FOUNDATIONS OF SQL

BY VISHWA PATEL

SHIFTKEY LABS

ABOUT ME

- Vishwa Patel
- Graduated from Dal in Fall 2024!
- Full time employee at Bell as Technical Network Architect
- Feel free to connect with me on LinkedIn with any questions!

NOTES

- You can take it the way you like or refer to my slides
- To make ER diagram use link - https://drawsql.app/diagrams
- ER Model - https://erdplus.com/

COURSE OUTLINE

6th January

Week 1– Introduction to ER Diagrams

- Terminologies
- Designing an ER Diagram
- Converting ER diagrams to Database

13th January

Week 2 – DDL commands and SQL Basics

- Constraints
- Creating tables
- Inserting data into tables

20th January

Week 3 – Joins and Basic Functions

- Basic queries
- Joins
- Aggregate functions

27th January

Week 4 – Intermediate SQL concepts

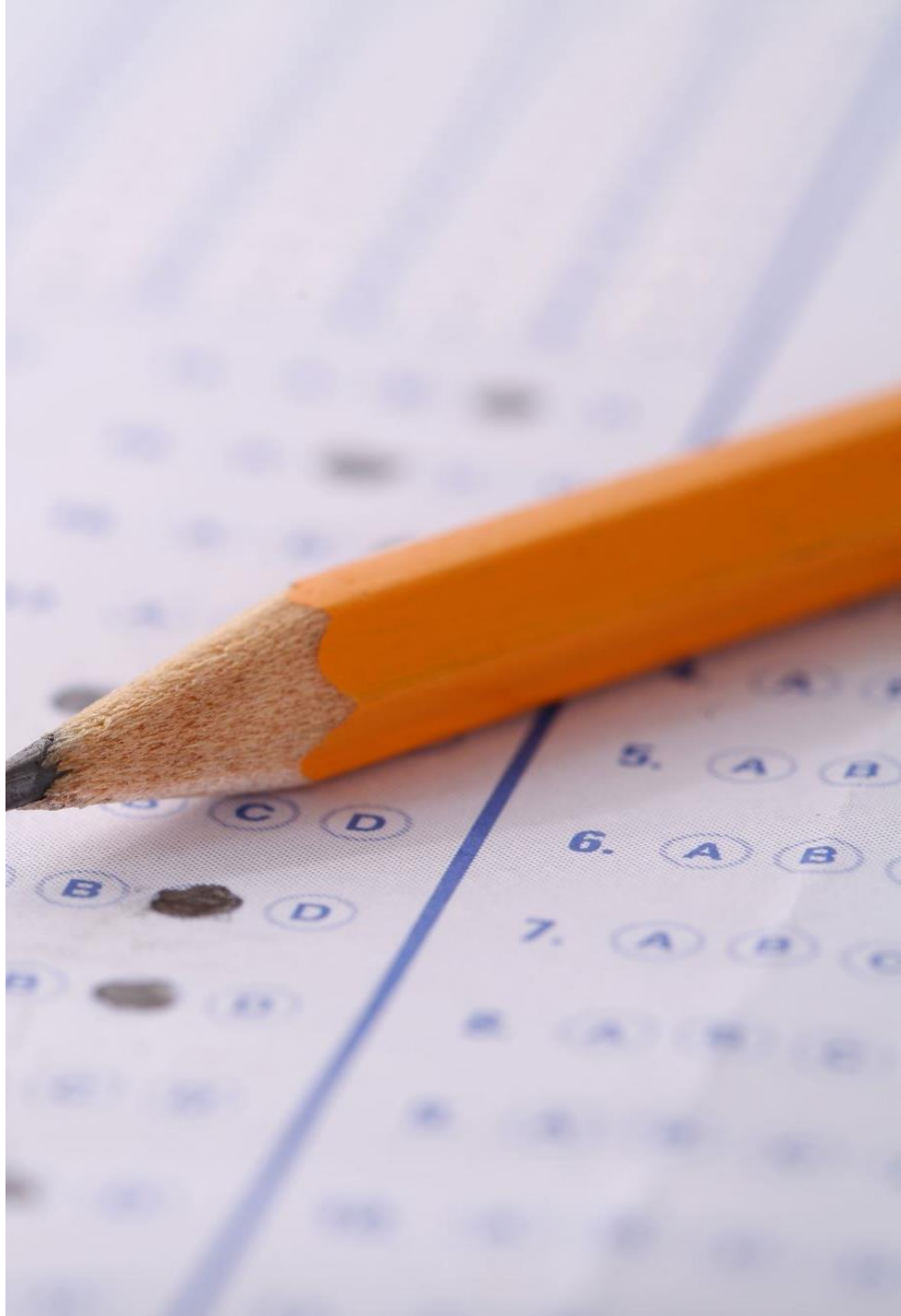
- Updating tables
- Subqueries
- Tips and Tricks

ASSESSMENT

Project submission required (further details to be provided soon!)

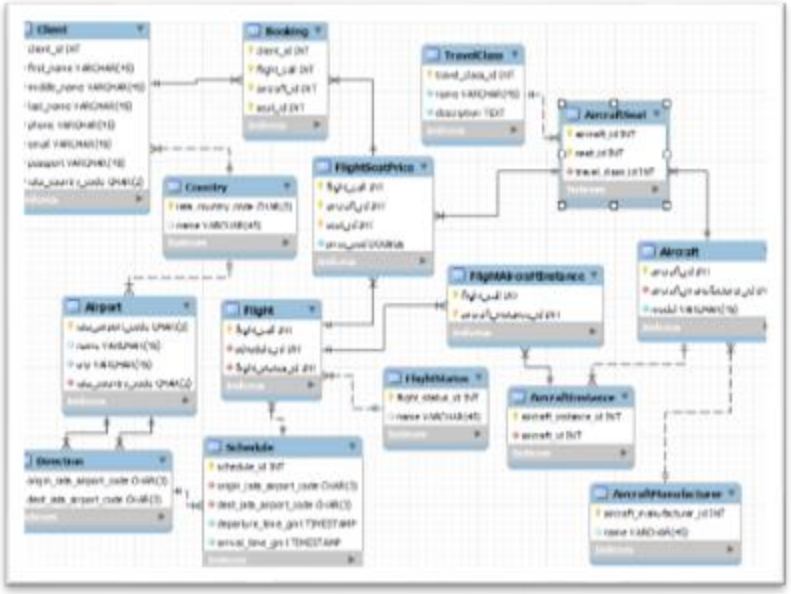
Criteria

- Need >85% in the project
- 75 % minimum attendance required (3/4 in-person classes)
- Deadline: 5th Feb 2025



WHAT IS DATABASE (DB)?

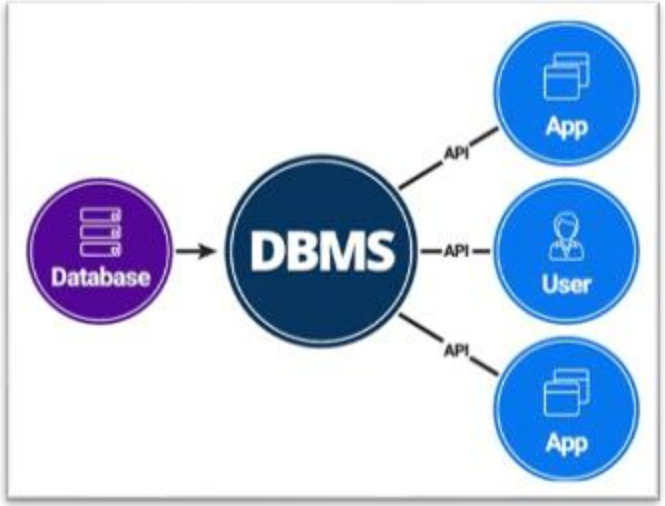
- Any collection of related information
- Computers make it easier to handle large amounts of data



This Photo by Unknown Author is licensed under CC BY-SA

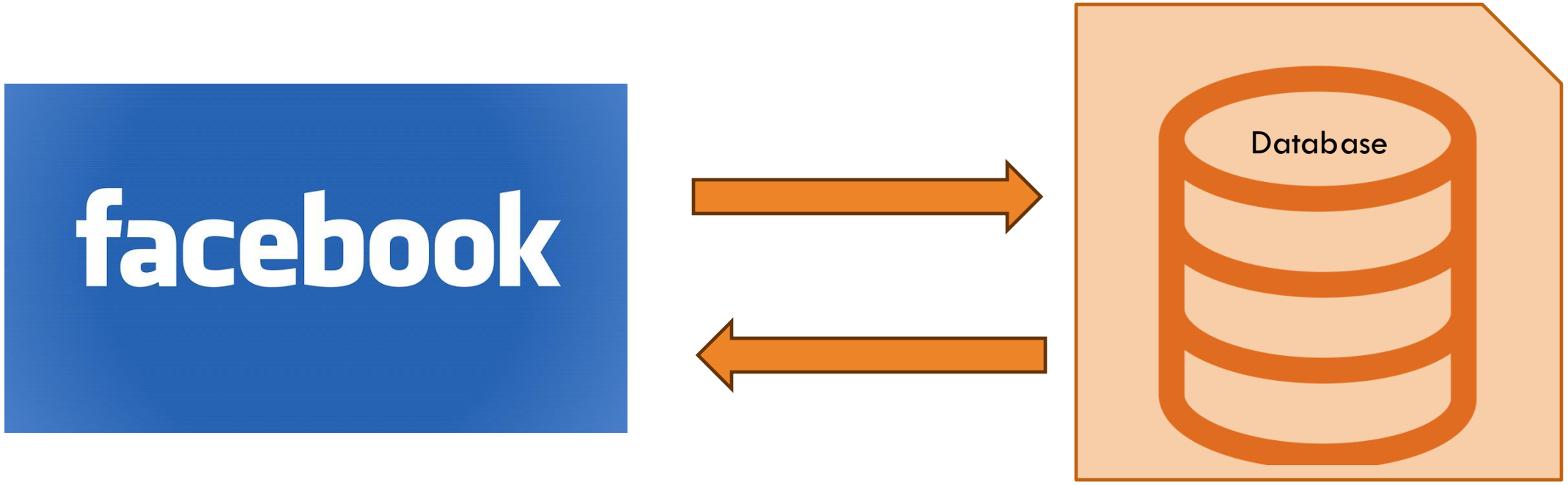
WHAT IS DBMS?

