql.org/docs/current/install-requirements.html)
    - GNU make version 3.81 or newer is required;
    - ISO/ANSI C compiler (at least C99-compliant).
    - tar is required to unpack the source distribution, in addition to either gzip or bzip2.
- Internet connectivity, IITD network for moodle and other resources that may be shared

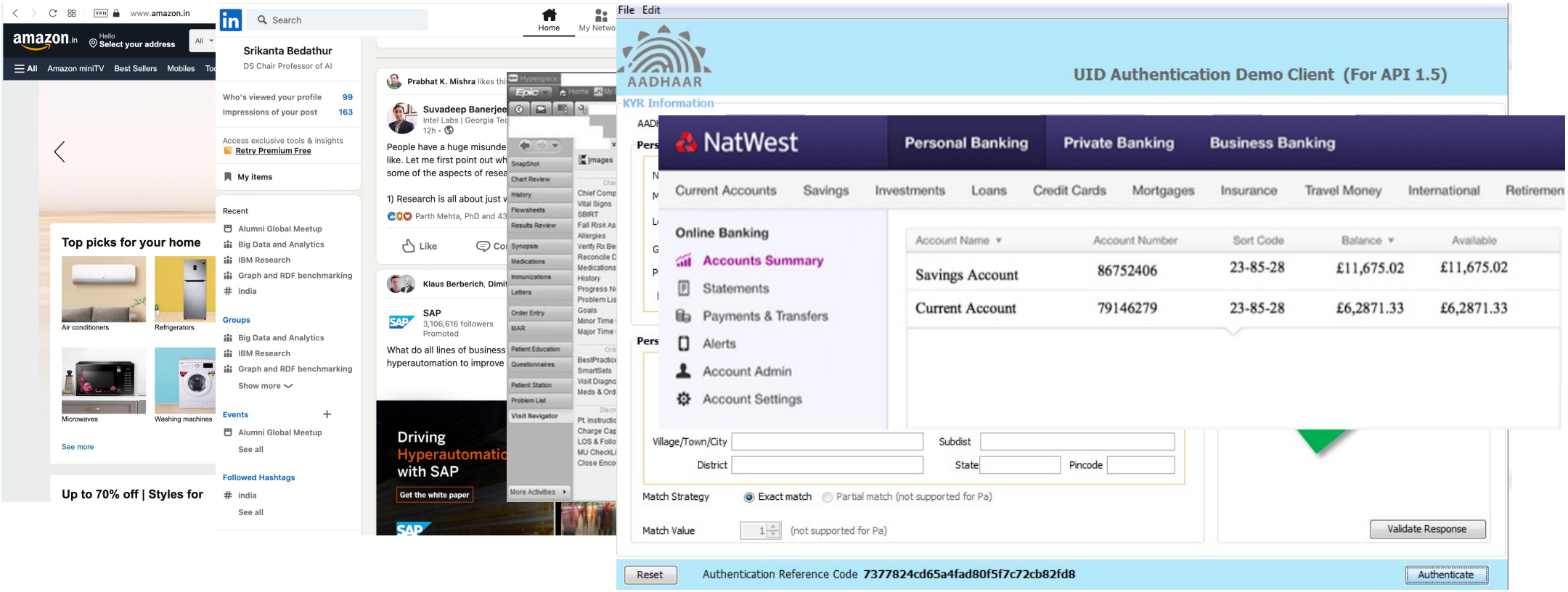
Intro to DBMS

Slides **partially** based on the wonderful textbook slides.

Semester II, 2022-23

DBMS Rules the World (really..)

- Behind nearly any application you have a DBMS



Do we really need Database Systems?

- One could use files and write applications
- But…
  - **Data redundancy and inconsistency:** potentially in many files, formats, leading to wastage of space and error-prone
  - **Tough to access:** for each new need, write a separate program
  - **Siloing of data:** each task of the applications may require separate formats, and files
  - **Integrity:** Need to provide **guarantees about data –** a student's age can not be 100, account balance > 0,
  - **Consistency:** failures can lead to data being wrong – imagine losing a friend because your "like" was not written to the file, losing money when your YONO crashed, patient record because there was a power failure, … Or even "harmless" updates
  - **Many users updating concurrently**

DBMS Enables

- Large amounts of data, persisting “forever”
  - Think “database”, think “disk”
- Physical and logical independence
  - Declarative languages for data manipulation
- Operations on the data
  - Creating a database
  - Insert, delete, modify, retrieve data
- Guarantees about the data

Data can be (in principle) anything, but we will focus mostly on **structured data**

- Student records, Customer records,

Data – Modeling, Storing

- Analyze what data is required / available for an application (learn by practice)
  Every patient has multiple diseases, there are many patients for each disease, each doctor is qualified to deal with a set of diseases
- **Data Models:** A collection of tools for describing
  - Data
  - Data relationships – how are students related to a course?
  - Data semantics – number 10 may not mean the same
  - Data constraints – e.g., an employee can not be his own manager, salary > 0, etc
- **Relational model**
- **Entity-Relationship** data model (mainly for database design)
- Object-based data models (Object-oriented and Object-relational)
- Semi-structured data model (XML)
- Other older models:
  - Network model
  - Hierarchical model

Data Modeling – Plan vs. Storage

- **Logical Schema** – the overall logical structure (plan) of the database
  - **Example**: The database consists of information about a set of customers and accounts in a bank and the relationship between them
  - Analogous to type information of a variable in a program
- **Physical schema** – the overall physical structure of the database
  - How exactly it is stored – independent of the logical plan
- **Instance** – the actual content of the database at a particular point in time
  - Analogous to the value of a variable

Applications only need Logical Schema

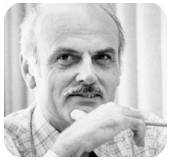
- **Physical Data Independence** – the ability to modify the physical schema without changing the logical schema
- Applications depend only on the logical schema
- In general, the interfaces between the various levels and components should be well defined so that changes in some parts do not seriously influence others.

