Lesson 5 & 6

Entity Relationship ER Diagram

Entity Relationship Diagram (ER Diagram)

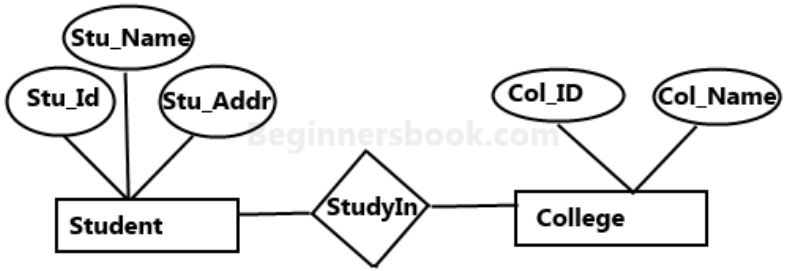
An entity relationship diagram (ERD) shows the **relationships** of entity sets stored in a database. An **entity** in this context is an object, a component of data. An entity set is a collection of similar entities. These entities can have **attributes** that define its properties.

By defining the entities, their attributes, and showing the relationships between them, an ER diagram illustrates the logical structure of databases.

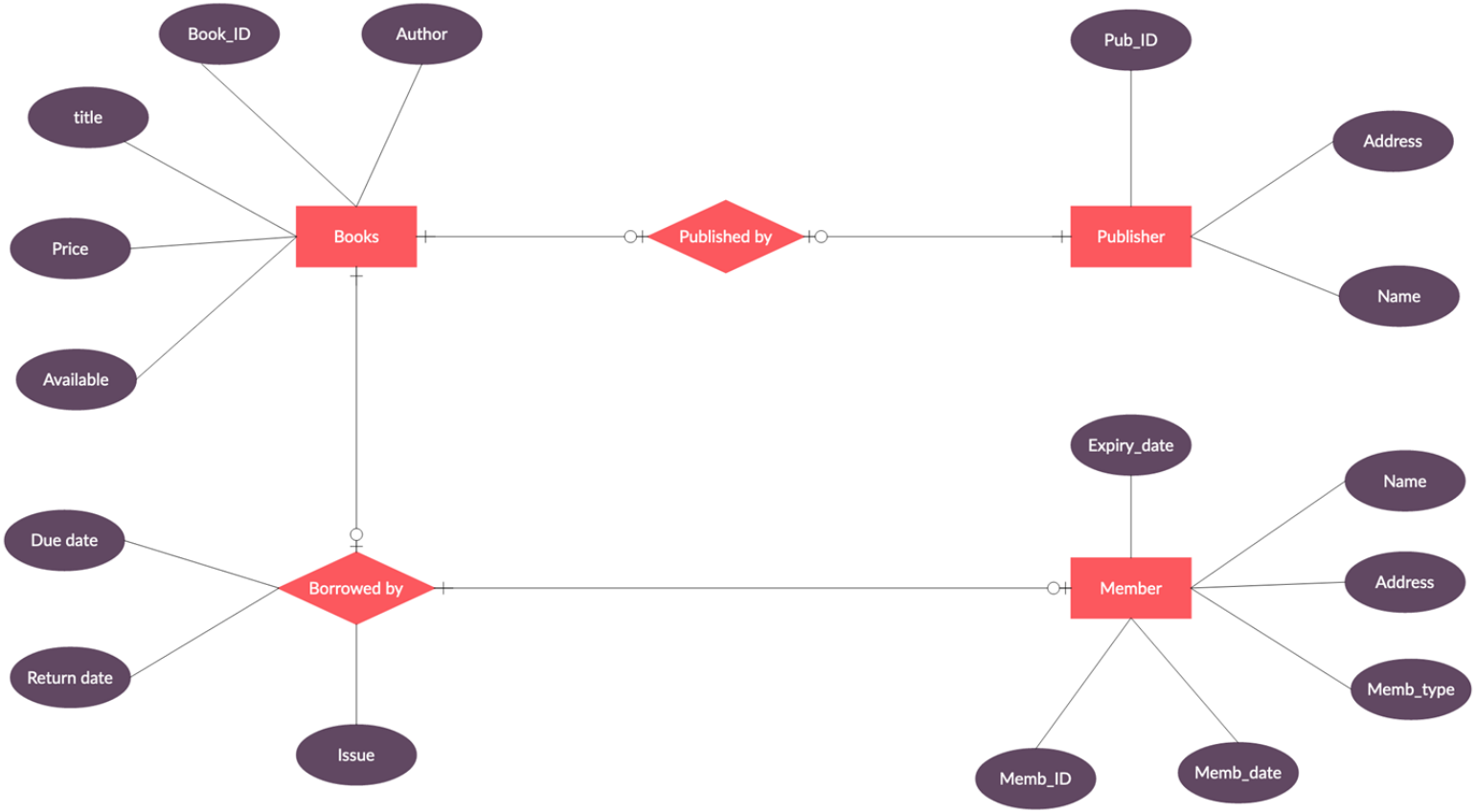
ER diagrams are used to **sketch out the design of a database**.

