

### PARTICIPATION CONSTRAINT

- Does every student have to take a course?
  - If so, this is a *participation constraint*: the participation of **Students** in **Enrolled** is said to be *total* (vs. *partial*).
  - Every **SID** value in **Students** table must appear in a row of the **Enrolled** table (with a non-null **SID** value!)

**Textbook Notation:** total participation represented by a thick (**bolded**) line originating from entity

26

### EXAMPLE: MANY-ONE RELATIONSHIP

**Note:** Two relationships connect the same entity sets, but are different.

27

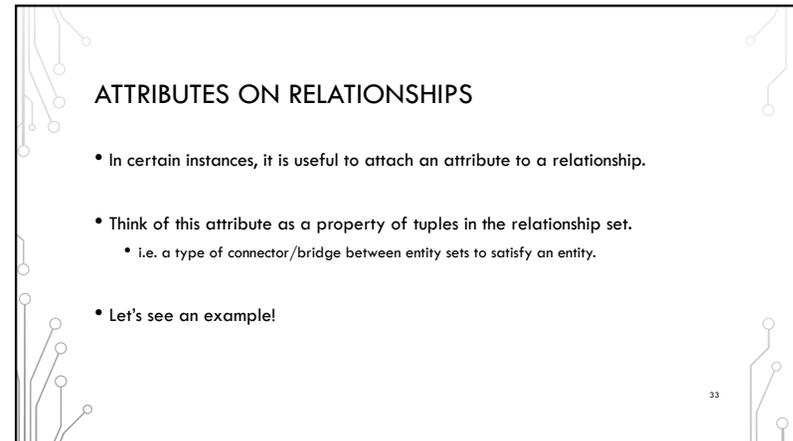
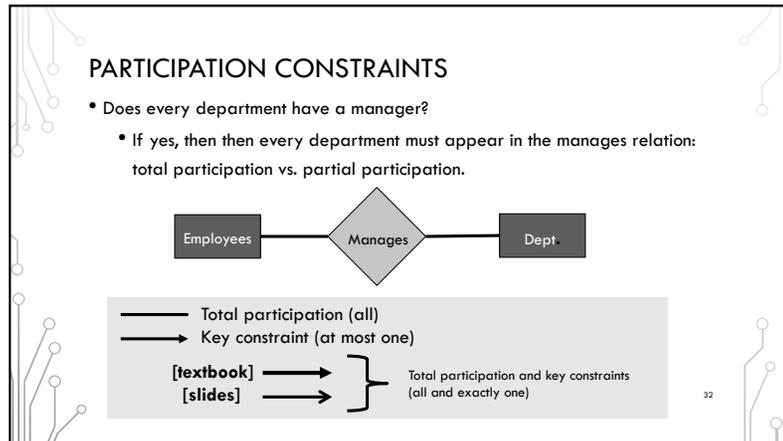
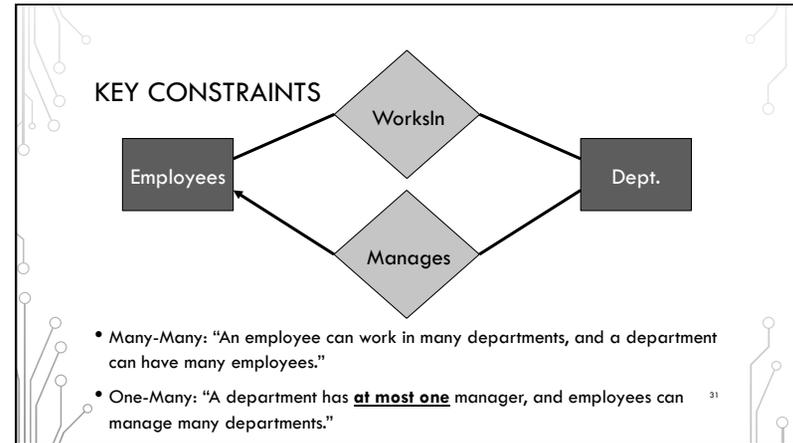
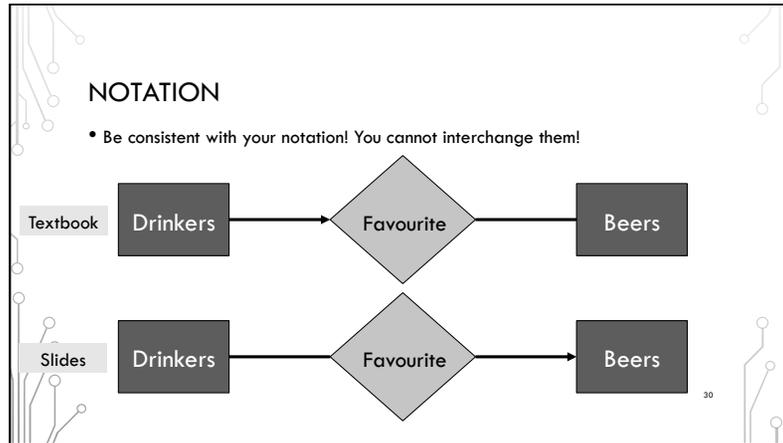
### EXAMPLE: MANY-ONE RELATIONSHIP