- Database Systems ensure well-defined API for accessing and manipulating data

Relational Model

- All the data is stored in various tables.
- Example of tabular data in the relational model

Ted Codd's paper in 1970
Turing Award 1981



Columns

Database

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

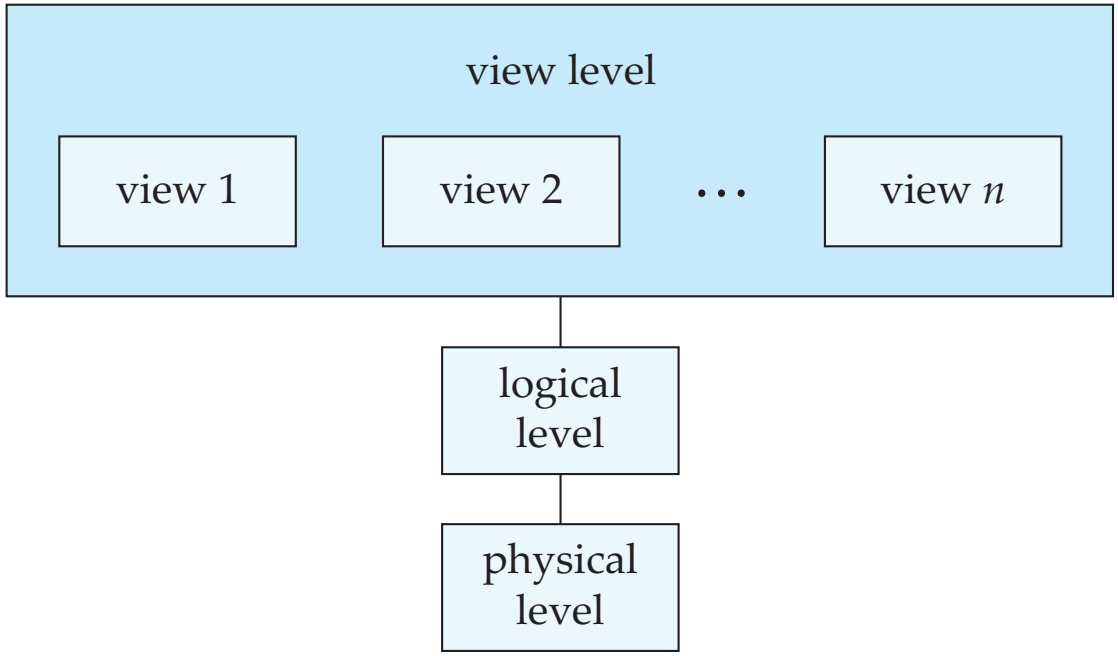
Rows

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

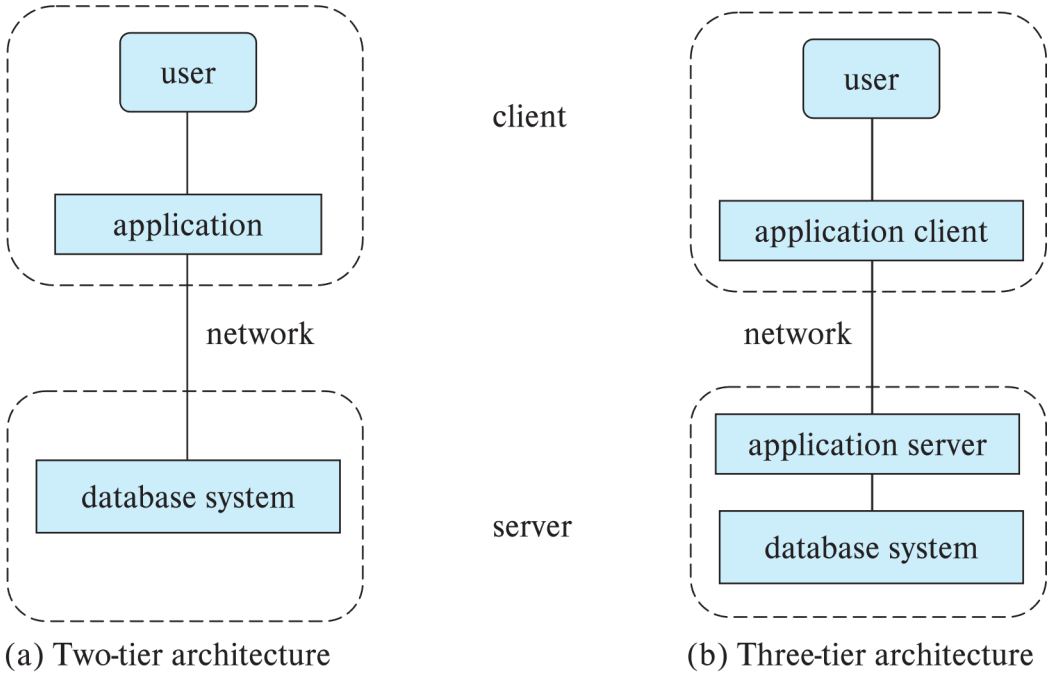
Abstractions … abstractions

- An abstract architecture for a database system



Database Applications

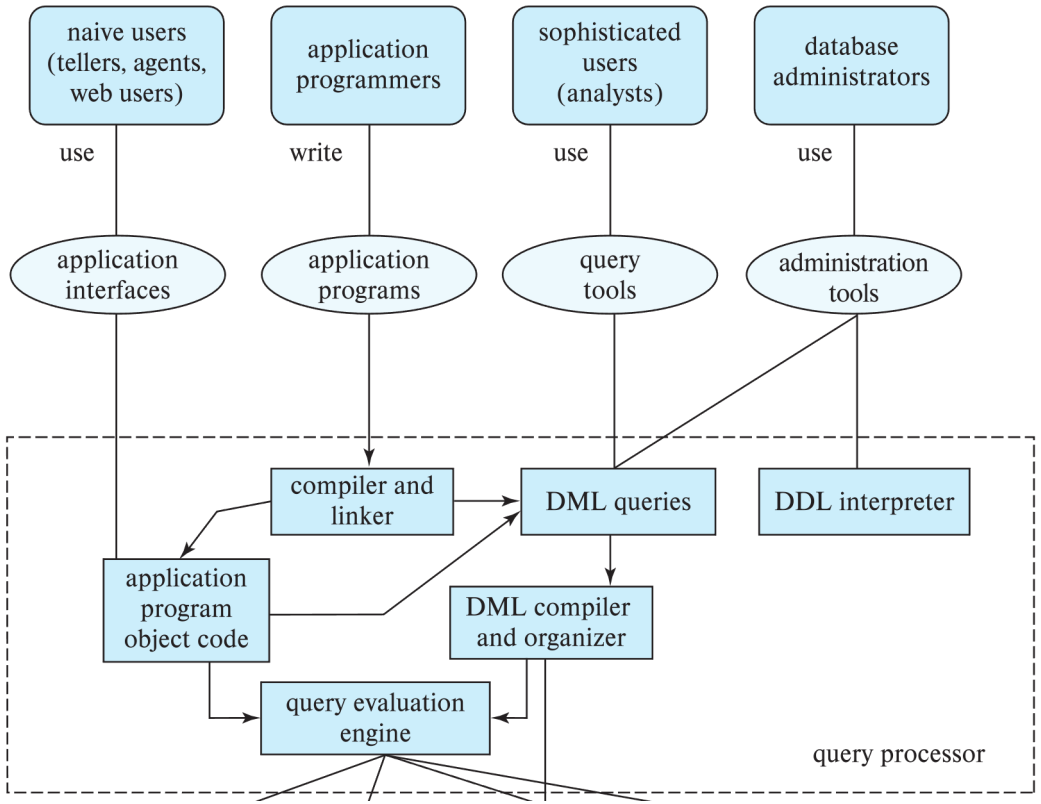
- **Two-tier architecture** -- the application resides at the client machine, where it invokes database system functionality at the server machine
- **Three-tier architecture** -- the client machine acts as a front end and does not contain any direct database calls.
  - The client end communicates with an application server, usually through a forms interface.
  - The application server in turn communicates with a database system to access data.



(a) Two-tier architecture

(b) Three-tier architecture

Database Users



Database Administrator

Schema definition

Storage structure and access-method definition

Schema and physical-organization modification

Granting of authorization for data access

Routine maintenance

Periodically backing up the database

Ensuring that enough free disk space is available for normal operations, and upgrading disk space as required

Monitoring jobs running on the database

COL 362 & COL 632

Database Design with E-R Models

4 Jan 2023

"One whose plans are well-laid has nothing to fear." - Chanakya

An important change in policy - Attendance

- **Minimum 75% attendance is required.**
  - We will count from **next week** (after drop deadline).

- Audit-pass also requires a minimum attendance of 75%
- Attendance will be as recorded in Timble

- **Below required attendance will result in a grade reduction**

Data Models

- A set of rules and standards for organizing data