Entity Relationship Diagram (ER Diagram)

There are two reasons to create a database diagram. You're either designing a new schema or you need to document your existing structure.



Sample E-R Diagram


Book_ID
Author
title
Price
Available
Books
Published by
Pub_ID
Address
Name
Publisher
Expiry_date
Name
Address
Memb_type
Member
Memb_ID
Memb_date
Due date
Return date
Borrowed by
Issue

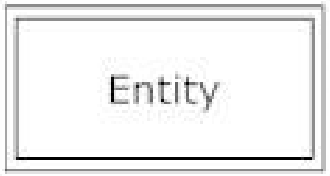
Common Entity Relationship Diagram Symbols - **Entity**

An ER diagram is a means of visualizing how the information a system produces is related. There are five main components of an ERD: **Entities**, which are represented by rectangles.

An entity is an object or concept about which you want to store information, such as a person/role (e.g. Student), object (e.g. Invoice), concept (e.g. Profile) or event (e.g. Transaction)

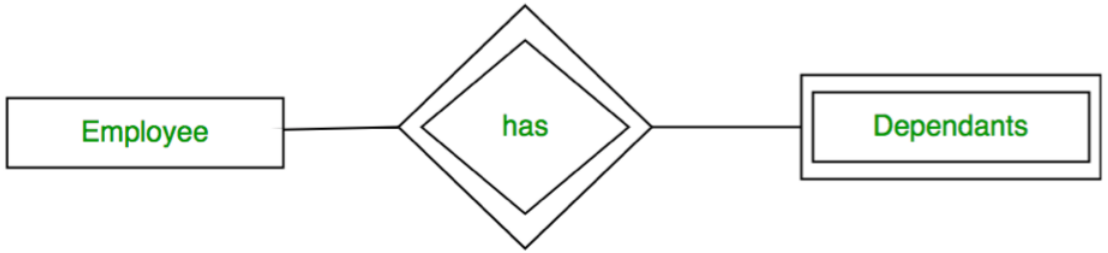
Common Entity Relationship Diagram Symbols - **Entity**

A **weak entity** is an entity that must defined by a **foreign key** relationship with another entity as it cannot be uniquely identified by its own attributes alone.



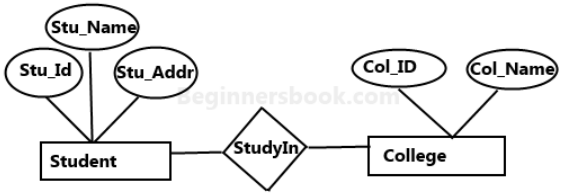
Common Entity Relationship Diagram Symbols - **Entity**

A **weak entity**



	STRONG ENTITY	WEAK ENTITY
1.	Strong entity always has primary key.	While weak entity has partial discriminator key.
2.	Strong entity is not dependent of any other entity.	Weak entity is depend on strong entity.
3.	Strong entity is represented by single rectangle.	Weak entity is represented by double rectangle.
4.	Two strong entity’s relationship is represented by single diamond.	While the relation between one strong and one weak entity is represented by double diamond.

Common Entity Relationship Diagram Symbols - **Attributes**

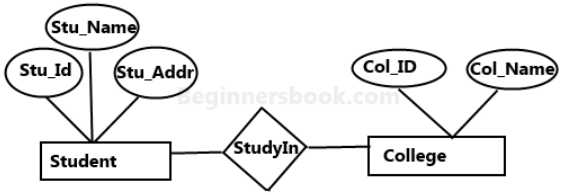


Sample E-R Diagram

Attributes, which are represented by ovals. A key attribute is the unique, distinguishing characteristic of the entity.

For example, an employee's social security number might be the employee's key attribute.

Common Entity Relationship Diagram Symbols - **Attributes**



Sample E-R Diagram

Composite Attribute

A **multi-valued** attribute can have more than one value. For example, an employee entity can have multiple skill values.

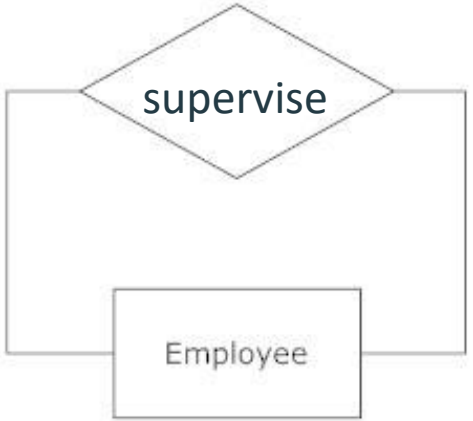
A **derived** attribute is based on another attribute. For example, an employee's monthly salary is based on the employee's annual salary.

Common Entity Relationship Diagram Symbols - **Actions (Relationships)**

Actions, which are represented by diamond shapes, show how two entities share information in the database.

Common Entity Relationship Diagram Symbols - **Actions**

In some cases, entities can be self-linked.
For example, employees can **supervise** other employees.