- Database Management System
- Special software program that helps users create and maintain database
  - Manages security of the data
  - Backups
  - Importing/Exporting Data
  - Easy to manage large amounts of data
  - Programming language can be used for interacting
- Performs CRUD operations – Create, Read/Retrieve, Update, Delete



This Photo by Unknown Author is licensed under CC BY-SA-NC

DATABASE INTERACTION



Database Management System (DBMS)

TYPES OF DATABASES

Relational Database	Non-Relational Database
SQL Database	No SQL/Not only SQL
Organizes data into one or more tables	Any database with information, not a table
Tables have rows and columns, more like excel sheet	Documents like JSON, XML file, etc.
Unique key for each row	Key – value storage

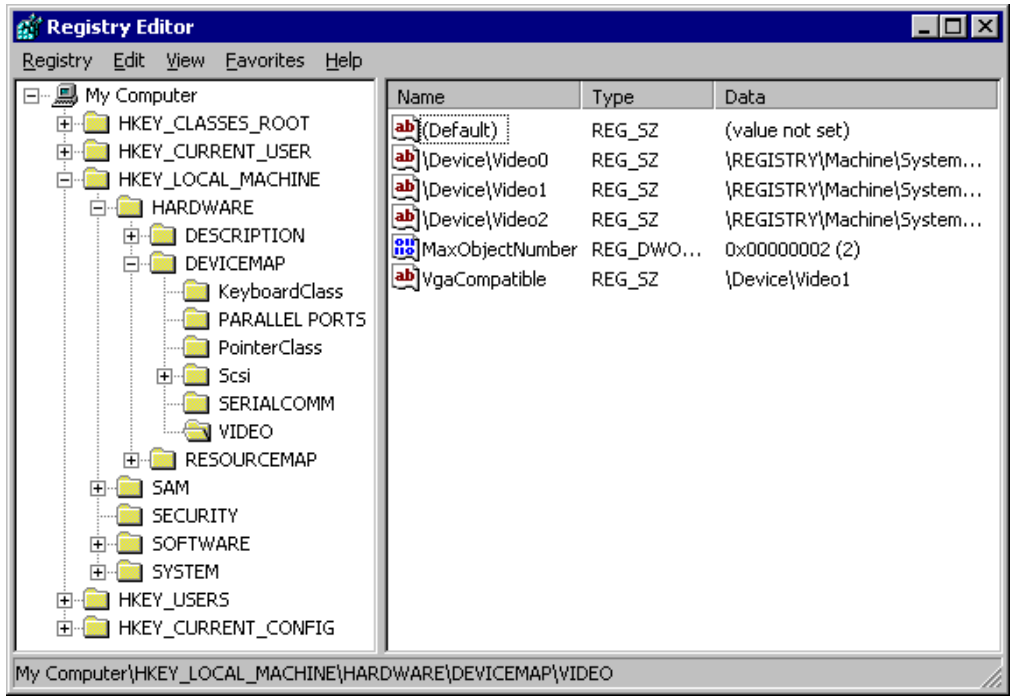
WHAT IS RDBMS?

- Relational Database Management System
- Create and maintain relational databases
- Uses SQL – standardized language for interacting with RDBMS
- SQL used for CRUD operations and other tasks like administrative, backups, etc.
- Used to define tables and structure

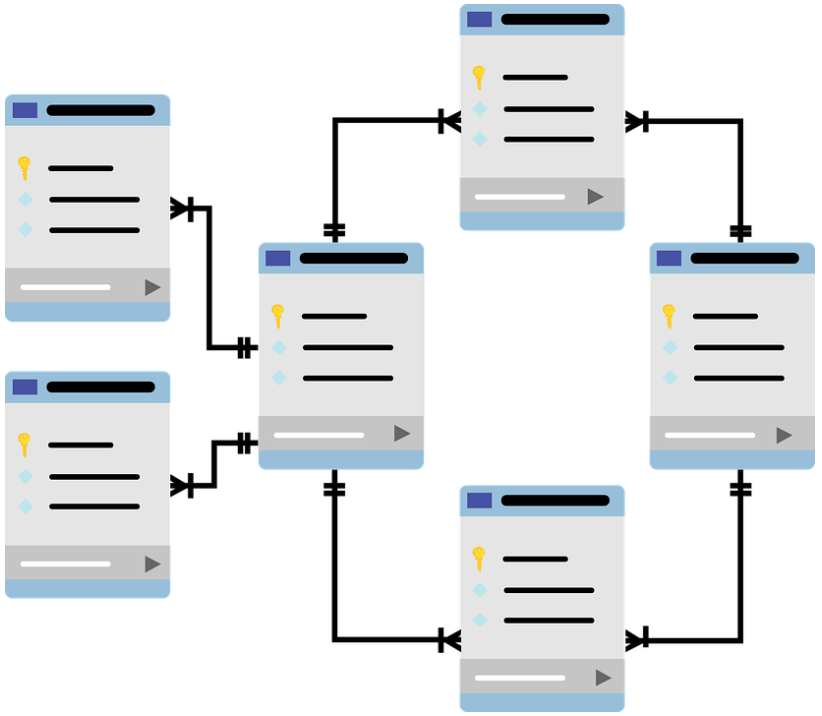
DBMS VS RDBMS

DBMS	RDBMS
Stores information as a file	Stores information in tables
Does not link data	Links data using keys, indexes, etc. to create relationships between tables.
Access one data at a time	Interact with multiple data elements at a time
Data redundancy is common	No Data redundancy
Deals with small amount of data	Can efficiently deal with large amounts of data
Example: Windows Registry	Examples: MySQL, PostgreSQL, Oracle, etc.

DBMS VS RDBMS



https://learn.microsoft.com/en-us/windows/win32/sysinfo/structure-of-the-registry



https://cdn.prod.website-files.com/644bb0d49c07b5dc9232d6f0/64aea5086620b6494985c654_Untitled%20(4).png

WHAT IS NRDBMS?

- Non-Relational Database Management System
- Helps create and maintain non-relational databases
- MongoDB, firebase, etc.
- They have their own language defined to communicate with the data

RDBMS VS NRDBMS

Relational database storage

customer_id	name	email	dob	mobile
1	John Drake	john.drake@gmail.com	24/11/1982	7843241098
2	Mary Chile	mary.chile@outlook.com	05/06/1981	8903424531
3	Mac Adams	mac_1979@gmail.com	23/04/1979	0920421454
4	Jill Smith	jellyjill@gmail.com	14/02/1987	8795092014

Key value database storage

Key	Value
customer_1	{ "name": "John Drake", "email": "john.drake@gmail.com", "dob": "24/11/1982", "mobile": 7843241098 }
customer_2	{ "name": "Mary Chile", "email": "mary.chile@outlook.com", "dob": "05/06/1981", "mobile": 8903424531 }
customer_3	{ "name": "Mac Adams", "email": "mac_1979@gmail.com", "dob": "23/04/1979", "mobile": 0920421454 }
customer_4	{ "name": "Jill Smith", "email": "jellyjill@gmail.com", "dob": "14/02/1987", "mobile": 8795092014 }

https://www.mongodb.com/resources/basics/databases/key-value-database#:~:text=key%2Dbased%20queries-,MongoDB%20as%20a%20key%20value%20store,for%20%E2%80%9Ckey%20value%E2%80%9D%20data.

ANY QUESTIONS?

INTRODUCTION TO SQL

WHAT IS SQL?

- Developed in 1970s by IBM
- Structured Query Language
- Interact with Relational Database Management System (RDBMS)
- MySQL is widely used as RDBMS
- Language to communicate with databases

WHY SQL?

- Faster processing speed
- Reliability and efficient
- Originally designed for non-programmers, makes it easy to learn
- Widely adopted
- Easily scalable

WHERE IS SQL USED?

- Widely used in all sectors (Healthcare, Finance, Social Media)
- Top companies like Netflix, Instagram, etc. use it for data analysis, database upgrades and maintenance of data.
- NASA uses SQL databases to manage space exploration mission's data
- Used in careers like data analysis, engineering and science.

SUMMARY

- Databases are nothing but collection of related information.
- DBMS to create, manage databases and perform CRUD operations.
- Databases are of 2 types, relational and non-relational.
- Relational managed by RDBMS like MySQL, etc.
- RDBMS stores data using table's structure whereas NRDBMS stores data is different formats.
- SQL is used in data analysis, maintaining the databases.

BUILD ER MODEL

LEARNING TERMINOLOGIES FOR STRUCTURING RELATIONAL DATABASE

WHAT IS TABLE?

- It is a structured collection of data
- Organized in rows and columns
- Each row has a unique ID to identify itself

customer_id	first_name	last_name	phone	email	street
1	Debra	Burks	NULL	debra.burks@yahoo.com	9273 Thorne Ave.
2	Kasha	Todd	NULL	kasha.todd@yahoo.com	910 Vine Street
3	Tameka	Fisher	NULL	tameka.fisher@aol.com	769C Honey Creek St.
4	Daryl	Spence	NULL	daryl.spence@aol.com	988 Pearl Lane
5	Charolette	Rice	(916) 381-6003	charolette.rice@msn.com	107 River Dr.
6	Lyndsey	Bean	NULL	lyndsey.bean@hotmail.com	769 West Road
7	Latasha	Hays	(716) 986-3359	latasha.hays@hotmail.com	7014 Manor Station Rd.
8	Jacquline	Duncan	NULL	jacquline.duncan@yahoo.com	15 Brown St.
9	Genoveva	Baldwin	NULL	genoveva.baldwin@msn.com	8550 Spruce Drive
10	Pamelia	Newman	NULL	pamelia.newman@gmail.com	476 Chestnut Ave.

https://www.sqlservertutorial.net/sql-server-basics/sql-server-select/

WHAT IS AN ATTRIBUTE?