- There are several data models
  - **E-R Models (diagrammatic, useful for humans)**
  - **Relational Models (formal, based on logic, most common)**
  - Object-oriented and Object-relational models
  - Semi-structured models such as XML
  - Graph models – RDF, Property-graphs, RDF*
  - Key-value models
  - …
- Data models provide an “API” to store / query / manipulate data

Database Design

- **Initial phase** -- characterize fully the data needs of the prospective database users
- **Second phase**
  - Choosing a data model
  - Applying the concepts of the chosen data model
  - Translating these requirements into a conceptual schema of the database.
  - Describe the kinds of operations (or transactions) that will be performed on the data.
- **Final phase** - moving to an actual database
  - Logical schema design
  - Physical storage design

Educational Institution

Students, Courses, Instructors, Rooms, Slots, Exams, teaching assistants, departments, administration, finance dept. (fees), degrees, co-curricular act. tech support, employees,

Design Pitfalls

- **Redundancy**
  - Bad design may repeat information
  - Redundant representation of information may lead to data inconsistency among the various copies of information

- **Incompleteness**
  - Bad design may make certain aspects of the application domain difficult or impossible to model.
  - *Not storing the topic of a special course offering will make it impossible for students to take the same course number again*

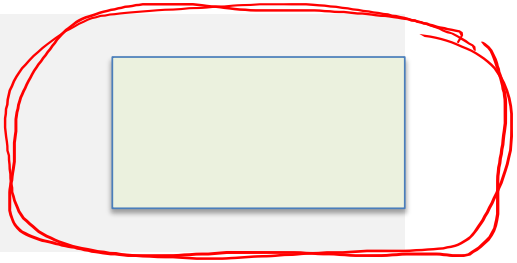
E-R Modeling

Semester II, 2022-23

Entity-Relationship Models

- Models an enterprise as a collection of entities and relationships
- **Entity:**
  a “thing” or “object” in the enterprise that is distinguishable from other objects
  - Described by a set of attributes
  - *Actors, Books, Students, Patients, Diseases …*
- **Relationship:**
  an association among several entities
  - Represented diagrammatically by an entity-relationship diagram