Draw ER Diagram for the database

EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1

DEPARTMENT

Dname	Dnumber	Mgr_ssn	Mgr_start_date
Research	5	333445555	1988-05-22
Administration	4	987654321	1995-01-01
Headquarters	1	888665555	1981-06-19

DEPT_LOCATIONS

Dnumber	Dlocation
1	Houston
4	Stafford
5	Bellaire
5	Sugarland
5	Houston

WORKS_ON

Essn	Pno	Hours
123456789	1	32.5
123456789	2	7.5
666884444	3	40.0
453453453	1	20.0
453453453	2	20.0
333445555	2	10.0
333445555	3	10.0
333445555	10	10.0
333445555	20	10.0
999887777	30	30.0
999887777	10	10.0
987987987	10	35.0
987987987	30	5.0
987654321	30	20.0
987654321	20	15.0
888665555	20	NULL

PROJECT

Pname	Pnumber	Plocation	Dnum
ProductX	1	Bellaire	5
ProductY	2	Sugarland	5
ProductZ	3	Houston	5
Computerization	10	Stafford	4
Reorganization	20	Houston	1
Newbenefits	30	Stafford	4

DEPENDENT

Essn	Dependent_name	Sex	Bdate	Relationship
333445555	Alice	F	1986-04-05	Daughter
333445555	Theodore	M	1983-10-25	Son
333445555	Joy	F	1958-05-03	Spouse
987654321	Abner	M	1942-02-28	Spouse
123456789	Michael	M	1988-01-04	Son
123456789	Alice	F	1988-12-30	Daughter
123456789	Elizabeth	F	1967-05-05	Spouse

Transfer the following verbal description into an ERM:

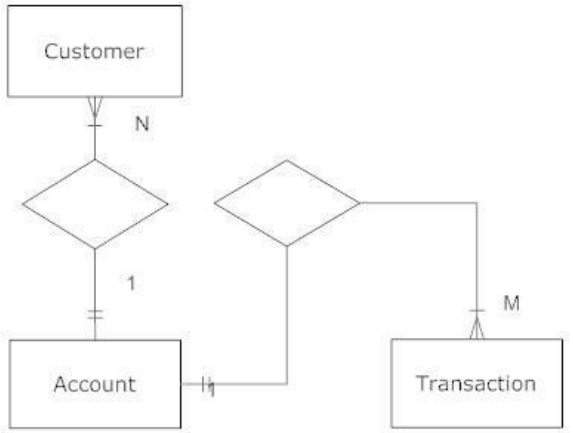
1. Customers (Cust-#) get discounts (Disc-#) on items (Item-#).
   Each item can only have one discount rate.
2. Items belong to a single category (Categ-ID).

Common Entity Relationship Diagram Symbols - **Cardinality**

Cardinality specifies how many **instances** of an entity relate to one instance of another entity.

For example, **ONE** team has **MANY** players. When present in an ERD, the entity **Team** and **Player** are inter-connected with a one-to-many relationship.

The three common cardinal relationships are **one-to-one**, **one-to-many**, and **many-to-many**.

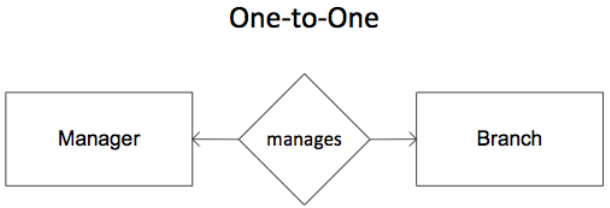


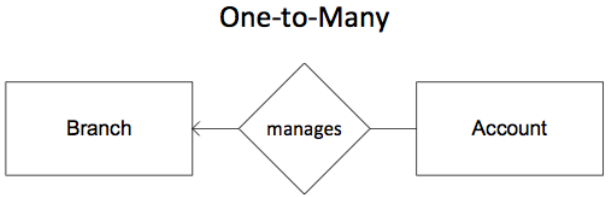
Cardinality

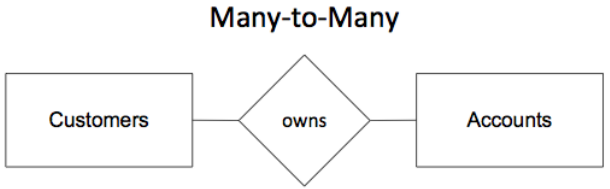
One-to-One cardinality

A **one-to-one** relationship is mostly used to split an entity in two to provide information concisely and make it more understandable.

The figure below shows an example of a one-to-one relationship.