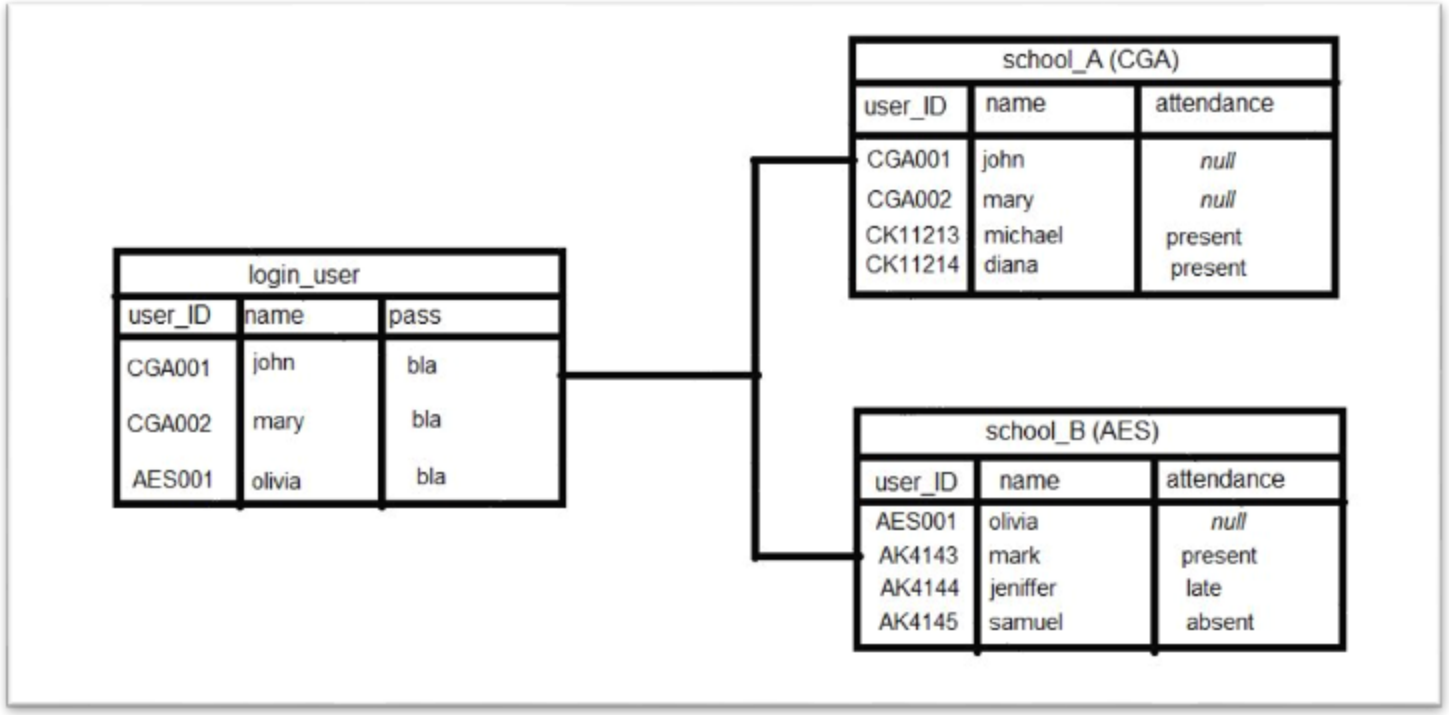
- Column of the table
- Value to describe a specific member
- Here, first_name, last_name, phone, etc.

customer_id	first_name	last_name	phone	email	street
1	Debra	Burks	NULL	debra.burks@yahoo.com	9273 Thorne Ave.
2	Kasha	Todd	NULL	kasha.todd@yahoo.com	910 Vine Street
3	Tameka	Fisher	NULL	tameka.fisher@aol.com	769C Honey Creek St.
4	Daryl	Spence	NULL	daryl.spence@aol.com	988 Pearl Lane
5	Charolette	Rice	(916) 381-6003	charolette.rice@msn.com	107 River Dr.
6	Lyndsey	Bean	NULL	lyndsey.bean@hotmail.com	769 West Road
7	Latasha	Hays	(716) 986-3359	latasha.hays@hotmail.com	7014 Manor Station Rd.
8	Jacquline	Duncan	NULL	jacquline.duncan@yahoo.com	15 Brown St.
9	Genoveva	Baldwin	NULL	genoveva.baldwin@msn.com	8550 Spruce Drive
10	Pamelia	Newman	NULL	pamelia.newman@gmail.com	476 Chestnut Ave.

https://www.sqlservertutorial.net/sql-server-basics/sql-server-select/

WHAT IS DATABASE SCHEMA?

- Structure of database in RDBMS
- Blueprint of the actual database
- Defines, tables, attributes and relationships.



https://stackoverflow.com/questions/42217521/how-to-connect-each-person-in-a-table-to-another-table-respectively-in-a-databas

ER MODEL – ENTITY RELATIONSHIP MODEL

- Designing database schema – all different tables and attributes in the table
- Defining relationships among them
- Middleman between actual database implemented and the business requirements
- Map out entities, attributes and relationships

BUSINESS REQUIREMENT

The school system must maintain a database of students, capturing essential details such as their first name, last name, GPA, and a unique student number. Students can enroll in multiple classes, and each class can accommodate multiple students, reflecting a many-to-many relationship. For each class a student takes, their grades should be recorded to track academic performance. Additionally, classes may vary in structure, as some will include exams while others will not. This structure should support flexibility and accuracy in tracking academic records and class configurations.

PROBLEM STATEMENT

- A school has many students.
- Each student has multiple attributes like first name, last name, GPA, student number.
- Each student can take multiple classes and classes can contain multiple students.
- A student will get a grade for the classes they take.
- Some classes will have exams, and some might not.

SOLUTION

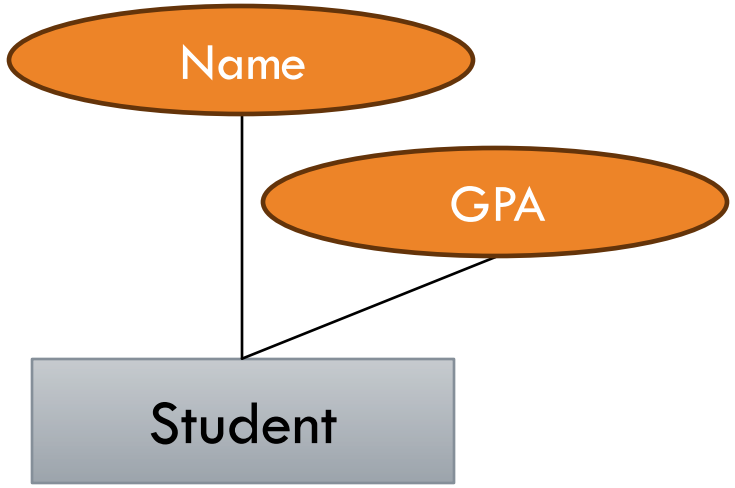
Build an ER Model!

ENTITY

- A thing or an object that can be defined and have stored information
- Think of them as nouns
- They are formed into tables
- Example: student, teacher, professor, department, etc.

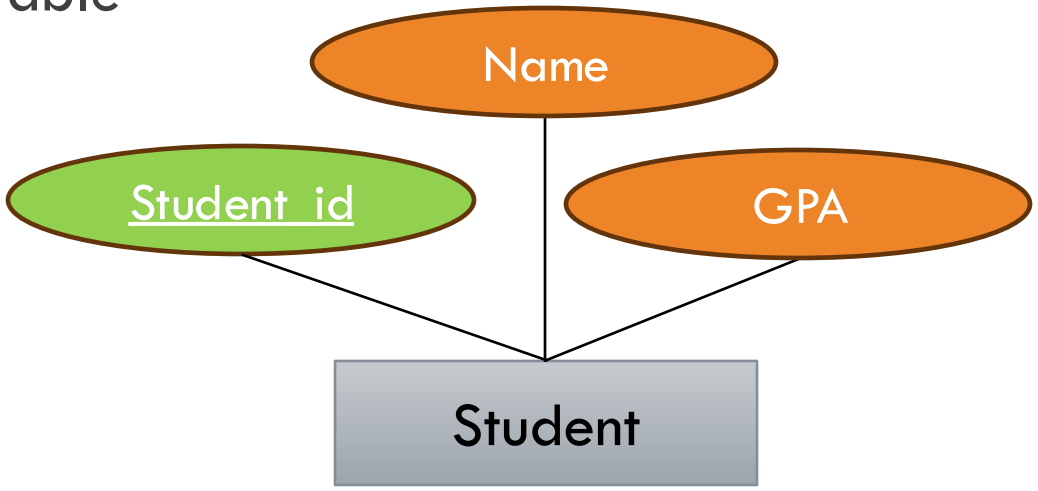
ENTITY ATTRIBUTES

- Specific information about the entity
- They describe the entity
- Columns of the table