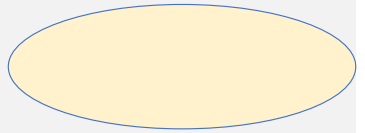
Entity set



- An **entity** is an object that exists and is distinguishable from other objects.
  - Example: specific person, company, event, plant
- An **entity set** is a set of entities of the same type that share the same properties.
  - Example: set of all persons, companies, trees, holidays
- An entity is represented by a set of attributes; i.e., descriptive properties possessed by all members of an entity set.
  - Example:

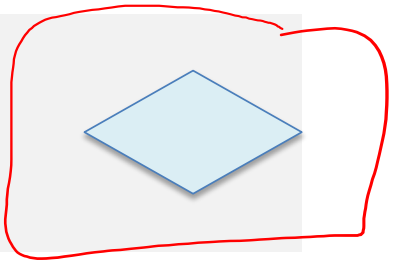
    instructor = (*ID, name, salary*)
    course= (*course_id, title, credits*)

Attributes



- Attributes are properties of entities
- They are *atomic - i.e., strings, integers, reals, …*
- Denoted by “connecting” them to corresponding entity
- Not shared by different entities

Relationships



- A **relationship** is an association among several entities

  Example:

  44553 (Peltier) *advisor* 22222 (Einstein)

  student entity relationship set *instructor* entity

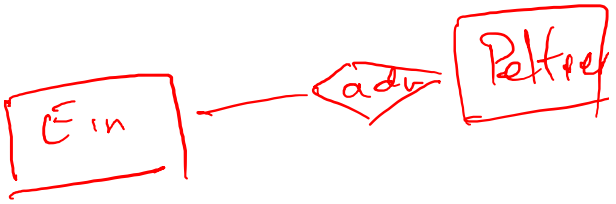
- A **relationship set** is a mathematical relation among $n \geq 2$ entities, each taken from entity sets

$$\{(e_1, e_2, \ldots e_n) \mid e_1 \in E_1, e_2 \in E_2, \ldots, e_n \in E_n\}$$

  where $(e_1, e_2, \ldots, e_n)$ is a relationship

  - Example:

    $(44553,22222) \in advisor$



Some Example Entities and Attr

Person / Actor.

Robert Redford

Redford in 2012

Born	Charles Robert Redford Jr. August 18, 1936 (age 86) Santa Monica, California, U.S.
Alma mater	University of Colorado Boulder Pratt Institute American Academy of Dramatic Arts
Occupations	Actor · director · producer · activist
Years active	1958–present
Spouses	Lola Van Wagenen (m. 1958; div. 1985) Sibylle Szaggars (m. 2009)
Children	4, including James and Amy
Awards	Full list
Website	sundance.org

full name

DOB

? location of place of Birth

Klara and the Sun

	First edition
Author	Kazuo Ishiguro
Country	United Kingdom
Language	English
Genre	Science fiction · dystopian
Publisher	Faber and Faber
Publication date	2 March 2021
Media type	Print
Pages	307
ISBN	978-0-571-36487-9 (Hardback)
OCLC	1158700107
Dewey Decimal	823/.914
LC Class	PR6059.S5 K57 2021
Website	www.klaraandthesun.com

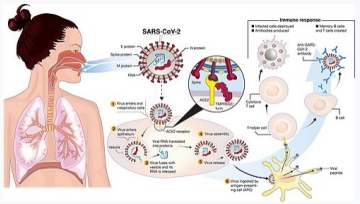
Actor

Date of Birth

Local

Coronavirus disease 2019 (COVID-19)