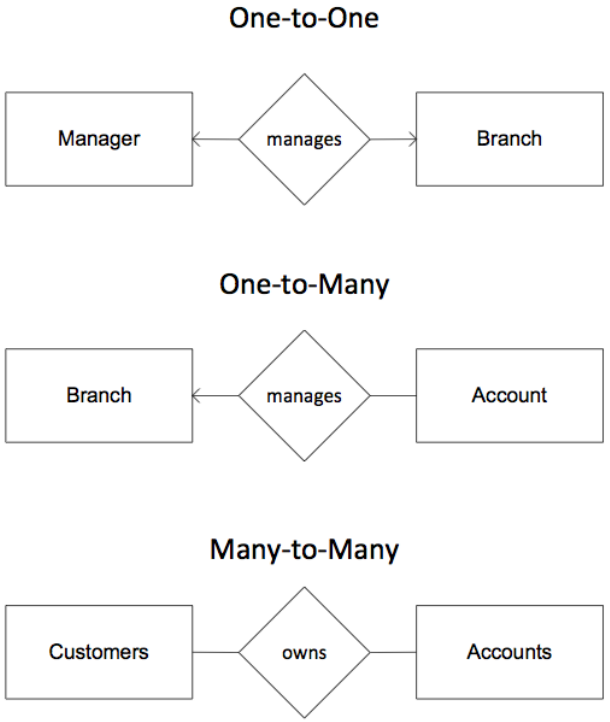


Cardinality

One-to-Many cardinality

A **one-to-many** relationship refers to the relationship between two entities X and Y in which an instance of X may be linked to many instances of Y, but an instance of Y is linked to only one instance of X.

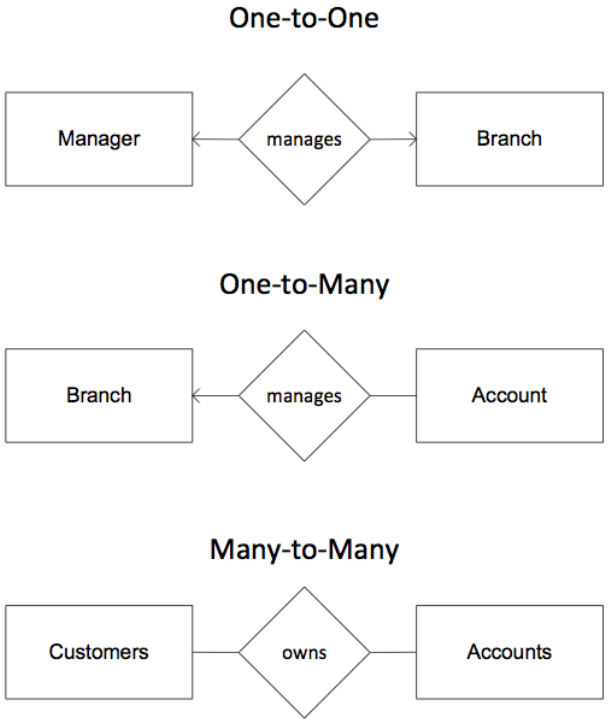
The figure below shows an example of a one-to-many relationship.



Cardinality

Many-to-Many cardinality

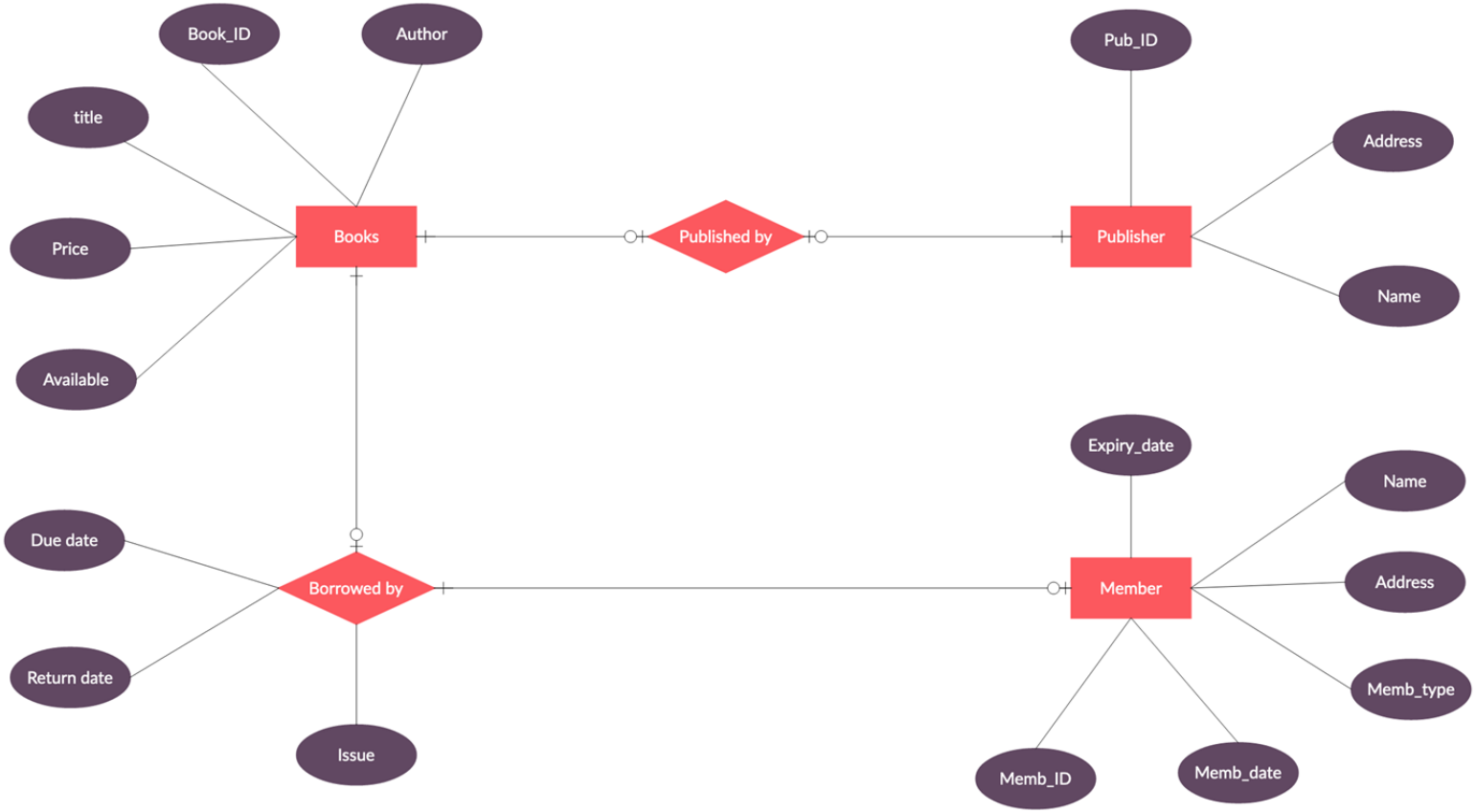
A many-to-many relationship refers to the relationship between two entities X and Y in which X may be linked to many instances of Y and vice versa.