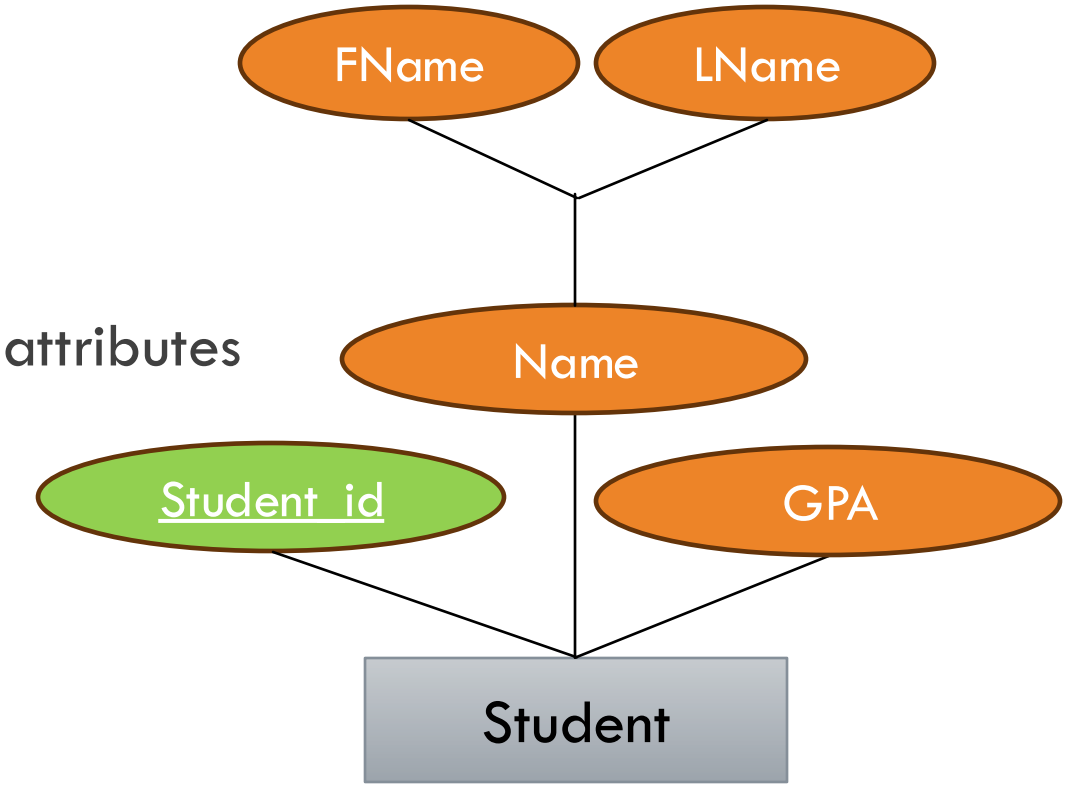
PRIMARY KEY

- An attribute that uniquely identifies an entry in the table
- Generally primary key is not in a different color
- Primary key is underlined to identify in ER diagram
- Student_id will be the primary key here



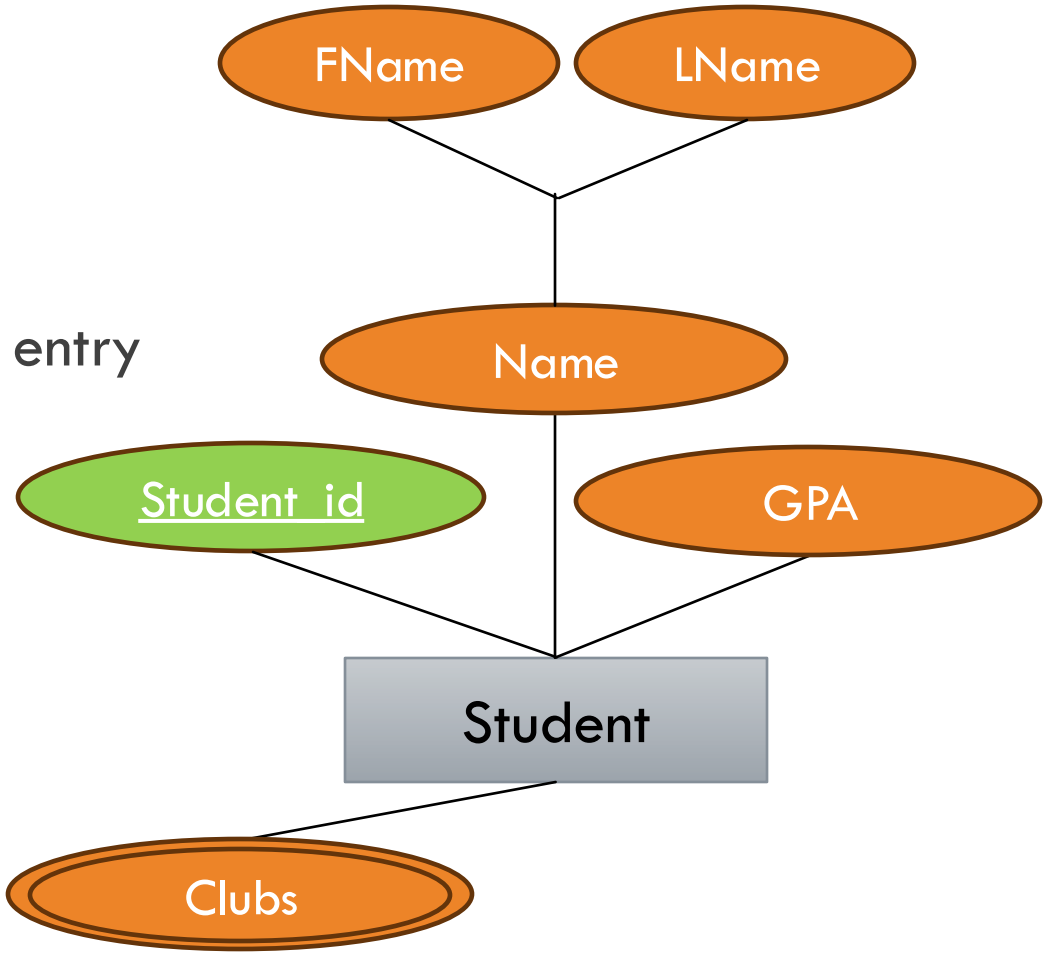
COMPOSITE ATTRIBUTE

- An attribute that can be broken down into smaller attributes
- Like FName and Lname for Name
- Main attribute connected to the sub attribues



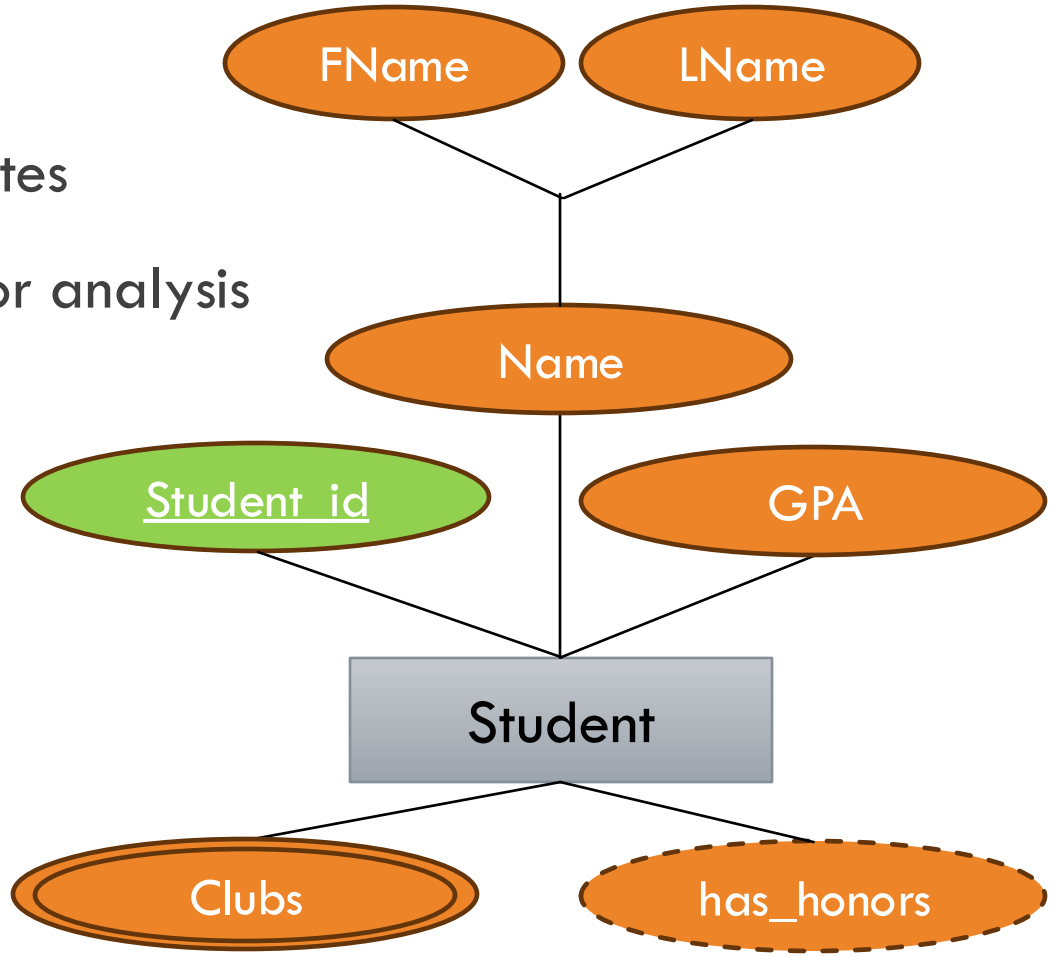
MULTI-VALUED ATTRIBUTES

- An attribute with more than one value for the same entry
- Looks like attribute with an extra circle
- Students can be included in more than one Clubs



