

# Entity Relationship Model

# Entity-Relationship (ER) Model

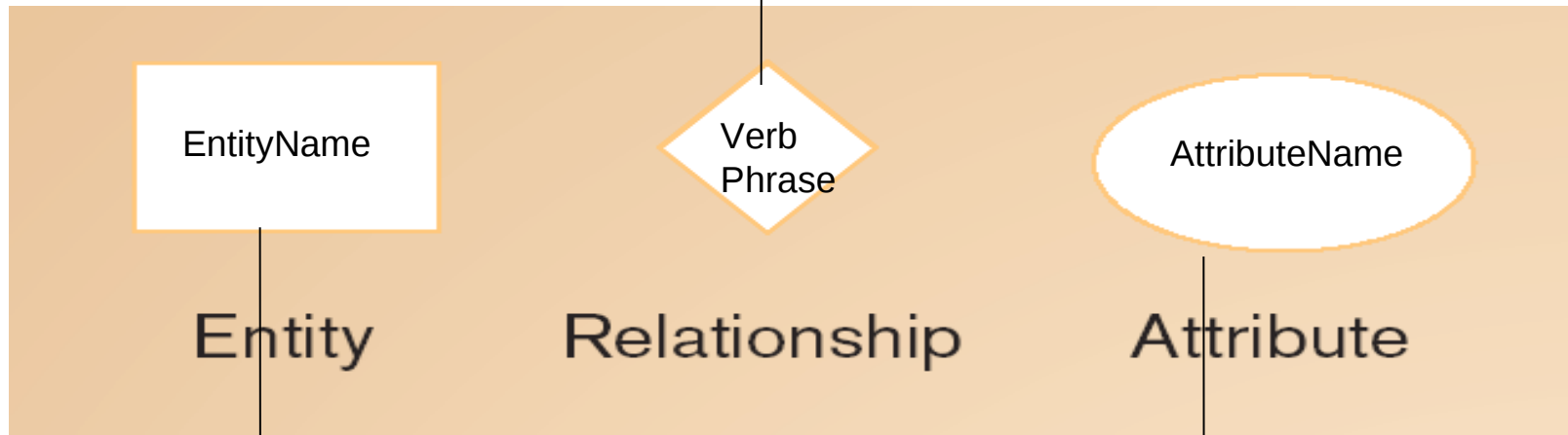
- **ER Modeling** is a *top-down* approach to database design.

- Notation uses three main constructs
  - Entities
  - Relationships
  - Attributes

Chen Model &  
Crow's Foot  
Model

# Chen Notation

Association between  
the instances of one  
or more entity types

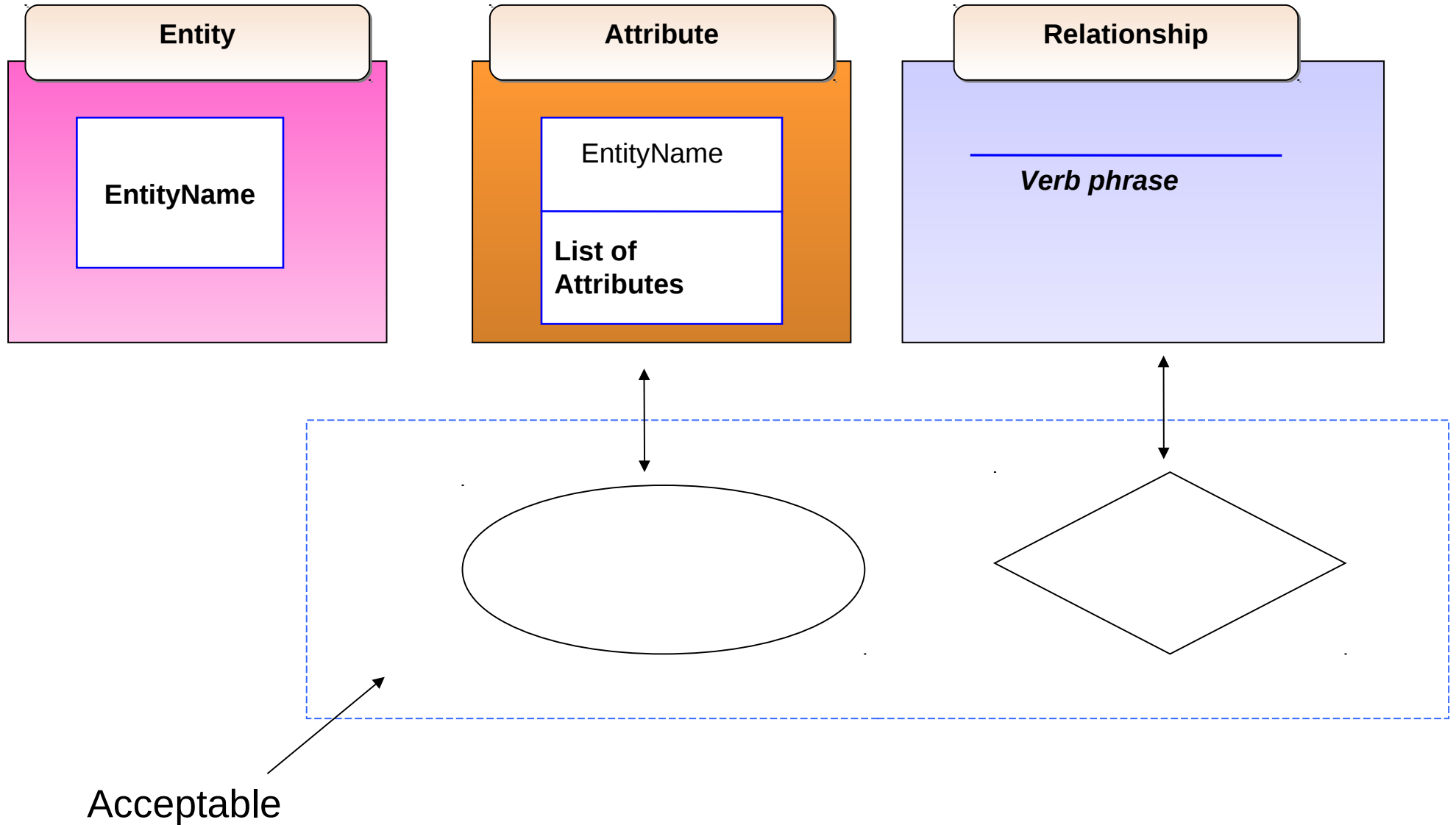


Person, place, object, event or  
concept about which data is to  
be maintained

named property or  
characteristic of an  
entity

Represents a set or collection of  
objects in the real world that  
share the same properties

# Crow's Foot Notation



# Entities

**Entity** is a thing or object in the real world that is distinguishable from other objects.

Example: specific person, company, event

**Entity set** is a set of entities of the same type that share the same properties or attributes.

Example: set of all persons, companies, trees, holidays

- Guidelines for naming and defining entity types:
  - An entity type name is a singular noun
  - An entity type should be descriptive and specific
  - Event entity types should be named for the result of the event, not the activity or process of the event.

An Entity

STUDENT

Entity set

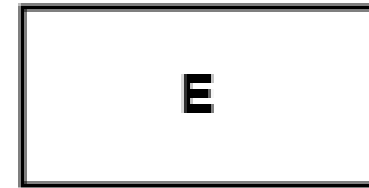
STUDENTS

# Entity Set

## Strong Entity Set-

A strong entity set possess its own primary key.

It is represented using a single rectangle.

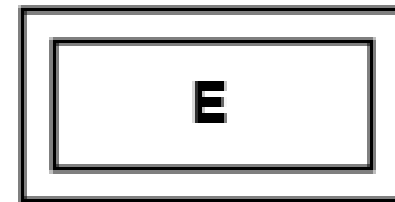


**Strong Entity Set**

## Weak Entity Set-

A weak entity set do not possess its own primary key.

It is represented using a double rectangle.



**Weak Entity Set**

# Attributes

Attribute is the particular properties or characteristics that describe entity.

An entity is represented by a set of attributes, that is descriptive properties possessed by entities of an entity set.