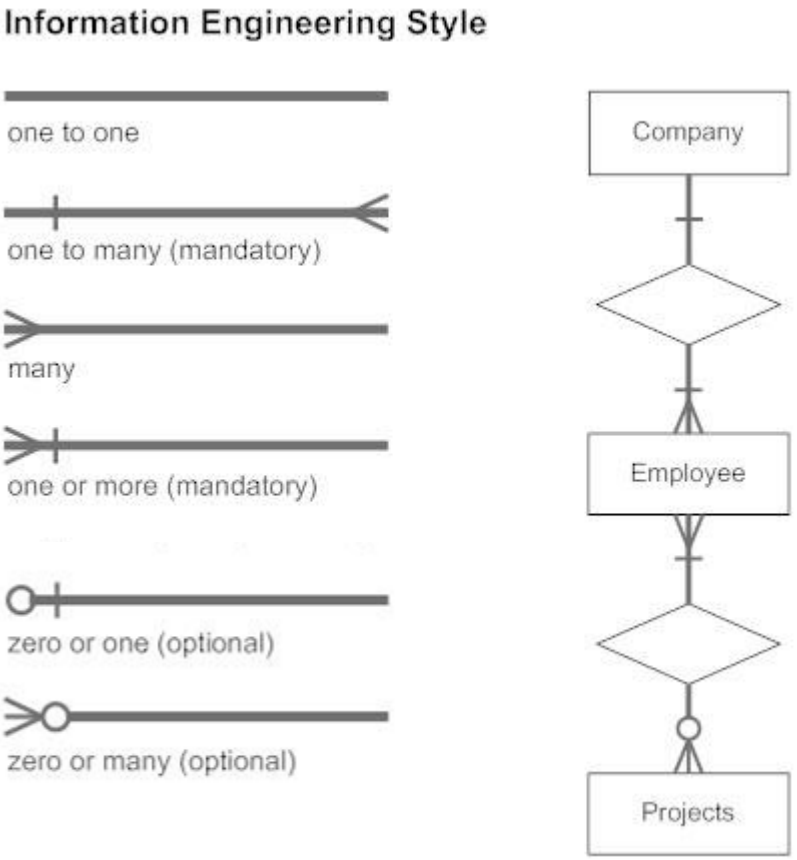
Common Entity Relationship Diagram Symbols - **Cardinality**

While **cardinality** specifies the **occurrences** of a relationship, **ordinality** describes the **relationship** as either mandatory or optional.