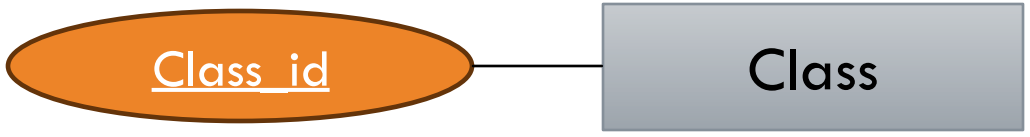
DERIVED ATTRIBUTE

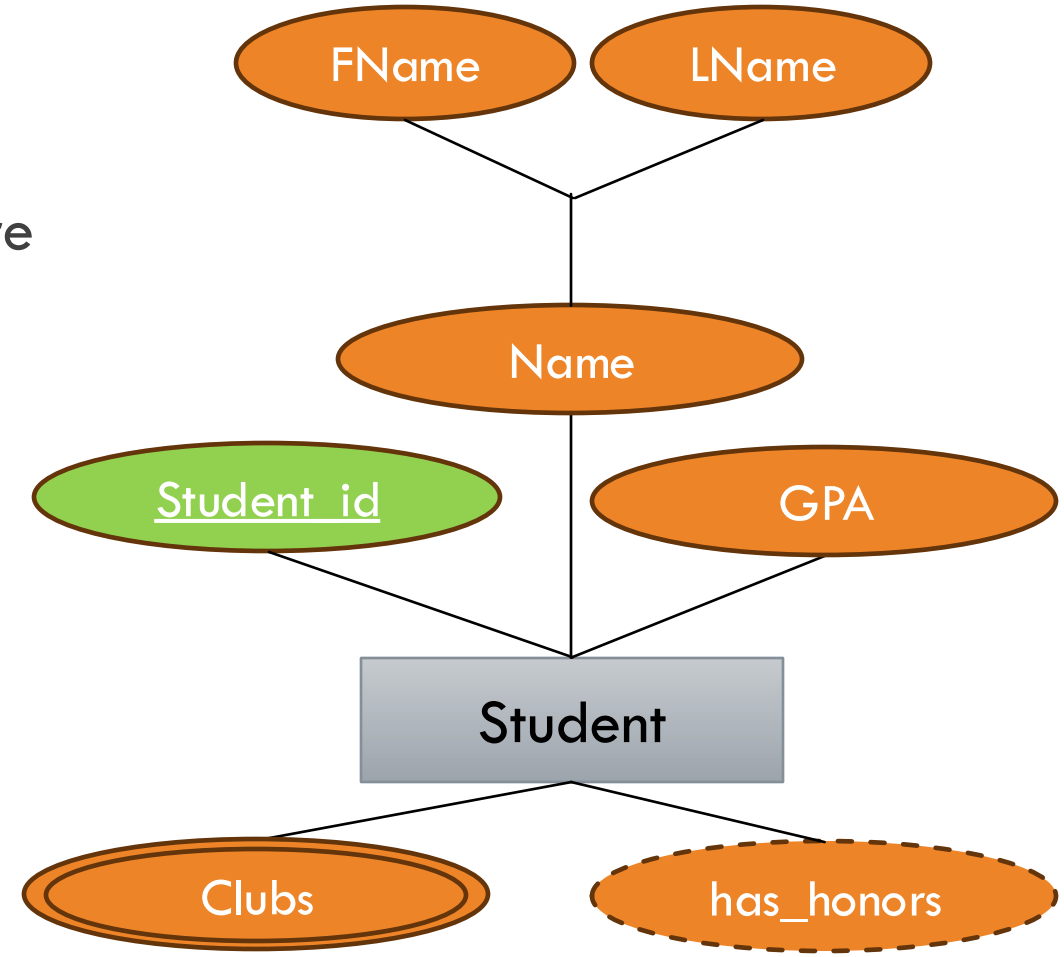
- An attribute that can be derived from other attributes
- We won't keep track of it, we could just derive it for analysis
- has_honors, derived from GPA
- Don't need to manually enter the value for it
- Calculate it at time of retrieval
- Circle with dashed border



MULTIPLE ENTITIES

- You can have as many entities as you like or require according to the problem statement
- We can have a class entity for storing the details about class like class_id, name, etc.



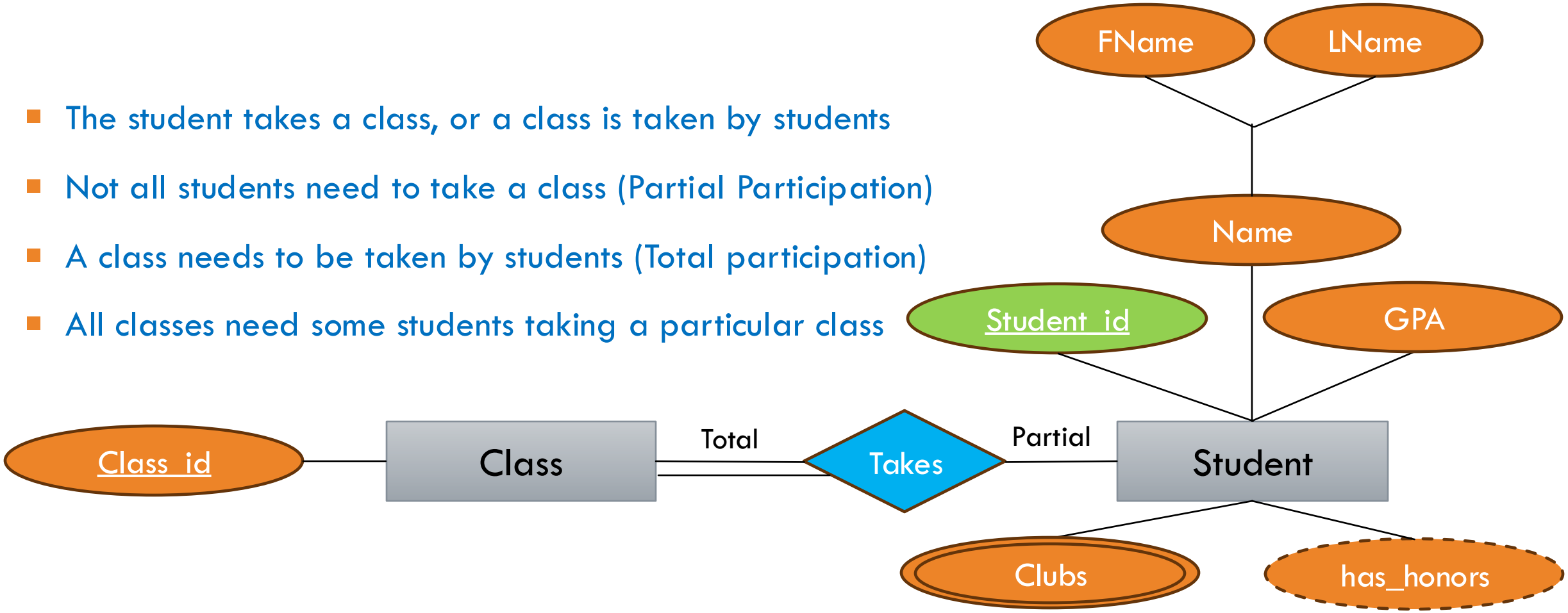


RELATIONSHIPS

- When we have multiple entities, we would want to define how they are related to each other
- Define a relationship in a diamond shape
- It's a verb, like takes, works, controls, etc.

RELATIONSHIPS

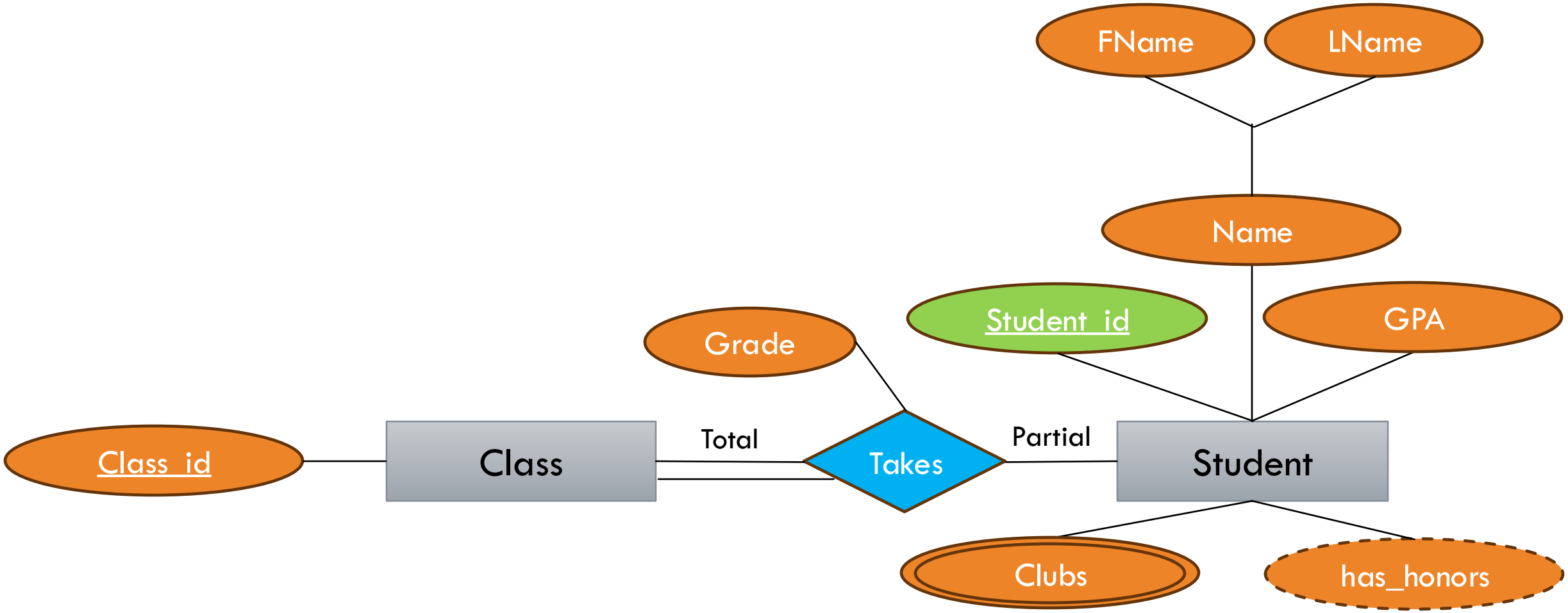
- The student takes a class, or a class is taken by students
- Not all students need to take a class (Partial Participation)
- A class needs to be taken by students (Total participation)
- All classes need some students taking a particular class



RELATIONSHIP ATTRIBUTE

- An attribute about the relationship
- Here, take relationship will give student a grade
- Grade isn't stored on student or class entity; it is only if a student takes it.