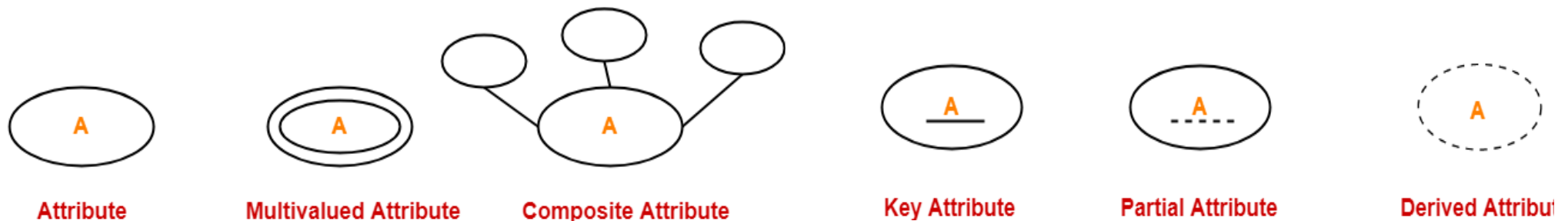
Example of entity types and associated attributes:

**STUDENT:** Student\_ID, Student\_Name, City, Phone\_Number

## Attribute & Attribute Value

**Student\_ID, Student\_Name, City, Phone\_Number**

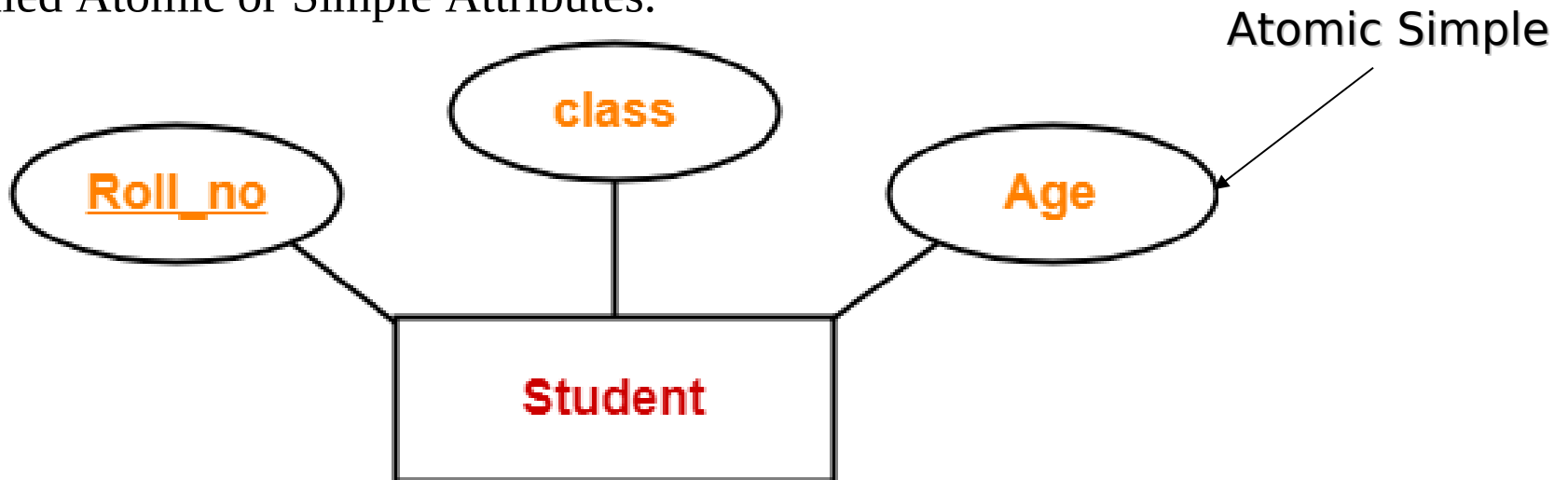
1	Ajay Patil	Pune	9800000000
2	Ram Patil	Delhi	9900000000



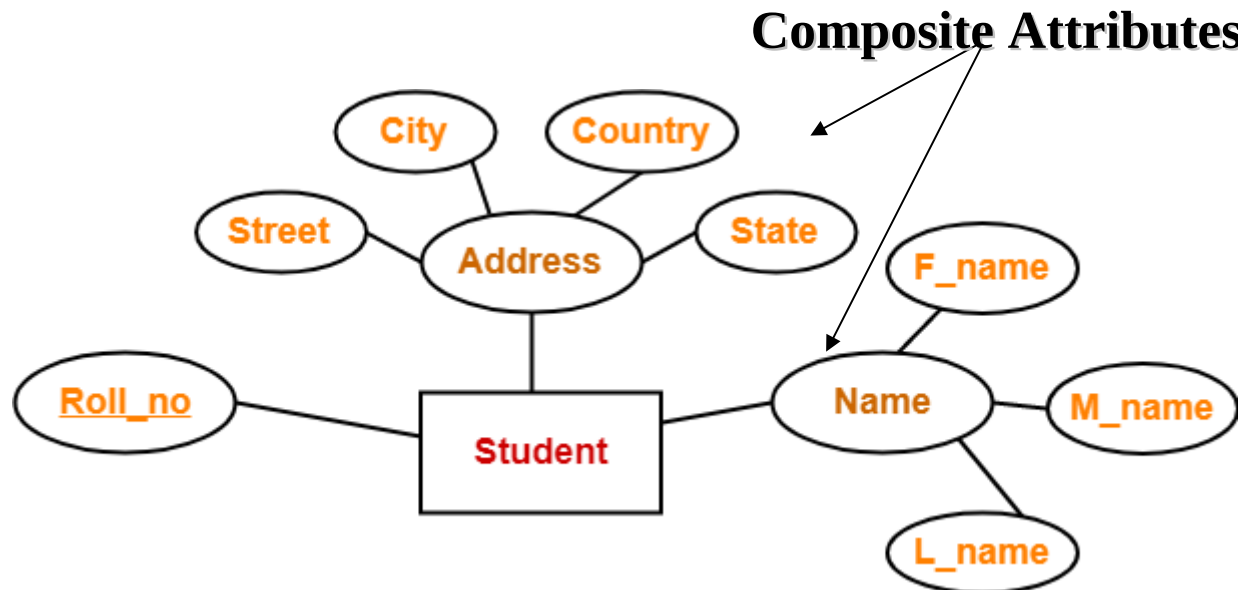
# Simple Attribute & Composite Attribute

1) **Simple Attribute** are those attribute that can not be subdivided into subparts.

It also called Atomic or Simple Attributes.



2) **Composite Attributes** are those attribute that can be subdivided into subparts.