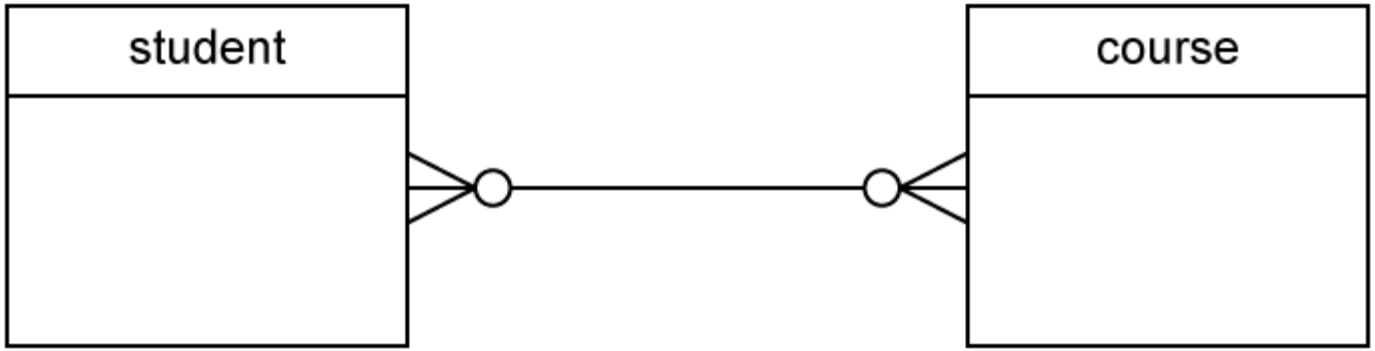
In other words, cardinality specifies the maximum number of relationships and ordinality specifies the absolute minimum number of relationships.


Book_ID
Author
title
Price
Available
Books
Published by
Pub_ID
Address
Name
Publisher
Expiry_date
Name
Address
Memb_type
Member
Memb_ID
Memb_date
Due date
Return date
Borrowed by
Issue

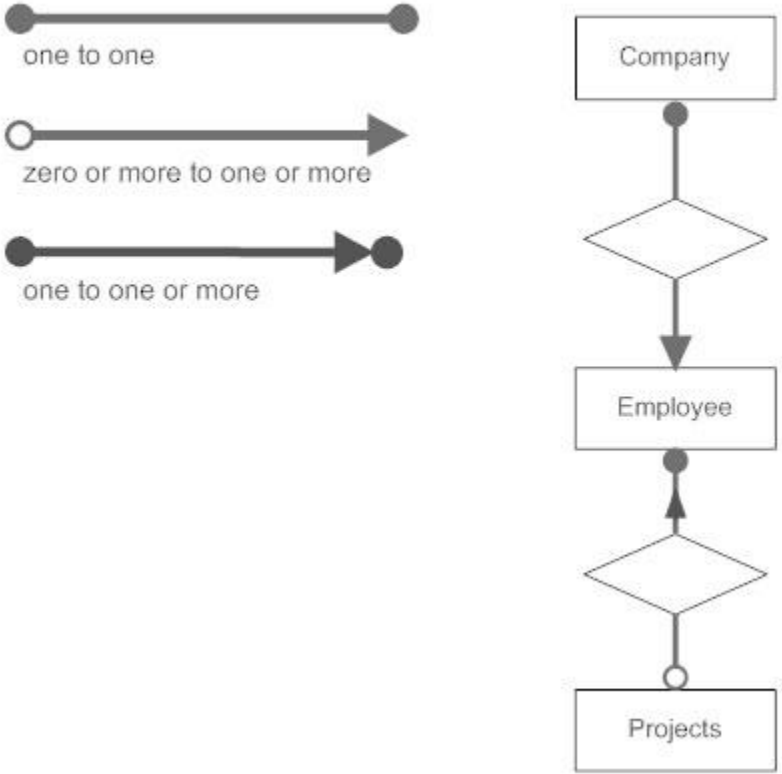
Crow's Foot


Information Engineering Style
one to one
one to many (mandatory)
many
one or more (mandatory)
zero or one (optional)
zero or many (optional)
Company
Employee
Projects

Crow’s Foot



Bachman Style



Martin Style

1 - one, and only one (mandatory)

* - many (zero or more - optional)

1...* - one or more (mandatory)

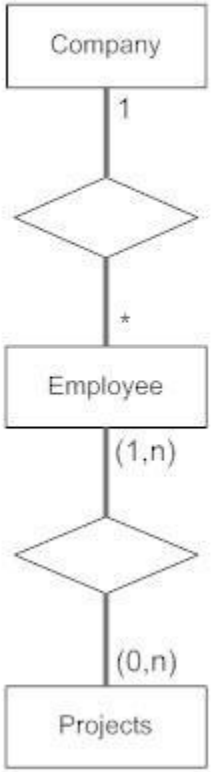
0...1 - zero or one (optional)

(0,1) - zero or one (optional)

(1,n) - one or more (mandatory)

(0,n) - zero or more (optional)

(1,1) - one and only one (mandatory)



Constructing ER Diagram

Identify the entities.

The first step in making an ERD is to identify all of the entities you will use. An entity is nothing more than a rectangle with a description of something that your system stores information about.

This could be a customer, a manager, an invoice, a schedule, etc. Draw a rectangle for each entity you can think of on your page. Keep them spaced out a bit.

Constructing ER Diagram

Identify relationships.

Look at two entities, are they related? If so draw a solid line connecting the two entities.

Describe the relationship.

How are the entities related? Draw an action diamond between the two entities on the line you just added. In the diamond write a brief description of how they are related.

