Let's Get Started

- To design a Relational Database, we start with an Entity Relationship Model (ER Model):
  - Describes what are the entities the database is going to record
  - Describes what are the attributes (and identifiers) of the entities
  - Describes how the relationships among these entities
  - To represent the Entity Relational Model in an explicit way, we use the Entity Relationship Diagram.

Entity Relationship Model

- An Entity Relationship model (or ER model) describes interrelated things of interest in a specific domain of knowledge.
- Concepts:
  - Entity
  - Attribute
  - Identifier
  - Relationship

Entity

- The identifiable abstract object of interest (in Programming, we call Class)
- Examples:
  - Students
  - Employees
  - Companies
  - Products
  - Transactions

Attributes

- Characteristics of an entity of interest.
- Examples:
  - Students: FirstName, StudentID, Major
  - Employees: ID, SSN, ContactInfo, Department, Supervisor
  - Companies: LegalName, Location, Category, Rank
  - Products: SKU#, Category, InStockPrice, InStockQuantity
  - Transactions: CustomerID, StoreID, ProductID, ProductDescription, Price, Quantity, Tax, Total, Time

Instance

- A record / member of an entity of interest (with actual value of attributes)
- Examples:
  - A Student: FirstName = "Joe", StudentID = "DB001", Major = "Data Science"
  - A Employee: ID = "E0099", SSN = "123-45-6789", ContactInfo = "N/A", Department = "Education", Supervisor = "Smith"

Identifiers

- A special attribute used to identify a specific instance of an entity
  - May be natural (your DBMS doesn't need to create it): SSN
  - May be artificial (your DBMS needs to create it): EmployeeID
- Examples:
  - Students: FirstName, StudentID, Major
  - Employees: ID, SSN, ContactInfo, Department, Supervisor
  - Companies: LegalName, Location, Category, Rank
  - Products: SKU#, Category, InStockPrice, InStockQuantity