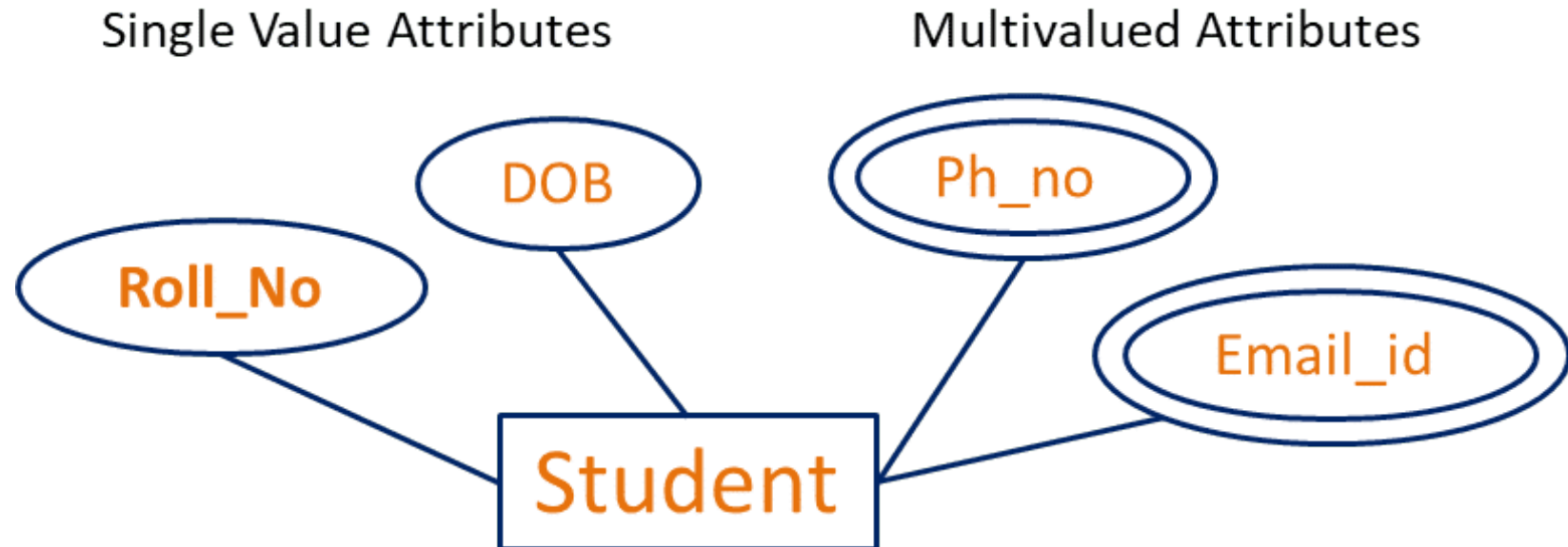


## Single-Valued & Multi Valued Attributes

**3) Single Valued Attribute** are those attribute, which can take only one value.  
which means a single valued attribute can have only a single value.

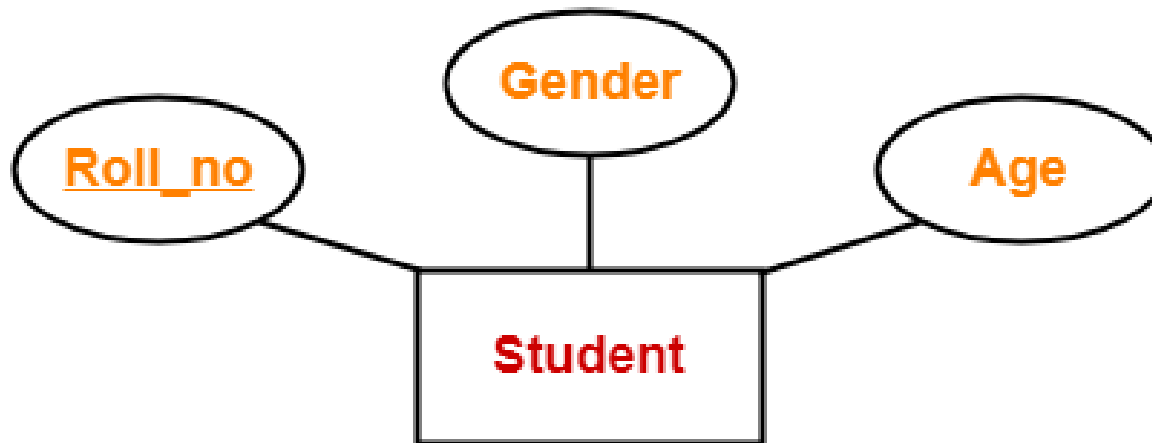
**4) Multi Valued Attribute** are those attribute, which can take more than one value.

which means a multi valued attribute can have multiple values.

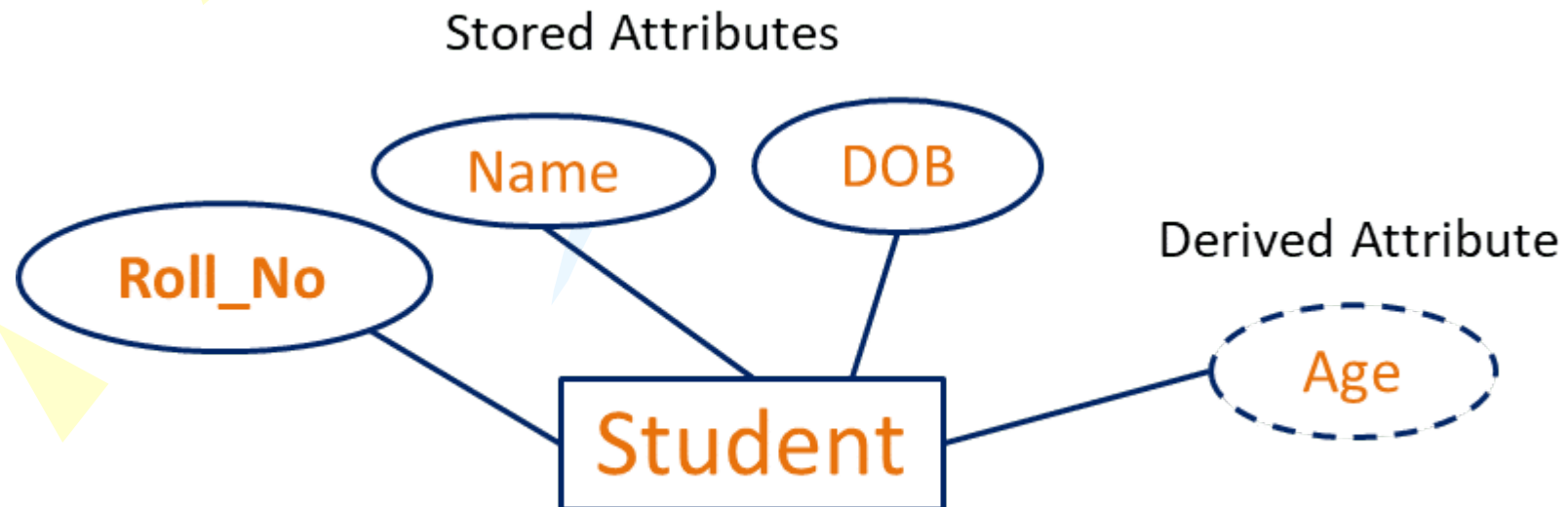


## Key Attributes and Derived Attribute

5) **Key attributes** are those attributes which can identify an entity uniquely in an entity set.



6) **Derived Attribute** The value for this type of attribute can be derived from the values of other related attributes or entities.



# Relationships

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

→ Given a name that describes its function.

- relationship name is an active or a passive verb.



Author

Relationship name:  
*writes*



Book

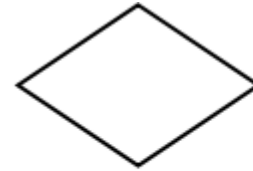
An author writes one or more books  
A book can be written by one or more authors.

# Relationships

## Strong Relationship Set-

A strong relationship exists between two strong entity sets.

It is represented using a diamond symbol.



**Strong Relationship Set**

## 2. Weak Relationship Set-

A weak or identifying relationship exists between the strong and weak entity set.

It is represented using a double diamond symbol.

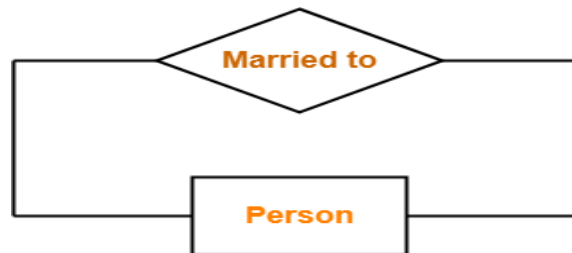


**Weak or Identifying Relationship Set**

# Degree of Relationships

**Degree of relationship set:** number of entity sets that participate in a relationship set.

**1) Unary relationship set:** Unary relationship set is a relationship set where only one entity set participates in a relationship set.