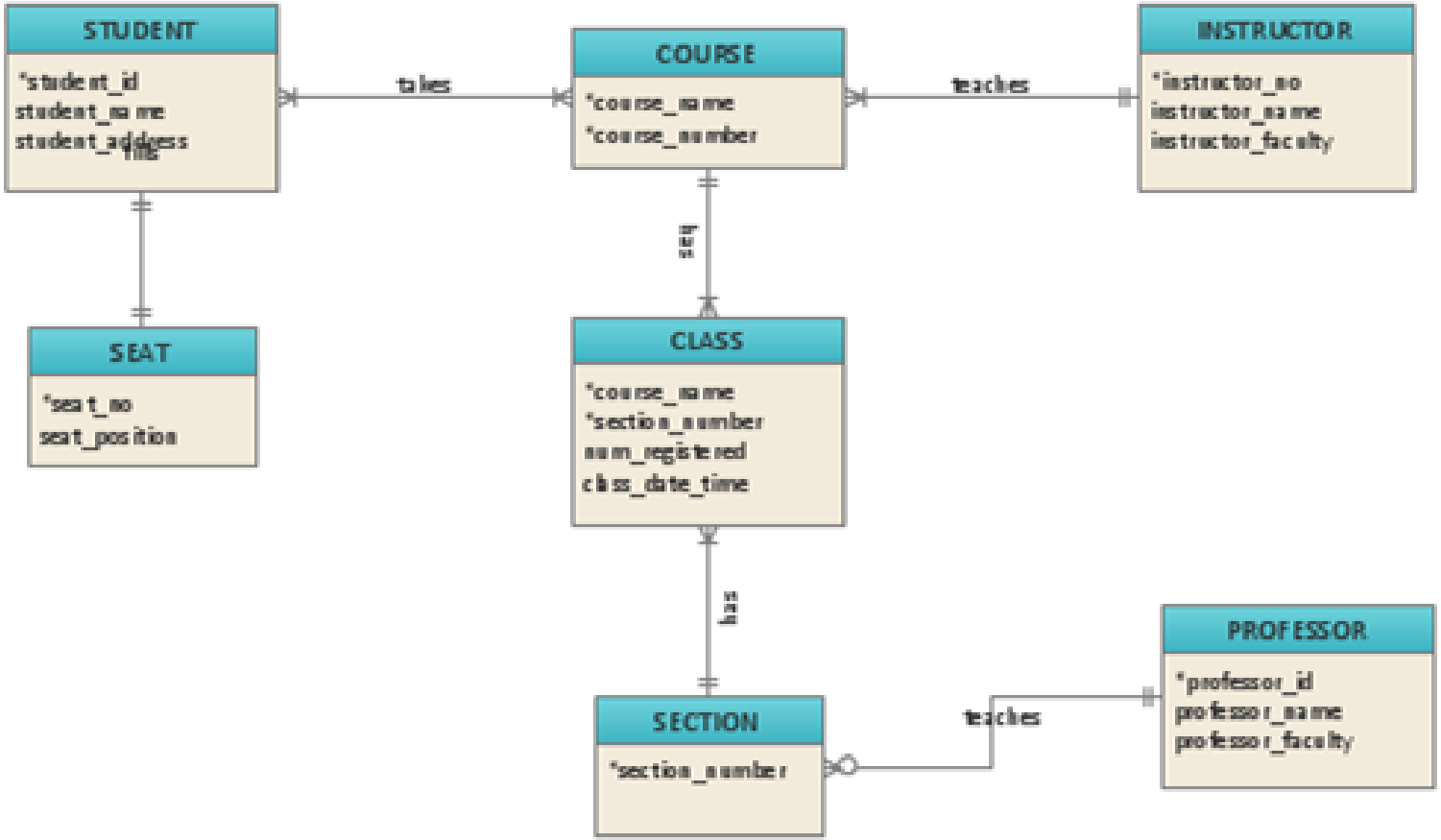
Constructing ER Diagram

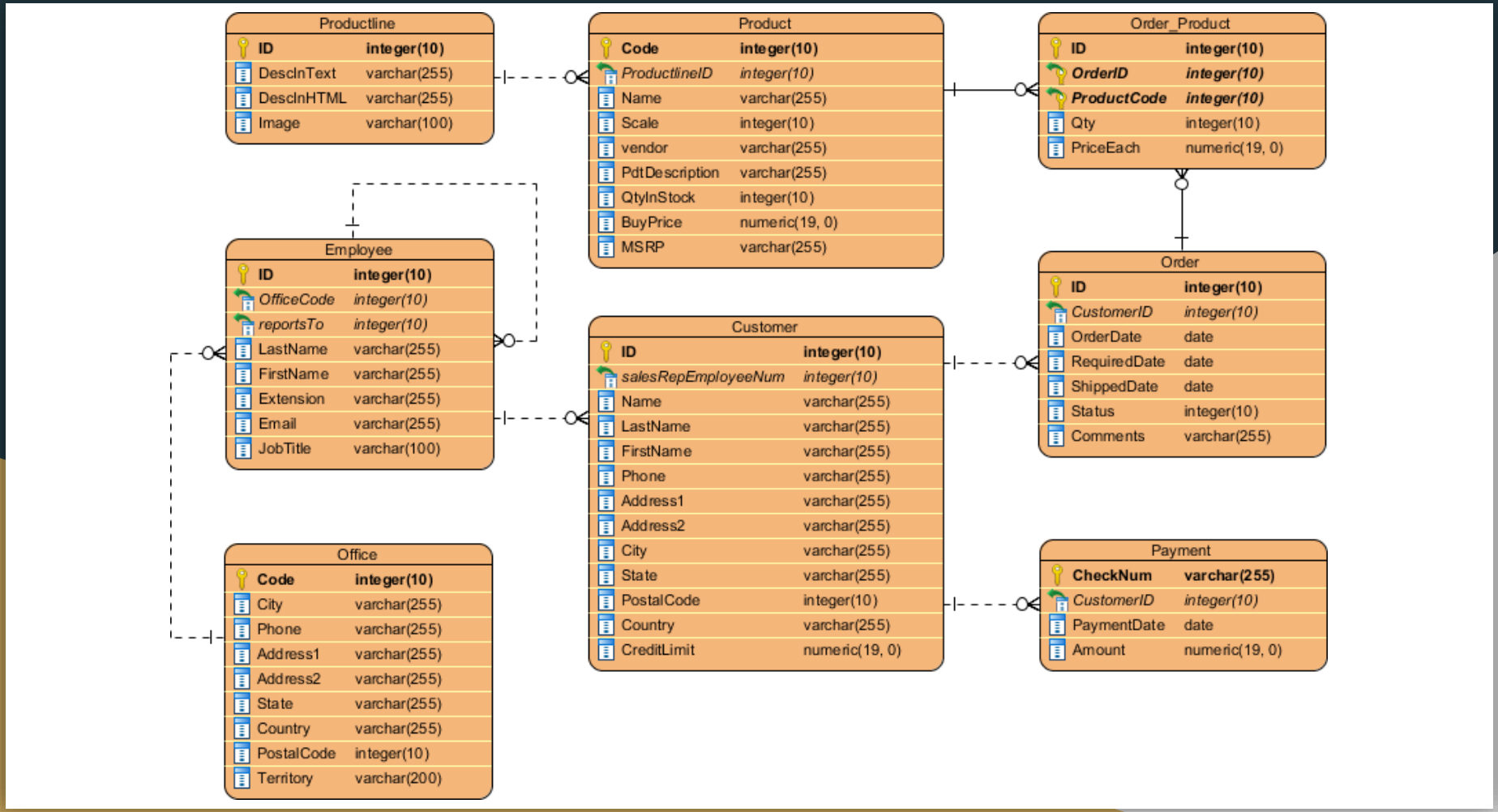
Add attributes.

Any key attributes of entities should be added using oval-shaped symbols.

Complete the diagram.

Continue to connect the entities with lines, and adding diamonds to describe each relationship until all relationships have been described. Each of your entities may not have any relationships, some may have multiple relationships. That is okay.


STUDENT
*student_id
student_name
student_address
takes
COURSE
*course_name
*course_number
teaches
INSTRUCTOR
*instructor_no
instructor_name
instructor_faculty
has
SEAT
*seat_no
seat_position
CLASS
*course_name
*section_number
num_registered
class_date_time
has
SECTION
*section_number
teaches
PROFESSOR
*professor_id
professor_name
professor_faculty


Productline
ID integer(10)
DescInText varchar(255)
DescInHTML varchar(255)
Image varchar(100)
Product
Code integer(10)
ProductlineID integer(10)
Name varchar(255)
Scale integer(10)
vendor varchar(255)
PdtDescription varchar(255)