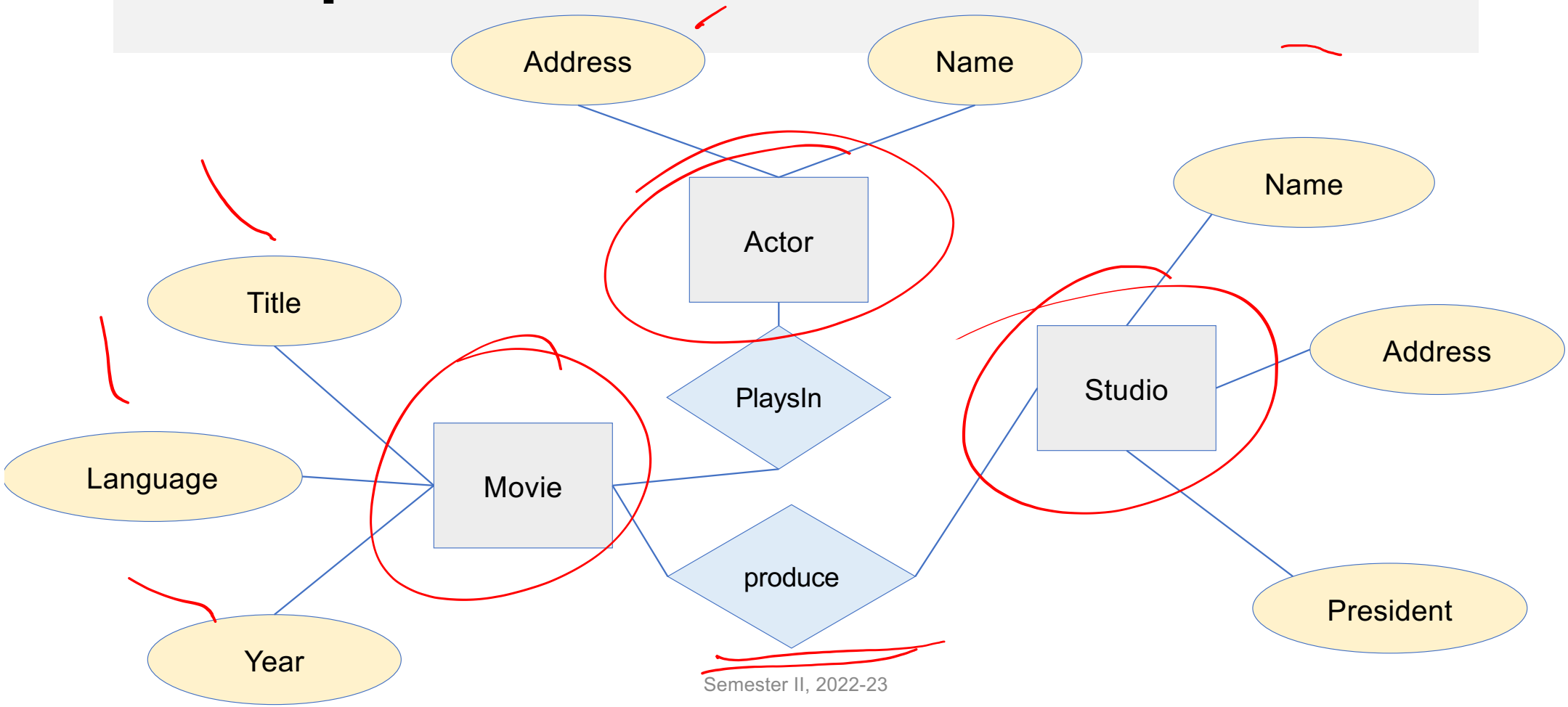
Other names COVID, (the) coronavirus



Transmission and life-cycle of SARS-CoV-2 causing COVID-19

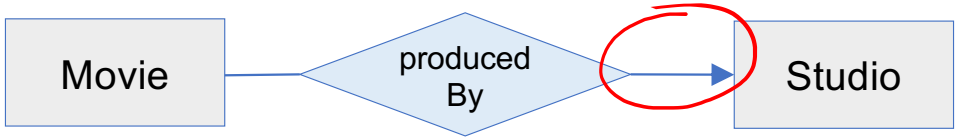
Pronunciation	/kəˈroʊnəvaɪrəs/ /koʊvɪdnaɪˈtiːn, kɒvɪd-/[1]
Specialty	Infectious disease
Symptoms	Fever, cough, fatigue, shortness of breath, vomiting, loss of taste or smell; some cases asymptomatic[2][3]
Complications	Pneumonia, viral sepsis, acute respiratory distress syndrome, kidney failure, cytokine release syndrome, respiratory failure, pulmonary fibrosis, paediatric multisystem inflammatory syndrome, long COVID
Usual onset	2–14 days (typically 5) from infection
Duration	5 days to chronic
Causes	Severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2)
Diagnostic method	rRT-PCR testing, CT scan, Rapid antigen test
Prevention	Vaccination,[4] face coverings, quarantine, physical/social distancing, ventilation, hand washing[5]
Treatment	Symptomatic and supportive
Frequency	660,399,206[6] confirmed cases
Deaths	6,690,155[6]

Example E-R Model

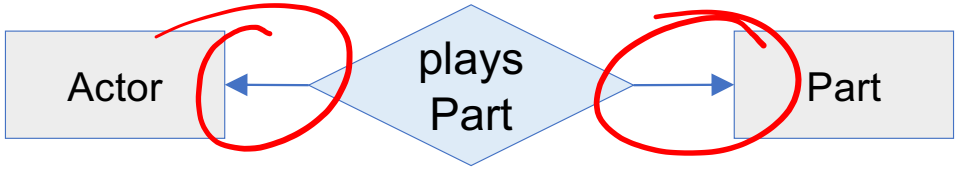


Types of relationships (1/3)