Unary Relationship Set

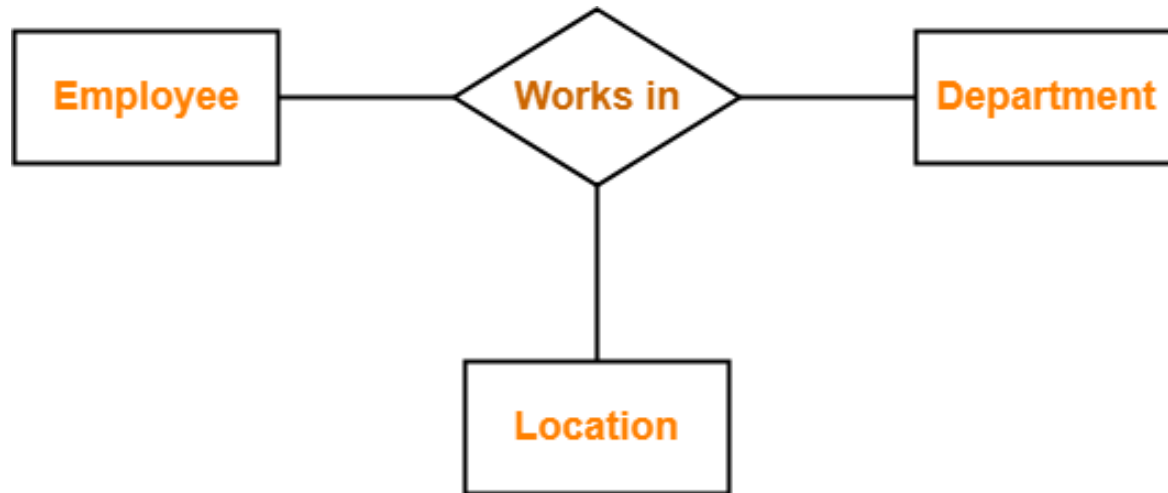
**2) Binary relationship set:** Binary relationship set is a relationship set where two entity sets participate in a relationship set.



Binary Relationship Set

# Degree of Relationships

**3) Ternary Relationship Set:** Ternary relationship set is a relationship set where three entity sets participate in a relationship set.



**Ternary Relationship Set**

**4) N-ary Relationship Set:** N-ary relationship set is a relationship set where 'n' entity sets participate in a relationship set.

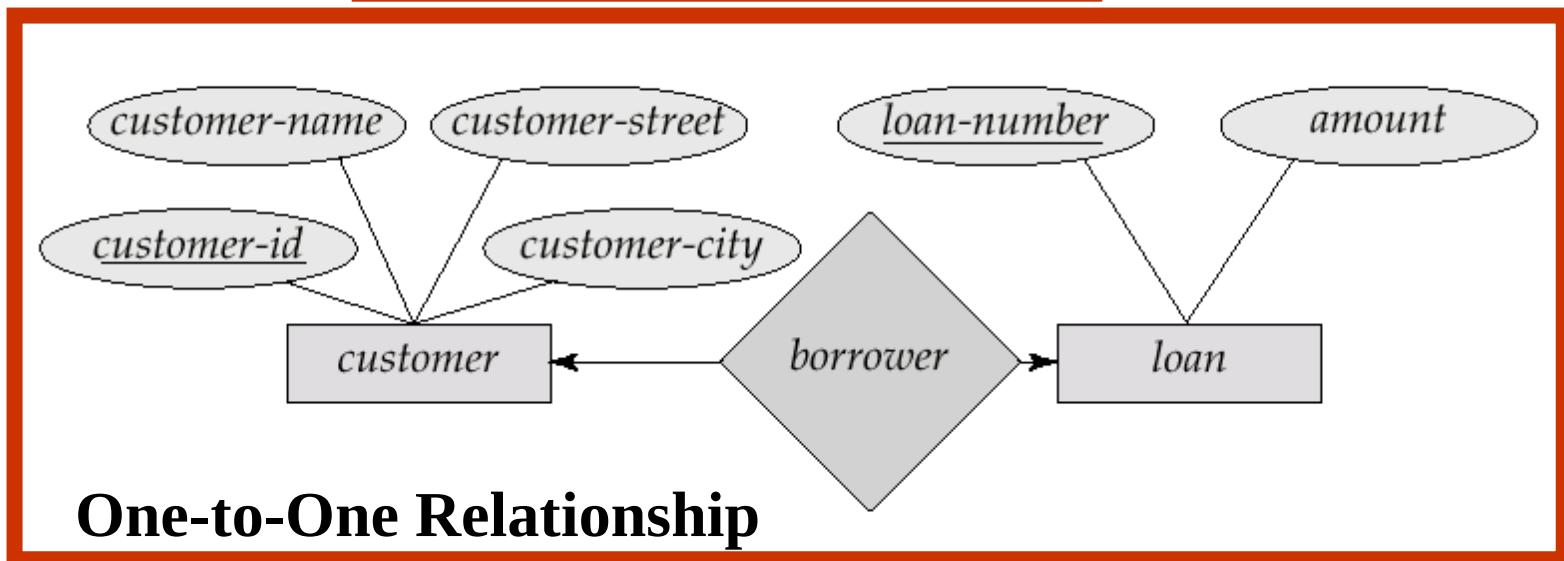
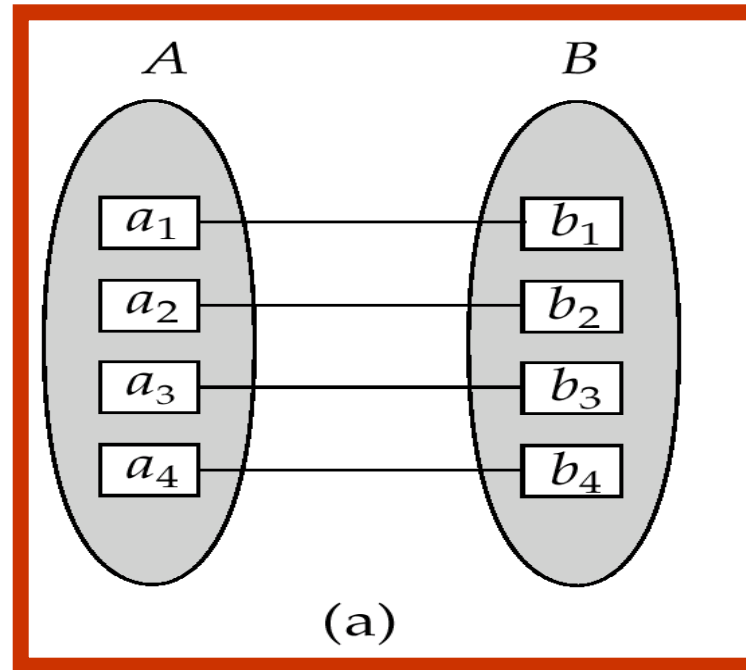
# Mapping Cardinality

**Mapping Cardinality** : express the number of entities to which another entity can be associated via a relationship set.

- Most useful in describing binary relationship sets.
- For a binary relationship set the mapping cardinality must be one of the following types:
  - 1) One to one
  - 2) One to many
  - 3) Many to one
  - 4) Many to many