QtyInStock integer(10)
BuyPrice numeric(19, 0)
MSRP varchar(255)
Order_Product
ID integer(10)
OrderID integer(10)
ProductCode integer(10)
Qty integer(10)
PriceEach numeric(19, 0)
Employee
ID integer(10)
OfficeCode integer(10)
reportsTo integer(10)
LastName varchar(255)
FirstName varchar(255)
Extension varchar(255)
Email varchar(255)
JobTitle varchar(100)
Order
ID integer(10)
CustomerID integer(10)
OrderDate date
RequiredDate date
ShippedDate date
Status integer(10)
Comments varchar(255)
Customer
ID integer(10)
salesRepEmployeeNum integer(10)
Name varchar(255)
LastName varchar(255)
FirstName varchar(255)
Phone varchar(255)
Address1 varchar(255)
Address2 varchar(255)
City varchar(255)
State varchar(255)
PostalCode integer(10)
Country varchar(255)
CreditLimit numeric(19, 0)
Office
Code integer(10)
City varchar(255)
Phone varchar(255)
Address1 varchar(255)
Address2 varchar(255)
State varchar(255)
Country varchar(255)
PostalCode integer(10)
Territory varchar(200)
Payment
CheckNum varchar(255)
CustomerID integer(10)
PaymentDate date
Amount numeric(19, 0)

The End