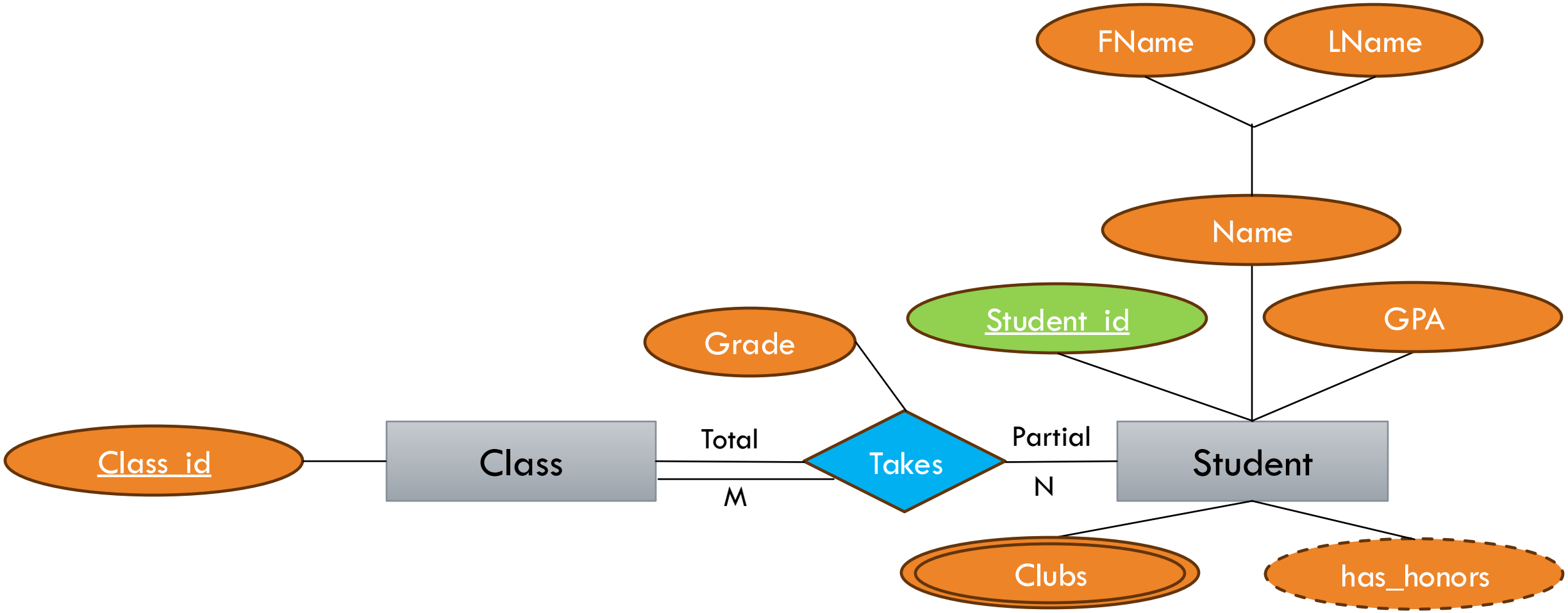
RELATIONSHIP ATTRIBUTE



RELATIONSHIP CARDINALITY

- Number of rows related between two entities
- 1:1 – A student can take one class, and class can be taken by one student
- 1:N – A student can take one class and a class can be taken by many students
- N: M – A student can take many classes and a class can be taken by any number any number of students
- Student and class are in many-to-many relationship
- Or a class could be taken by one student, but a student can take any number of classes

RELATIONSHIP CARDINALITY

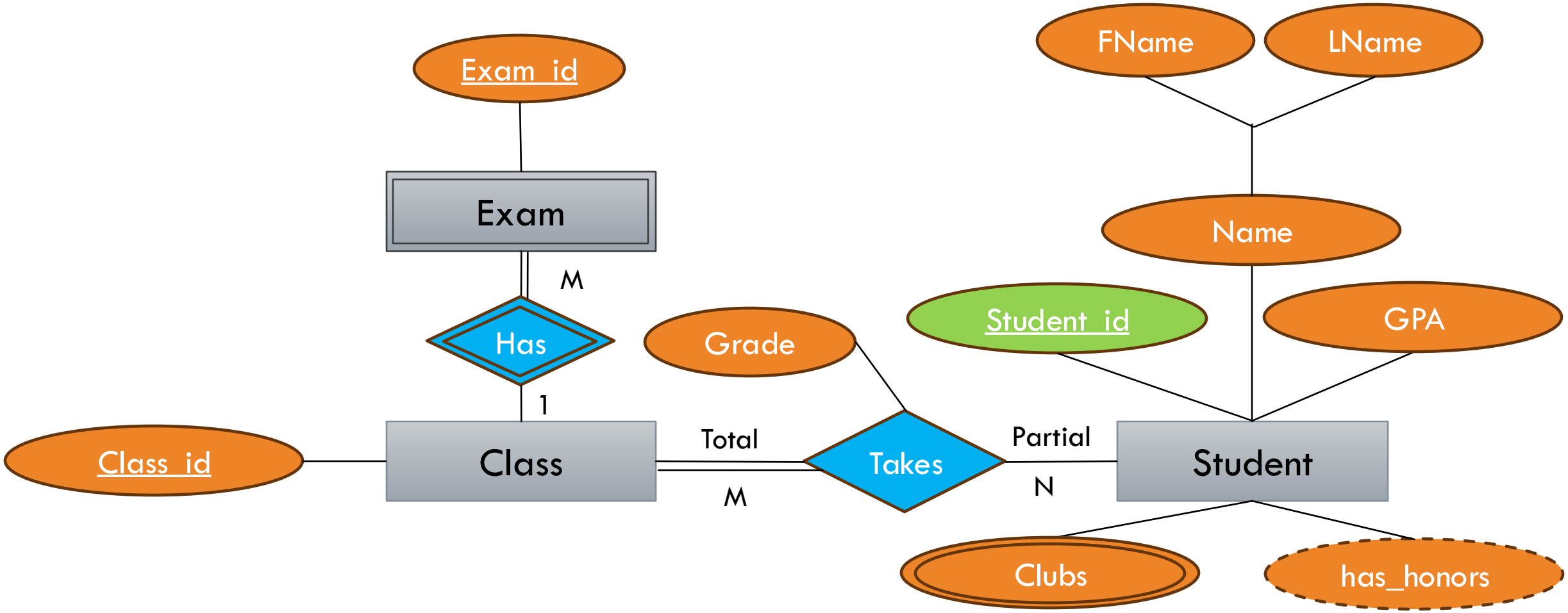


WEAK ENTITY

- An entity that cannot be uniquely identified by its attributes
- It is dependent on another entity
- Like, exam is an entity, but exam can't exist without a class, it must be associated with a class
- It's a double lined rectangle

IDENTIFYING RELATIONSHIP

- A relationship that serves to uniquely identify the weak entity
- It is a double lined diamond shape


Exam_id
Exam
M
Has
1
Class_id
Class
Total
M
Grade
Takes
Partial
N
Student
FName
LName
Name
Student_id
GPA
Clubs
has_honors

IMPORTANT NOTE

Whenever we have a weak entity and an identifying relationship, the weak entity always must have a total participation in the relationship.

Means all exams must have a class, but not all classes will have an exam.

PRACTICE ER MODELLING

LET'S DO AN EXAMPLE!

INTRODUCTION

- Document describing all different pieces of information and relation between them
- Usually a paragraph
- Make an ER diagram/Model from the document and then convert it into a database
- Information -> ER Diagram/Model -> Database Schema

BUSINESS REQUIREMENTS

The university system is structured into departments, each identified by a unique department number (dept_id) and name. Departments consist of multiple instructors and students. Students are uniquely identified by their student number, name, birth date, and age. Similarly, instructors are identified by their name and id. Each instructor is associated with only one department at a time, and every department is managed by a single instructor whose start date in the role must be tracked. Additionally, each department utilizes a subscribed service to support its operations, and these services are identified by a unique service_id and name.