## One-to-One mapping Cardinality

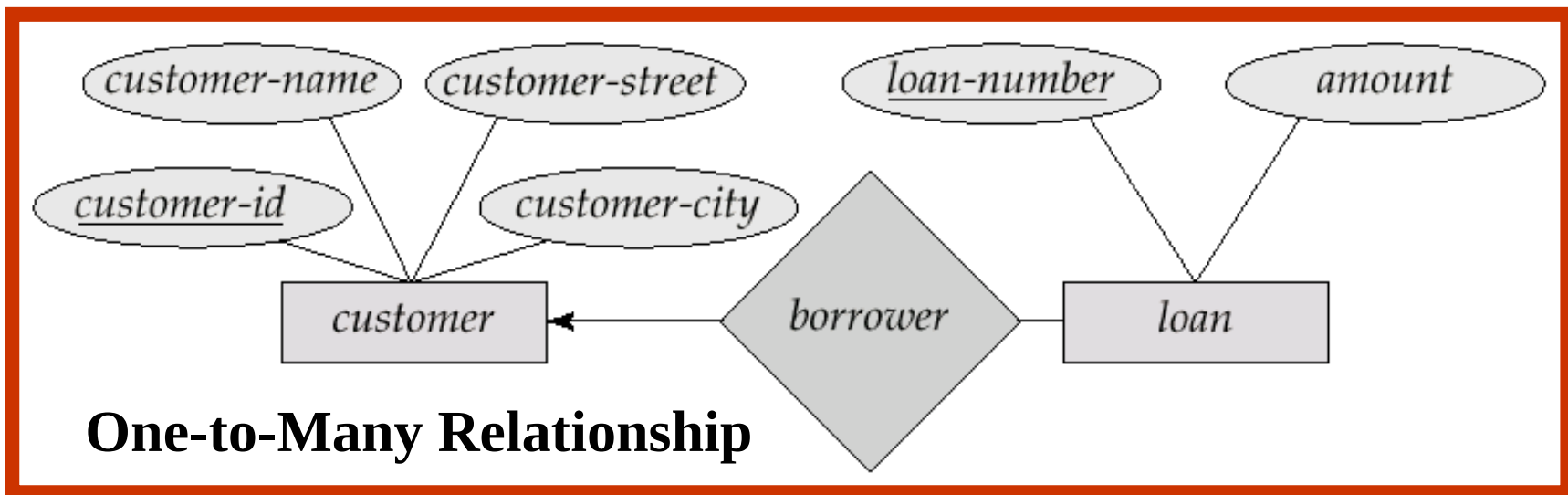
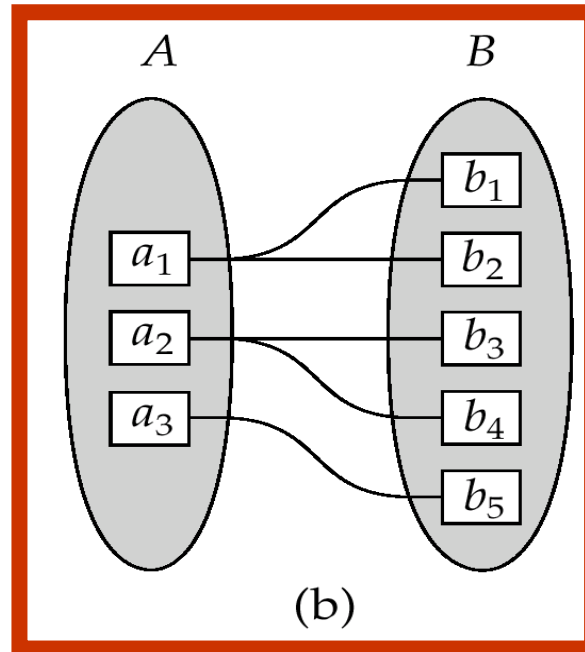
An entity in entity set  $A$  is associated with at most one entity in entity set  $B$ , and an entity in entity set  $B$  is associated with at most one entity in entity set  $A$  via relationship set  $R$





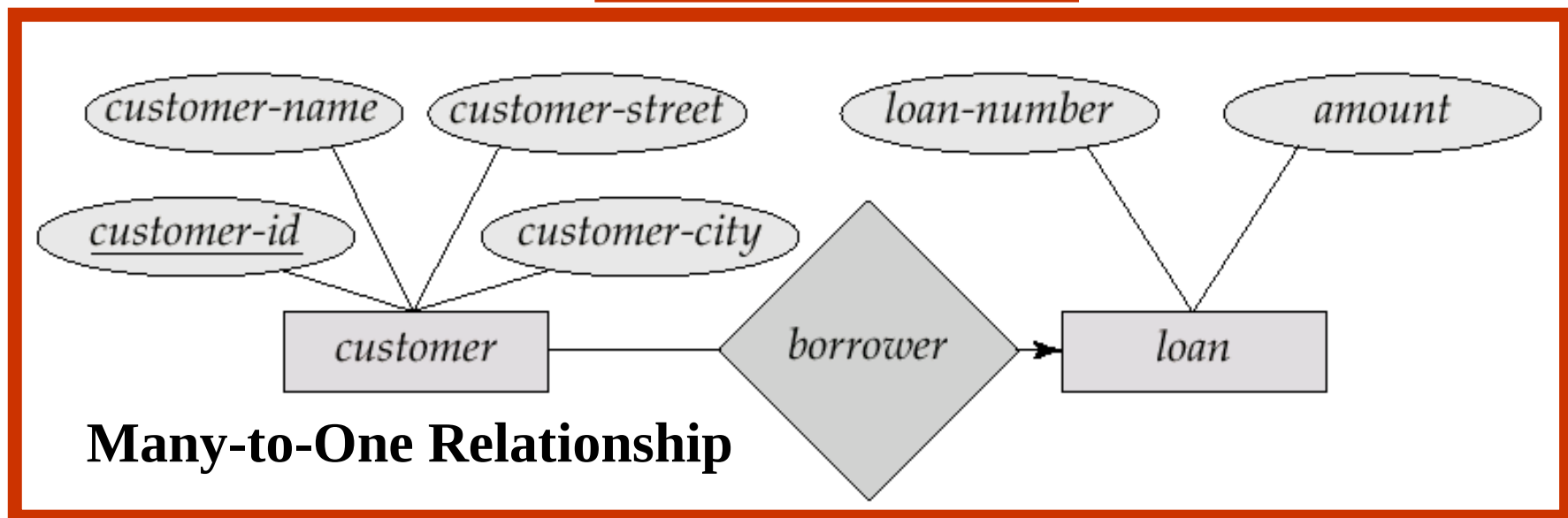
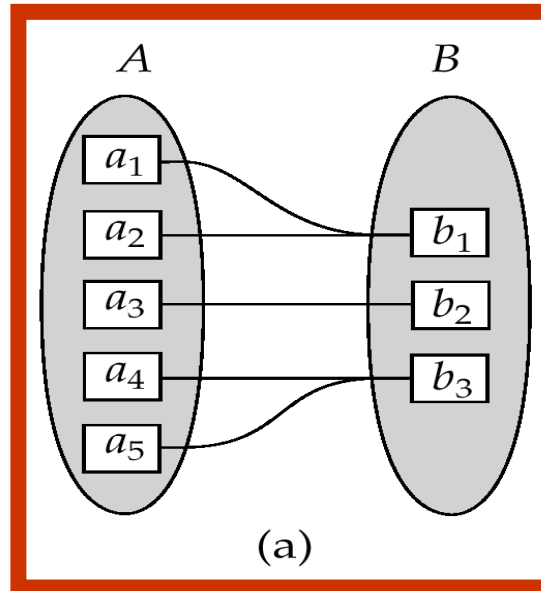
## One-to-Many mapping Cardinality

An entity in entity set A is associated with any number (zero or more) of entities in entity set B. An entity in entity set B can be associated with at most one entity in entity set A via relationship set R