Movie produced by ***at most*** one studio, but a studio produces many movies



An actor plays **at most** one role, a role is played **by at most** one actor



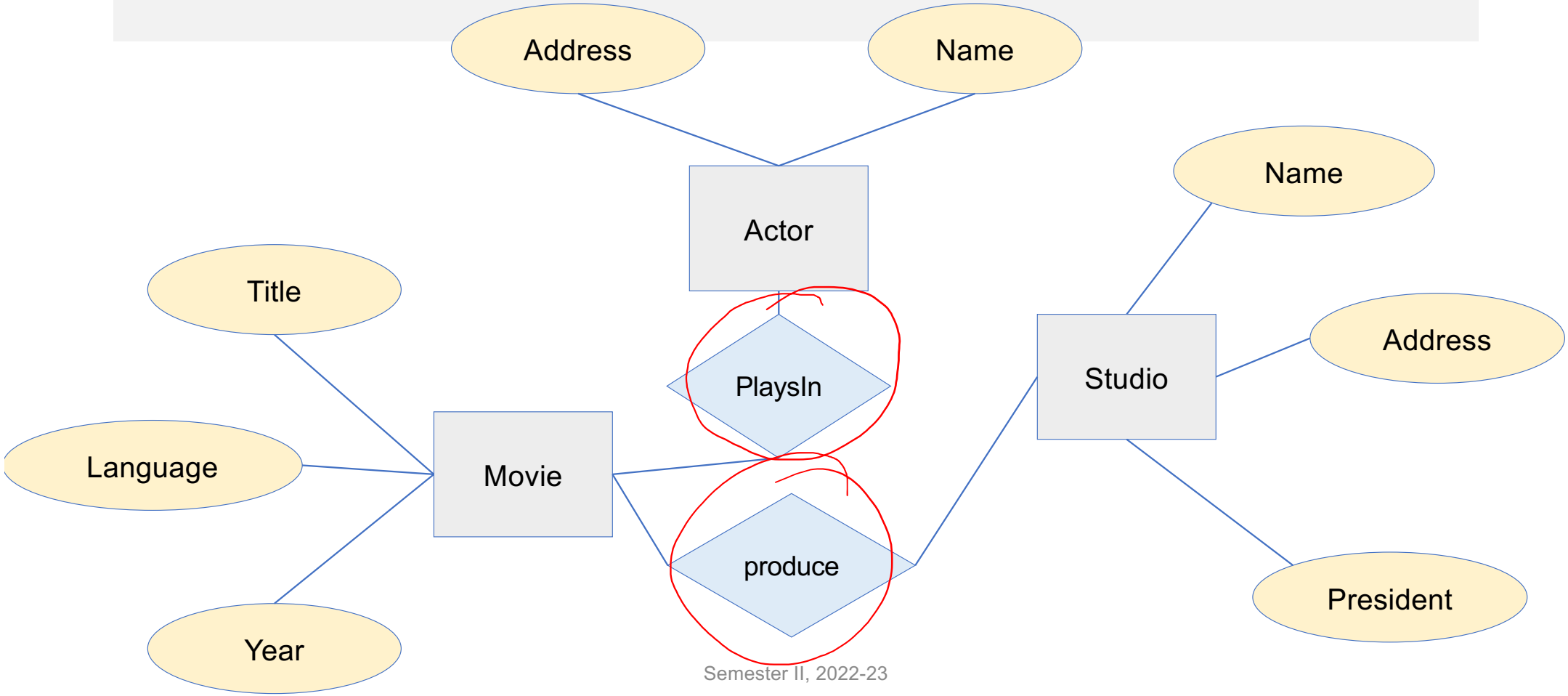
A movie has **many** actors, an actor plays in **many** movies

COL 362 & COL 632

ER + Intro. Relational

6 Jan 2023

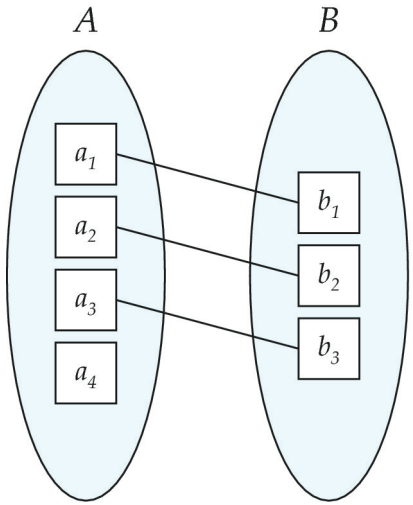
Example E-R Model



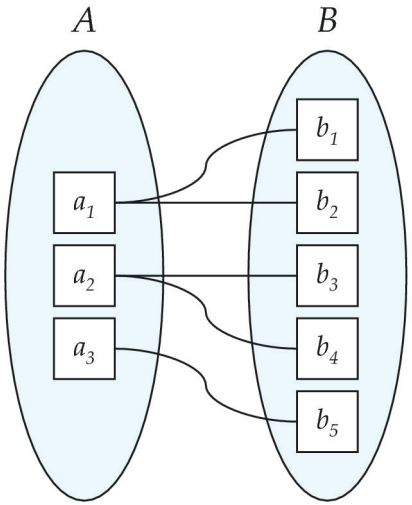
Types of relationships

- **one-one**
  - an entity of one entity set is related to at most one entity of another entity set and vice-versa
  - *A studio has **at most** president and one can be a president of only one studio, an IIT has **at most** one director and one can be a director of only one IIT (at a time)*
  - At most one != exactly one
- **many-one**
  - many entities of one entity set are related to at most one entity of another entity set
  - *A student can be enrolled in at most one program, but a program will have many students*
- **many-many**
  - Students – courses, movies – actors, …

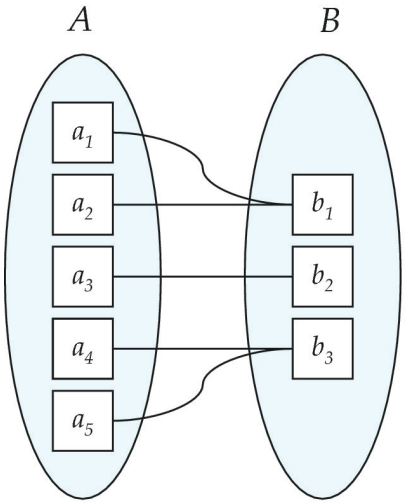
Types of relationships (contd.)