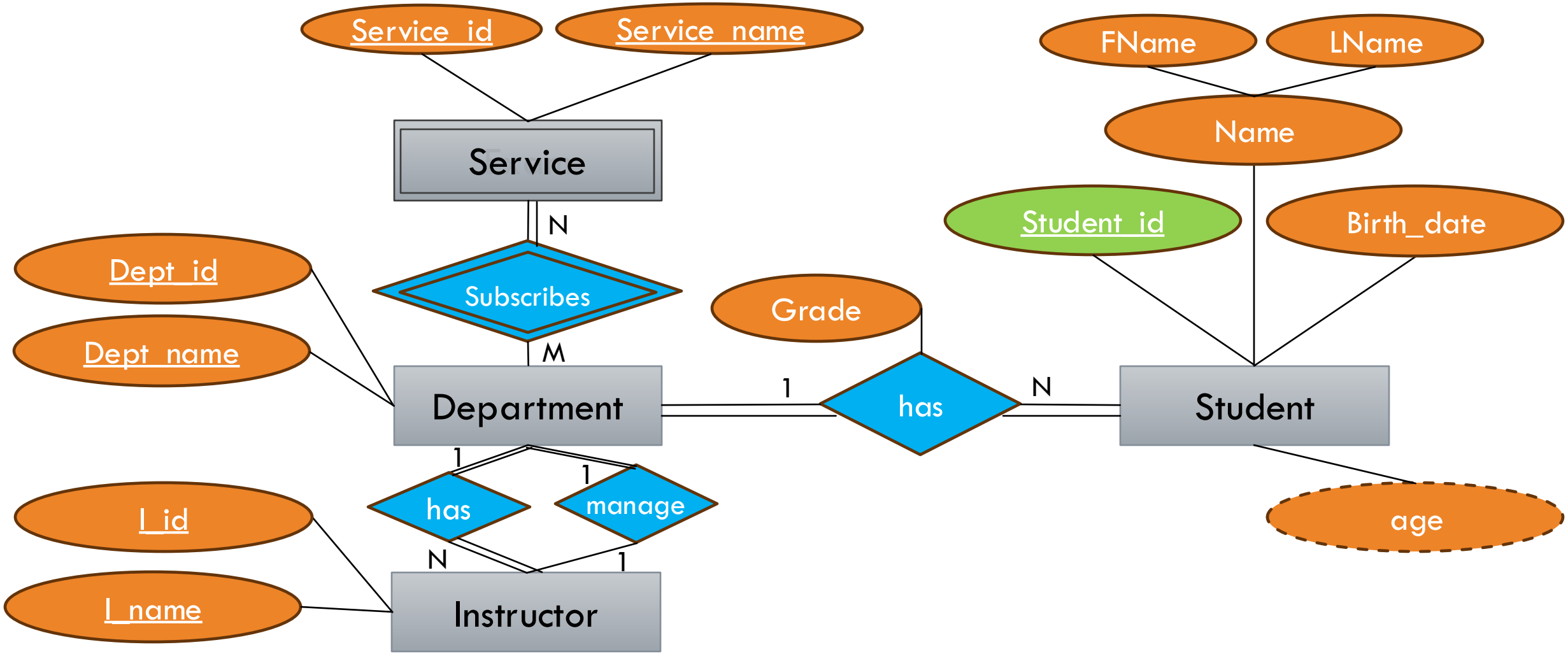
BUSINESS REQUIREMENTS

- A university is organized into departments. Each department has a unique number (dept_id), name.
- Each department has many instructors and many students. Every student has a unique number, name, birth_date and age to identify it. Every instructor has name and id.
- An instructor can be working for only one department at a time and each department will be managed by one instructor working it and we want to keep track of start date for them.
- Each department has a subscribed service they use for functioning. Each service has a service_id and name.

STEPS TO BUILD AN ER MODEL

- Identify the entities
- Identify the attributes for each entity
- Identify the primary key, derived attributes, etc.
- Identify relationship between each entity set