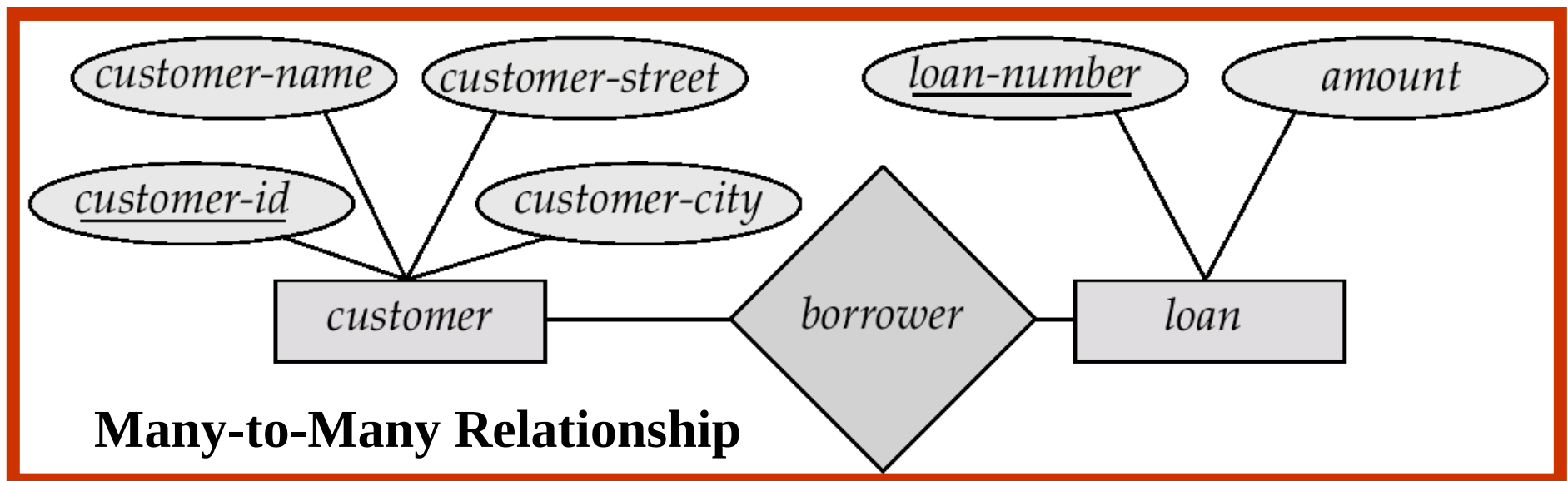
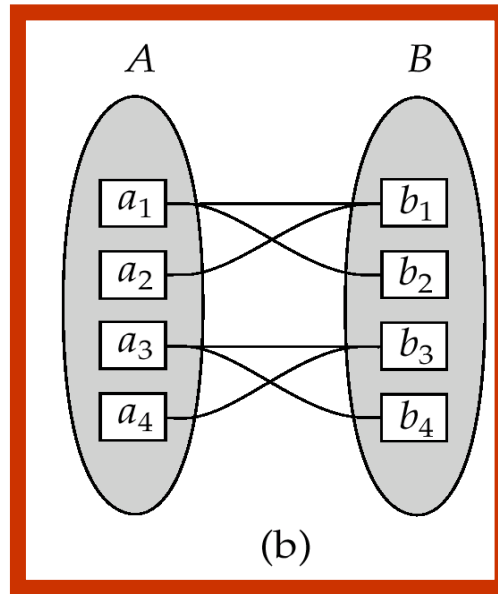
## Many-to-One mapping Cardinality

An entity in entity set A is associated with at most one entity in entity set B. An entity in entity set B can be associated with any number (zero or more) of entities in entity set A via relationship set R



## Many -to-Many mapping Cardinality

An entity in entity set A is associated with any number (zero or more) of entities in entity set B, and an entity in entity set B is associated with any number (zero or more) of entities in entity set A via relationship set R



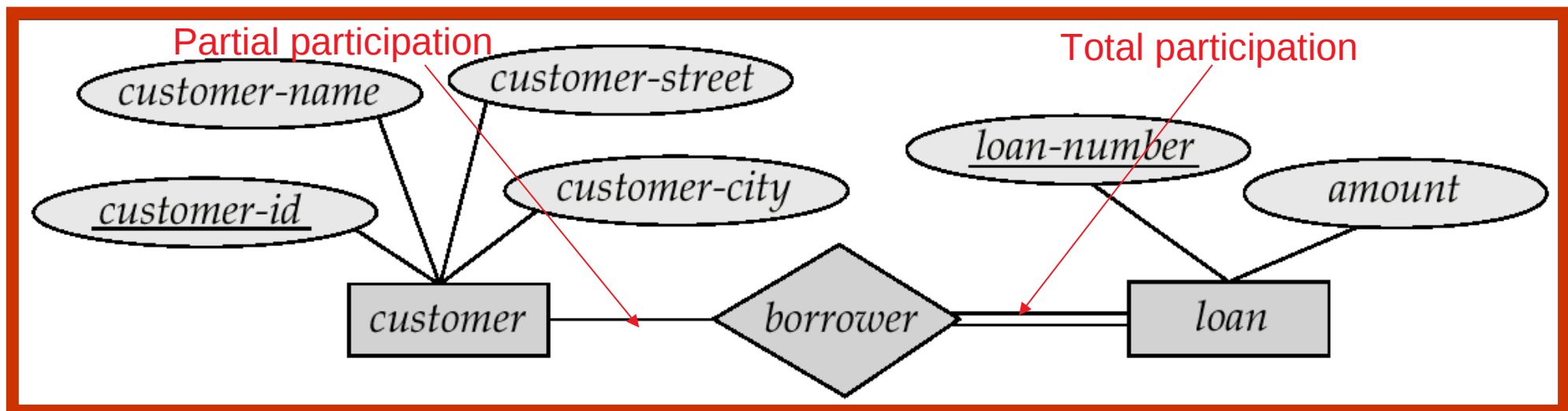
# Participation of an Entity Set in a Relationship Set

**Total participation :** every entity in the entity set participates in at least one relationship in the relationship set

E.g. participation of *loan* in *borrower* is total

**Partial participation:** some entities may not participate in any relationship in the relationship set

E.g. participation of *customer* in *borrower* is partial



# Sub Class and Super Class

## 1. Super Class

Super class is an entity type that has a relationship with one or more subtypes.  
For example: Shape super class is having sub groups as Square, Circle, Triangle.

## 2. Sub Class

Sub class is a group of entities with unique attributes.  
Sub class inherits properties and attributes from its super class.  
For example: Square, Circle, Triangle are the sub class of Shape super class.

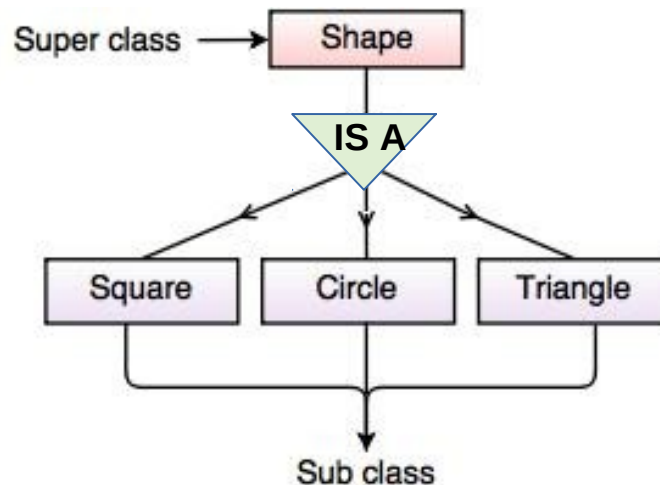


Fig. Super class/Sub class Relationship

# Specialization

- Specialization is a process that defines a group entities which is divided into sub groups based on their characteristic.
- It is a top down approach, in which one higher entity can be broken down into two lower level entity.
- It maximizes the difference between the members of an entity by identifying the unique characteristic or attributes of each member.

For example

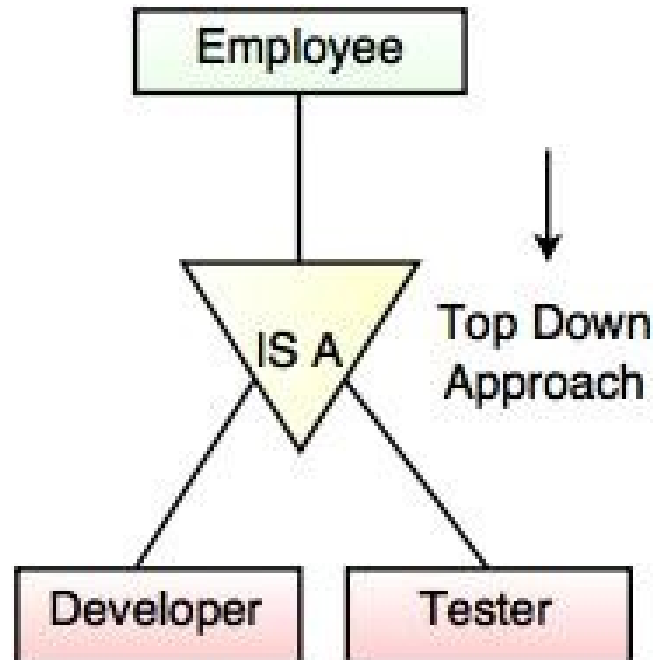


Fig. Specialization

# Generalization

- Generalization is the reverse process of Specialization.
- It is a bottom approach, in which two or more lower level entities combine to form a higher level entity.
- It defines a general entity type from a set of specialized entity type.
- It minimizes the difference between the entities by identifying the common features.