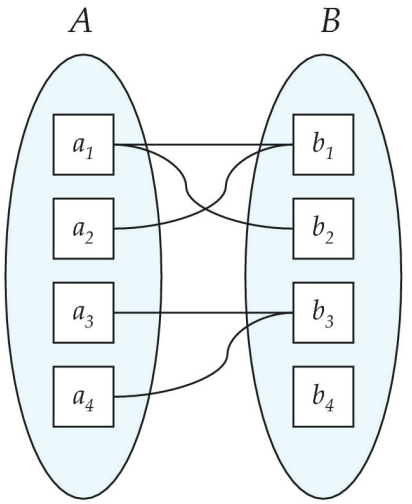
One to one



One to many



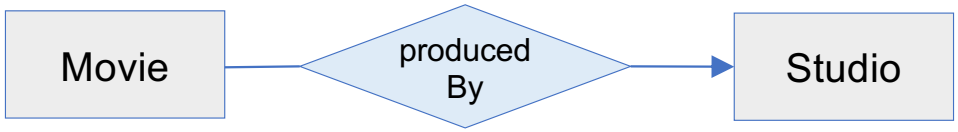
Many to one



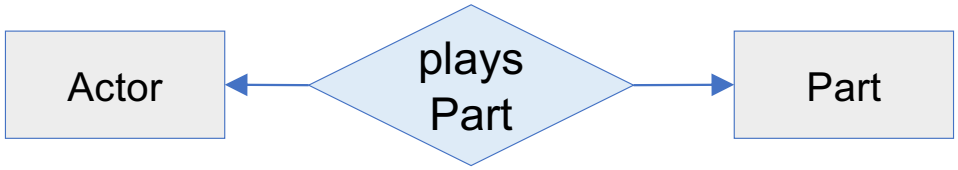
Many to many

Types of relationships (contd.)

Movie produced by ***at most*** one studio, but a studio produces many movies



An actor plays **at most** one role, a role is played **by at most** one actor



A movie has **many** actors, an actor plays in **many** movies

Imposing **constraints** on the relationships between entitysets

Determining the uniqueness of an entity

- By definition, individual entities are distinct, but a database needs to model this
- **The values of the attribute values of an entity must be such that they can uniquely identify the entity.**
- **Keys**
  - Uniquely identify an entity
  - Can the pair of attributes, (Name, Year of birth), identifies an actor uniquely?
- Now we can enforce single-value constraints
  - Unique values in a given context
  - Place of birth has to be unique
  - What if we do not know either of these two? Null values
- ***Every entity set*** should have a key
  - There could be more than one key - e.g., Aadhaar id and entry no. of a student (assuming all Indian nationals)
  - What if we do not have any such attribute set?

Primary key: minimal attribute-set that uniquely identifies an entity

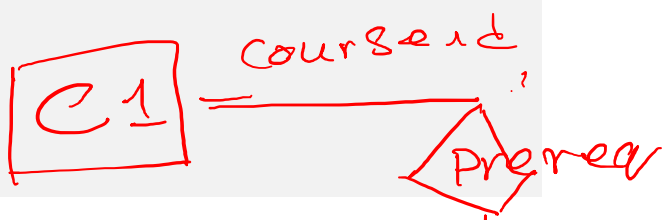
(Primary) Key for Relationship Sets

- To distinguish among the various relationships of a relationship set we use the individual (primary) keys of the entities in the relationship set.
  - Let ***R*** be a relationship set involving **entity sets E1, E2, .. En**
  - The primary key for **R** consists of **the union of the primary keys** of entity sets **E1, E2, ..En**
  - If the relationship set ***R*** has **attributes a1, a2, .., am** associated with it, then the primary key of ***R*** **also includes the attributes a1, a2, .., am**
- Example: relationship set “advisor”.
  - The primary key consists of *instructor.ID* and *student.ID*
- The choice of the primary key for a relationship set depends on the mapping cardinality of the relationship set.

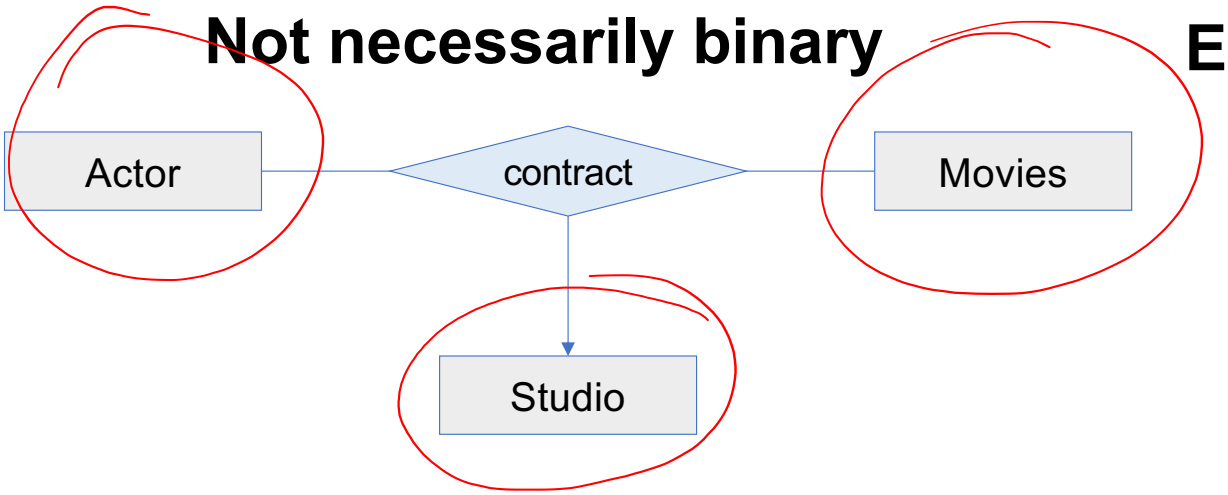
Choice of Primary key for Binary Relationship

- **Many-to-Many relationships**.
  - The preceding union of the primary keys is a minimal superkey and is chosen as the primary key.
- **One-to-Many relationships** .
  - The primary key of the “Many” side is a minimal superkey and is used as the primary key.
- **Many-to-one relationships.**
  - The primary key of the “Many” side is a minimal superkey and is used as the primary key.
- **One-to-one relationships.**
  - The primary key of either one of the participating entity sets forms a minimal superkey, and either one can be chosen as the primary key.