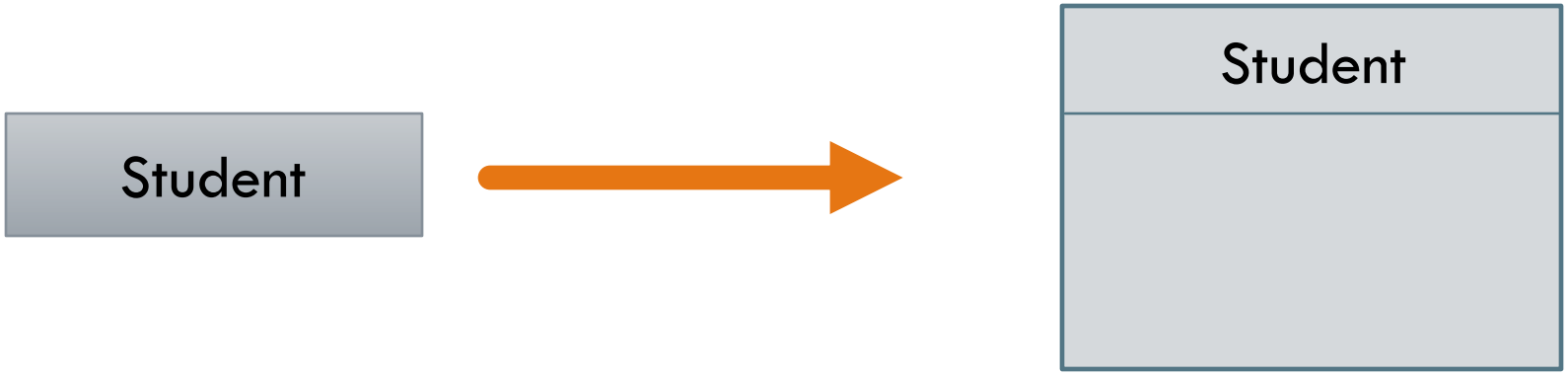
COMPLETE ER MODEL



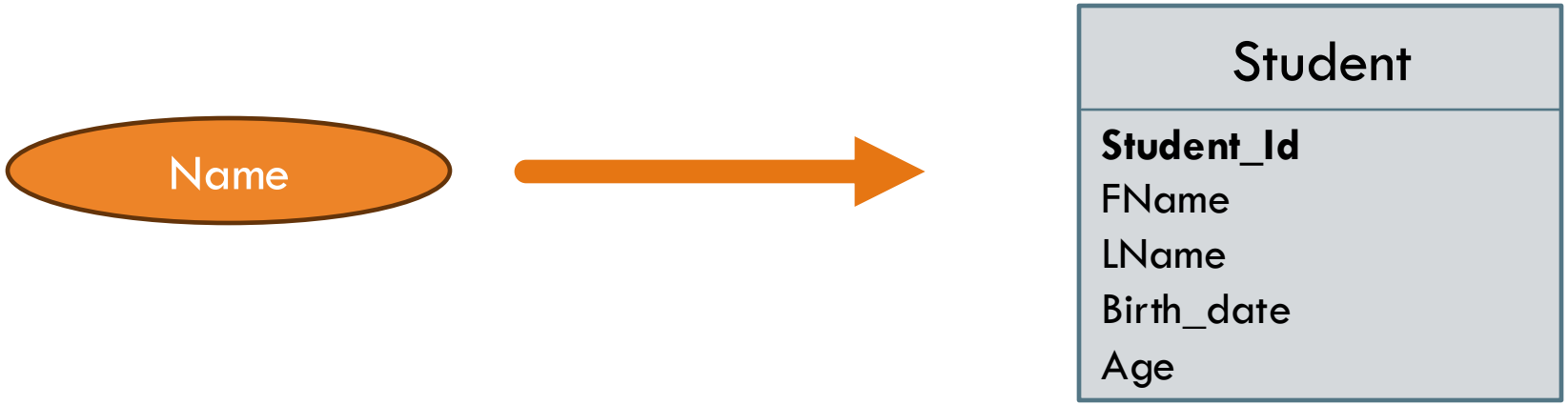
CONVERTING TO ER DIAGRAM

ER MODEL TO ER DIAGRAM

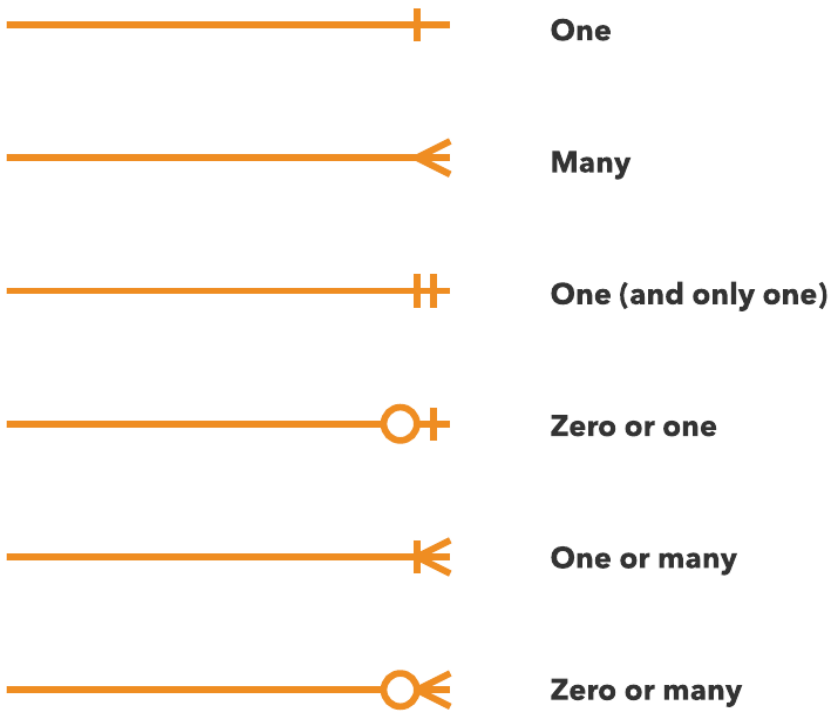
ENTITY



ATTRIBUTES



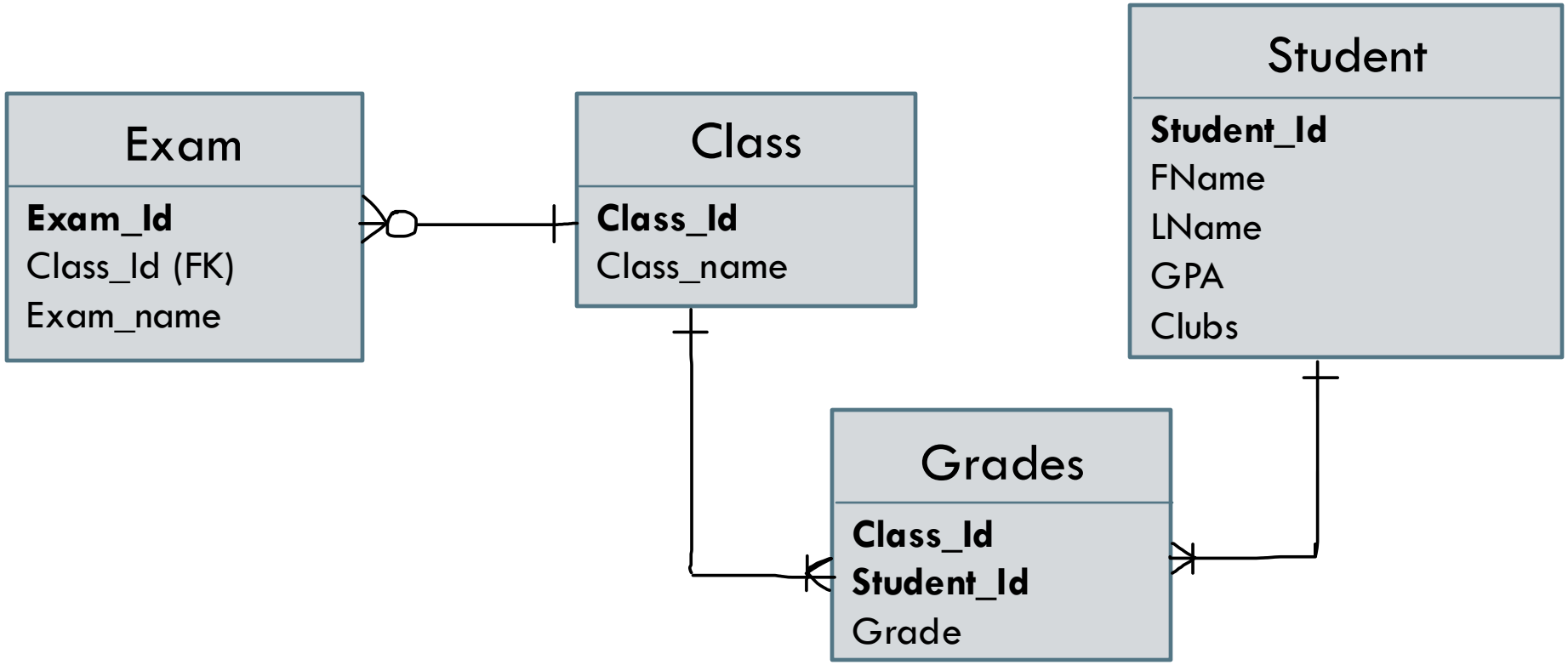
RELATIONSHIP CARDINALITY



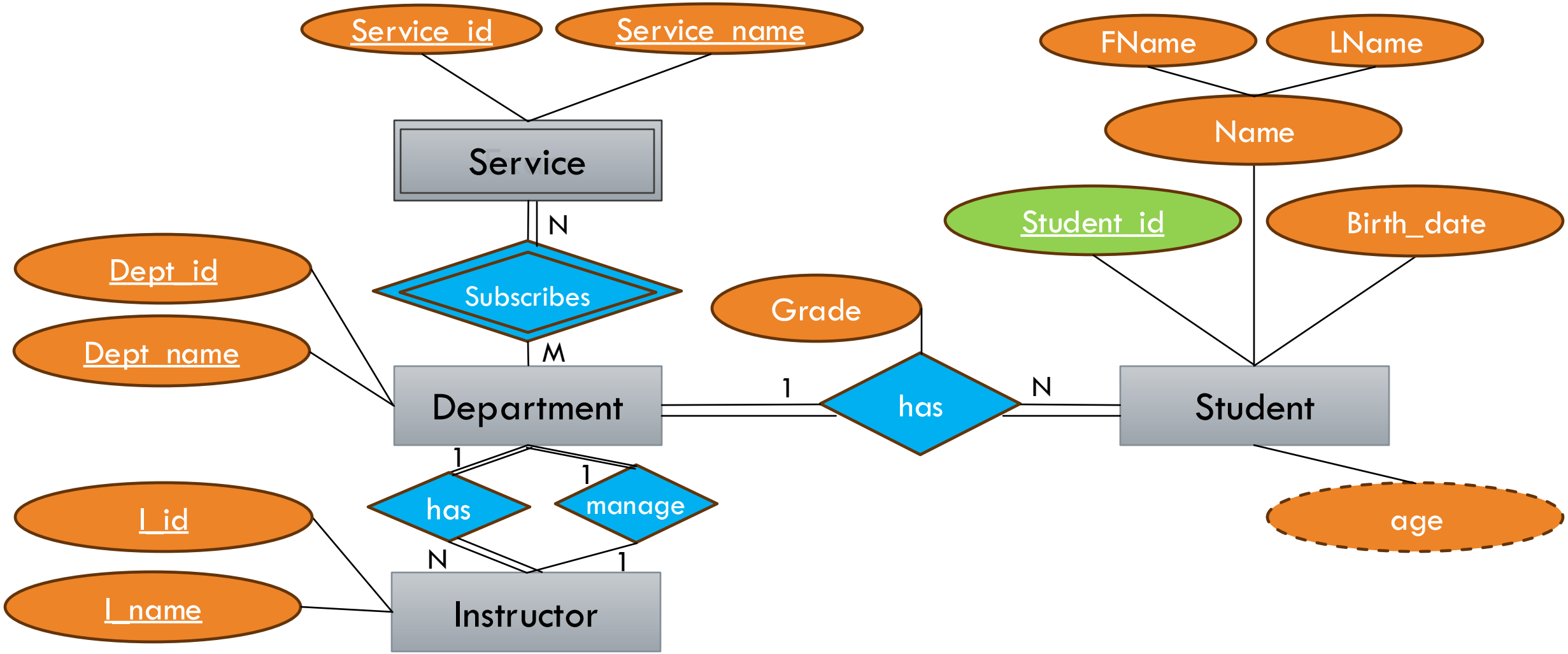
FOREIGN KEY

- Key to describe the link between 2 tables
- It is a primary key of another table that you are linking to
- Storing class_id in Exams table to find which classes have exams.

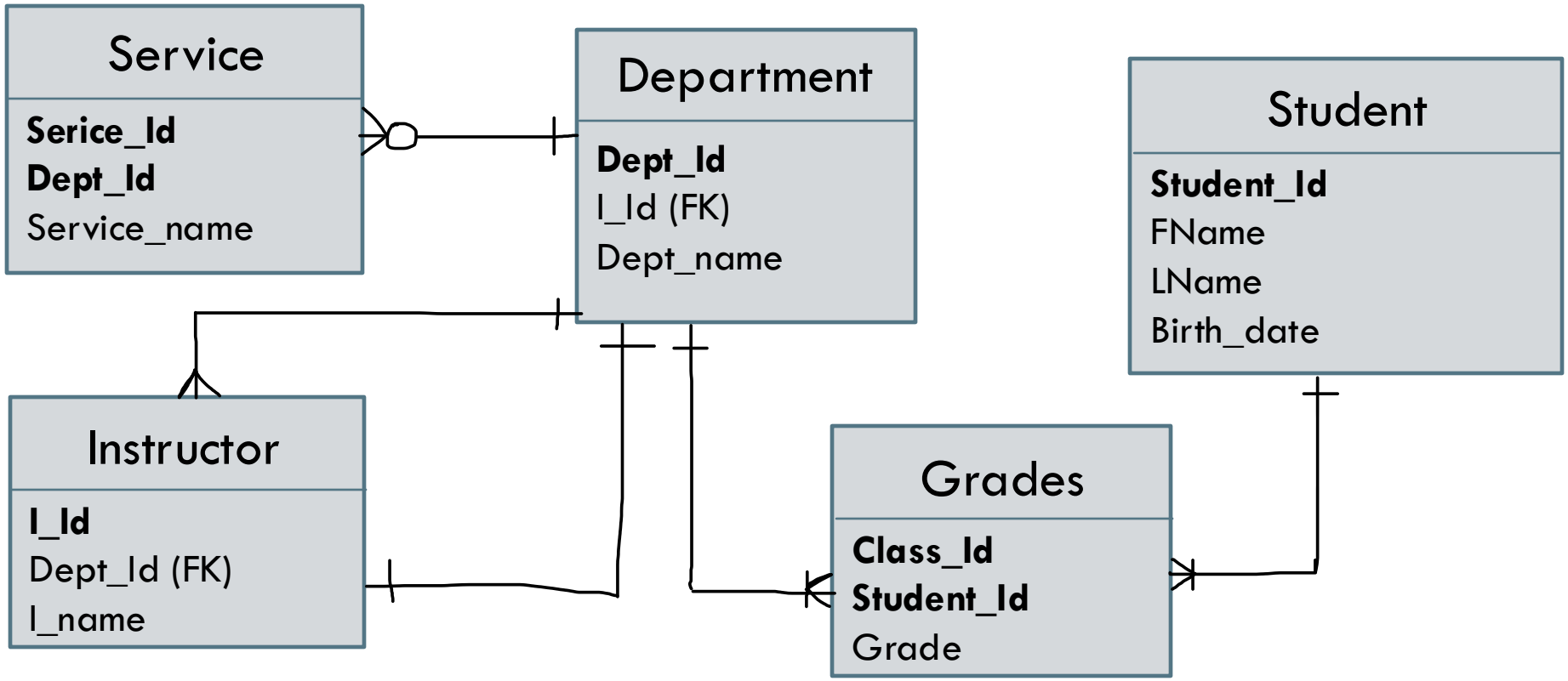
CONVERTING TO ER DIAGRAM



DESIGN ER DIAGRAM



UNIVERSITY ER DIAGRAM



SUMMARY

- ER Model contains, entities, attributes and relationships.
- ER Diagram/Model is the middleman between business requirement and the actual implementation of the database schema.
- Primary Key is the unique attribute to identify every row in the table.
- Foreign key is an attribute that links the table values to another table using their primary key