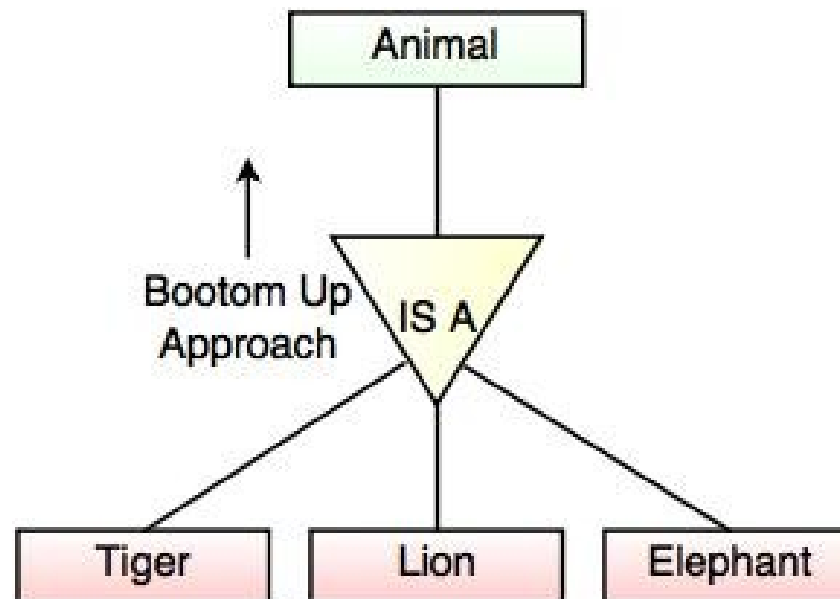
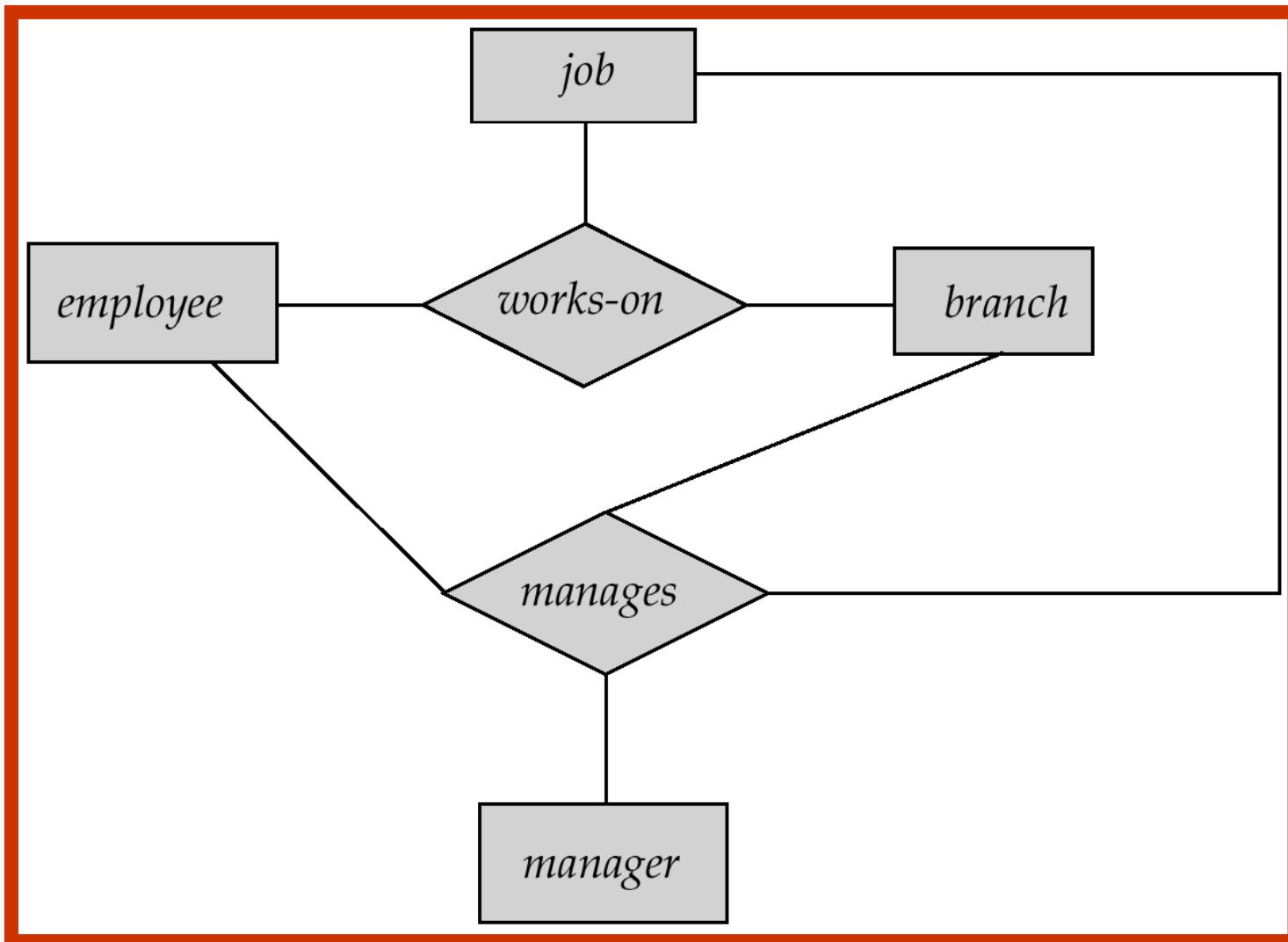


Fig. Generalization

# Aggregation





# Aggregation

Relationship sets *works-on* and *manages* represent overlapping information

- \_ Every *manages* relationship corresponds to a *works-on* relationship
- \_ However, some *works-on* relationships may not correspond to any *manages* relationships
- \_ So we can't discard the *works-on* relationship

Eliminate this redundancy via *aggregation*

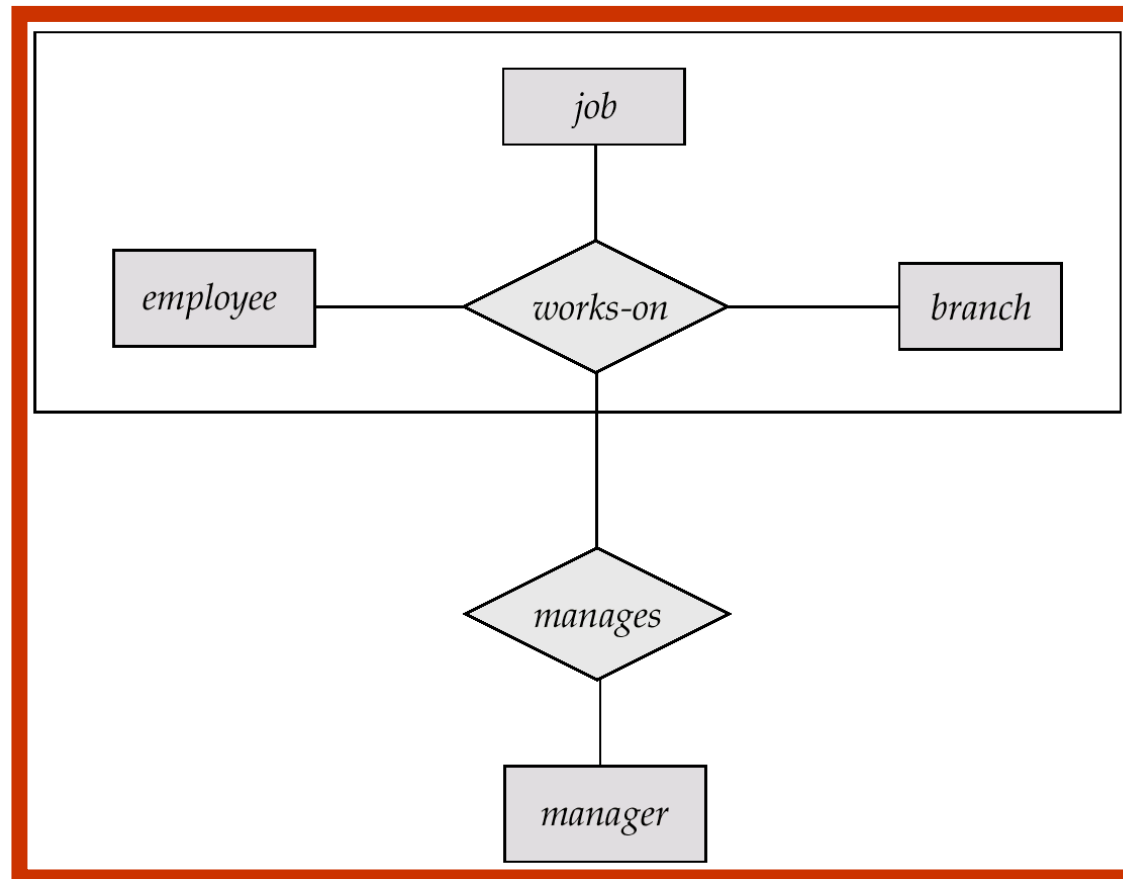
- \_ Treat relationship as an abstract entity
- \_ Allows relationships between relationships
- \_ Abstraction of relationship into new entity