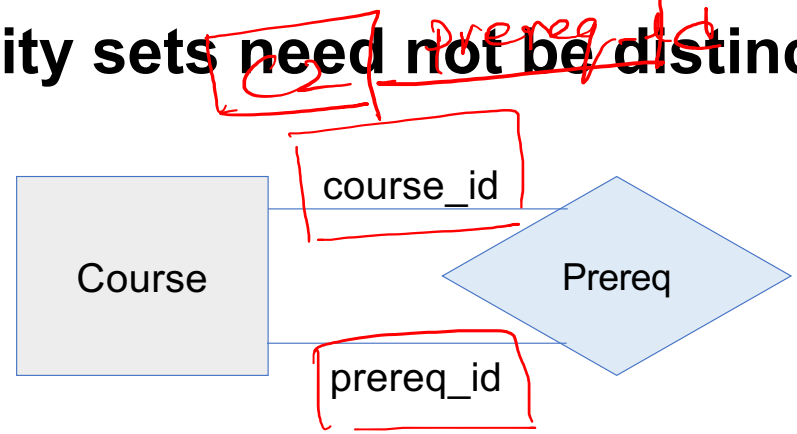
Multi-way relations and Role-based relations



Not necessarily binary

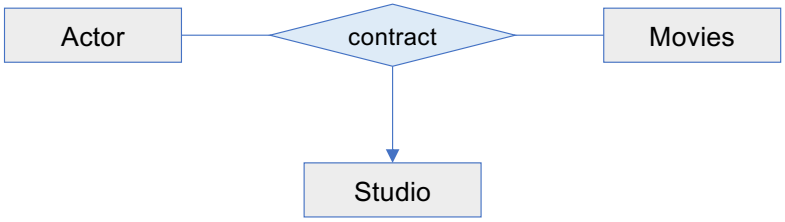


Entity sets need not be distinct

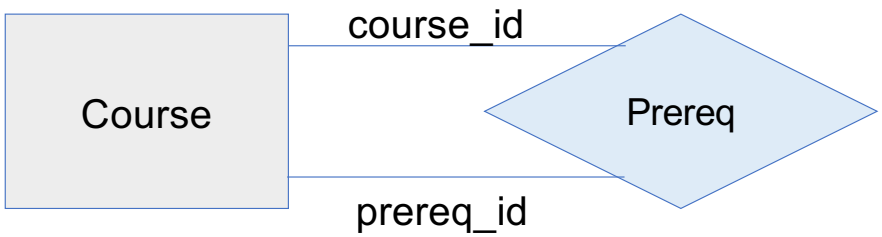


Types of Relationships (2/3)

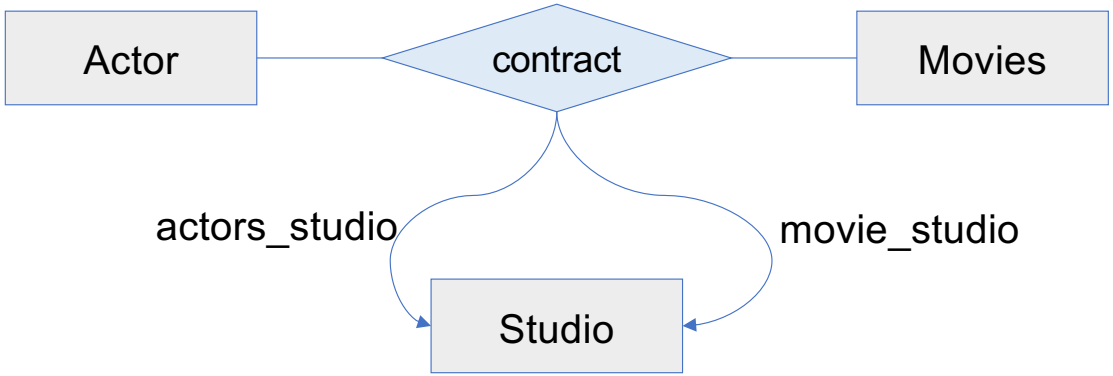
Not necessarily binary



Entity sets need not be distinct



Combining both



Self-study Topics

- Subclass mechanism in E-R
  - E.g., all instructors are also persons, so every attribute/relationship that Person entityset has will hold for Instructors too.
- Weak Entitysets
  - Some entities may not be able to have unique identity without redundant attribute stored from another entity
- Complex cardinality and domain constraints on relationships
- Constraints in presence of multi-way relationships
- Converting multi-way relationships to binary relationships

Self study from Chapter 6 of the DB-Book

Relational Model

Semester II, 2022-23

Everything is a “Relation”

- A relation is both a mathematical concept and just a table of values
- The relational model models “everything” as relations
- Schema of the relation:
  - **Actor-Movies** (Name varchar(20), Movie varchar(50), Character varchar(20))

Actor-Movies

Name	Movie	Character
Priyanka Chopra	Baywatch	Victoria Leeds
Tom Cruise	MI-I	Ethan Hunt
Anthony Hopkins	Thor: Ragnarok	Odin

- Relation is a **Set** not a bag or a sequence
- Attribute set is a **set** – order invariant.

E-R to Relational

- ER diagrams are easy to comprehend and closer to how we think
- Relational model is powerful because it is simple – only one kind of object
  - Any operation on the relation, results in yet another relation
- So, let’s convert our ER diagrams to relational!