Without introducing redundancy, the following diagram represents:

- \_ An employee works on a particular job at a particular branch
- \_ An employee, branch, job combination may have an associated manager

# Aggregation

Aggregation is an abstraction that treats relationships as entities. Sometimes we have to model relationship between a collection of entities and relationships.



# Keys

**A super key** of an entity set is a set of one or more attributes whose values uniquely determine each entity.

Consider the following Student schema-

**Student** ( roll , name , sex , age , address , class , section )

examples of super keys are-

( roll , name , sex , age , address , class , section )

( class , section , roll )

(class , section , roll , sex )

( name , address )

**A candidate key** of an entity set is a minimal super key.

Consider the following Student schema-

**Student** ( roll , name , sex , age , address , class , section )

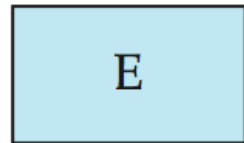
examples of candidate keys -

( class , section , roll )

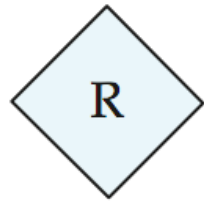
( name , address )

**A primary key** is a candidate key that the database designer selects while designing the database.

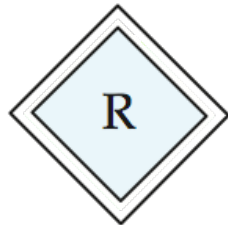
# Summary of Symbols Used in E-R Notation



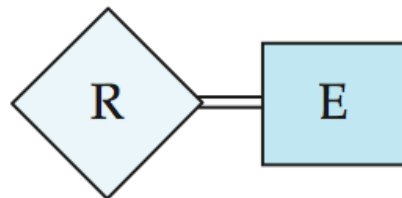
entity set



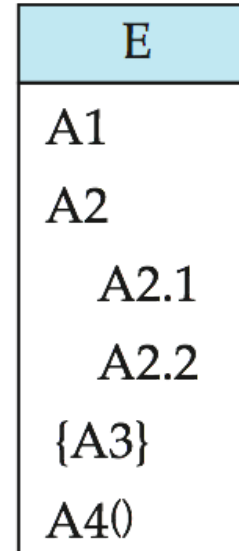
relationship set



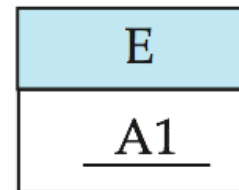
identifying  
relationship set  
for weak entity set



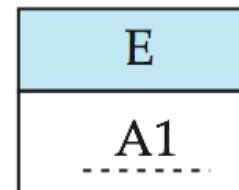
total participation  
of entity set in  
relationship



attributes:  
simple (A1),  
composite (A2) and  
multivalued (A3)  
derived (A4)

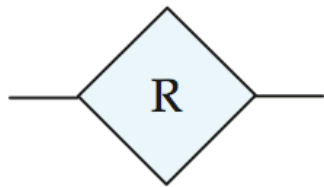


primary key

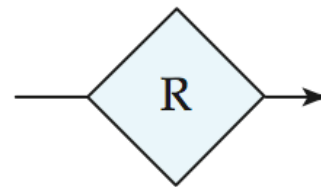


discriminating  
attribute of  
weak entity set

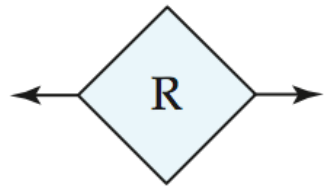
# Summary of Symbols Used in E-R Notation



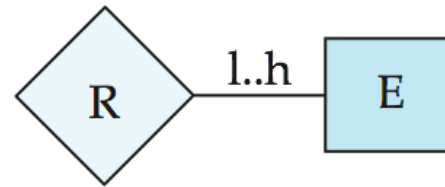
many-to-many  
relationship



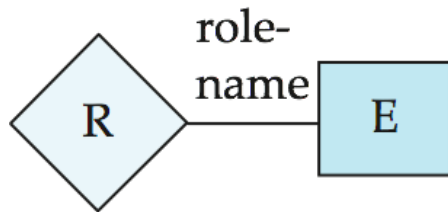
many-to-one  
relationship



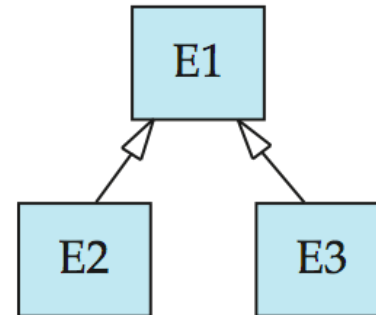
one-to-one  
relationship



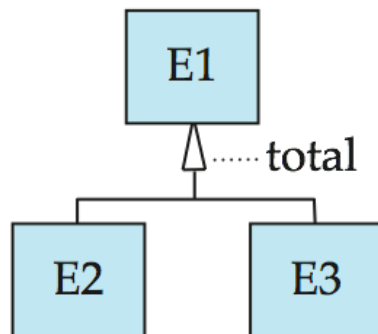
cardinality  
limits



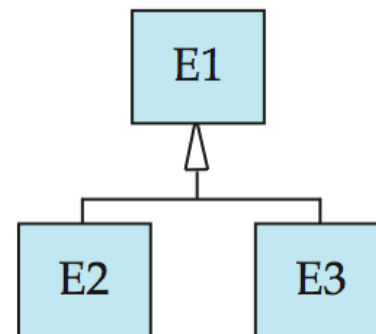
role indicator



ISA: generalization  
or specialization



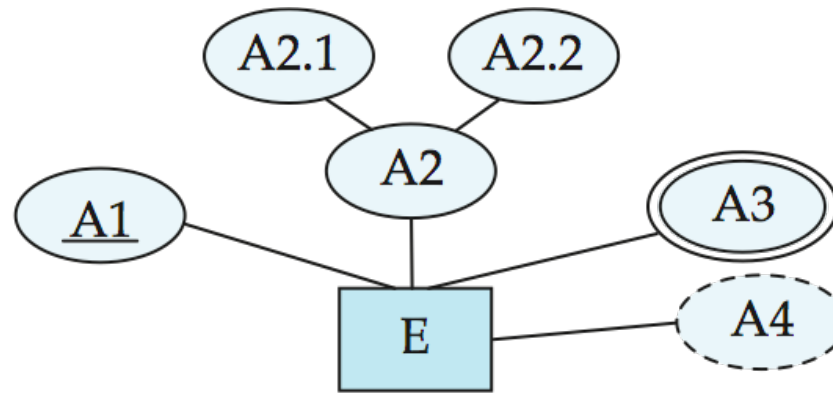
total (disjoint)  
generalization



disjoint  
generalization

# Alternative ER Notations

entity set E with  
simple attribute A1,  
composite attribute A2,  
multivalued attribute A3,  
derived attribute A4,  
and primary key A1



weak entity set



generalization



total  
generalization



# Alternative ER Notations

## Chen

## IDE1FX (Crows feet notation)