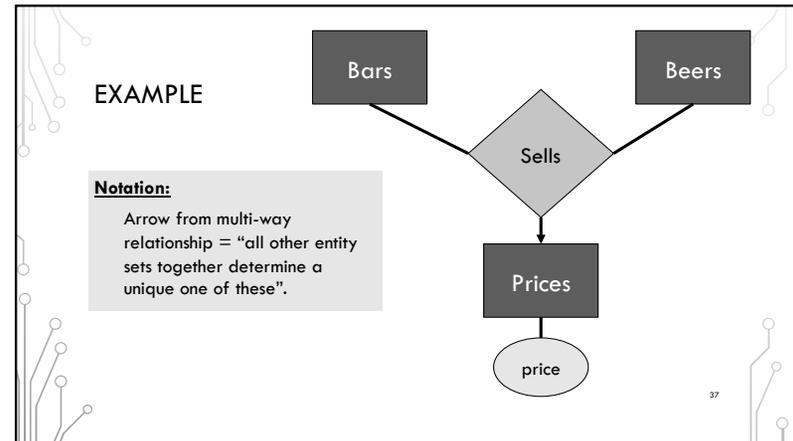
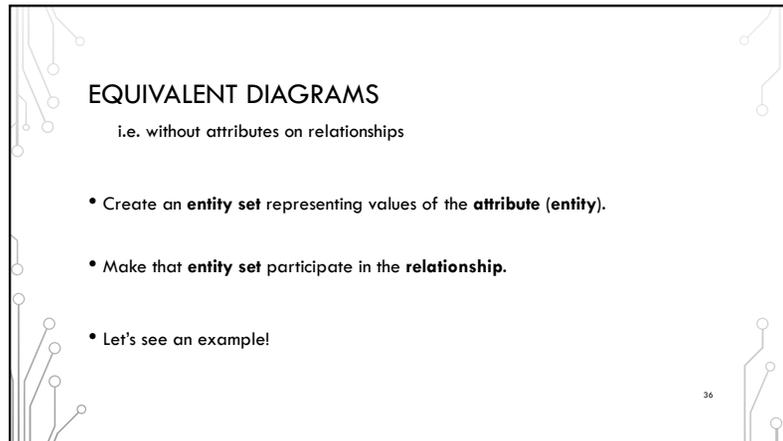
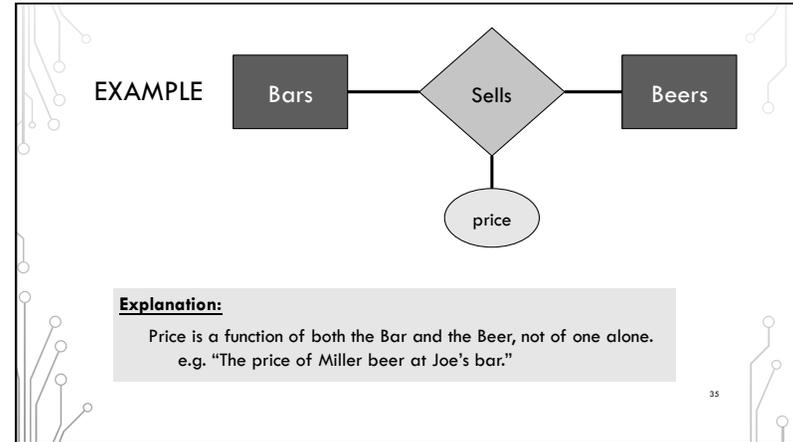
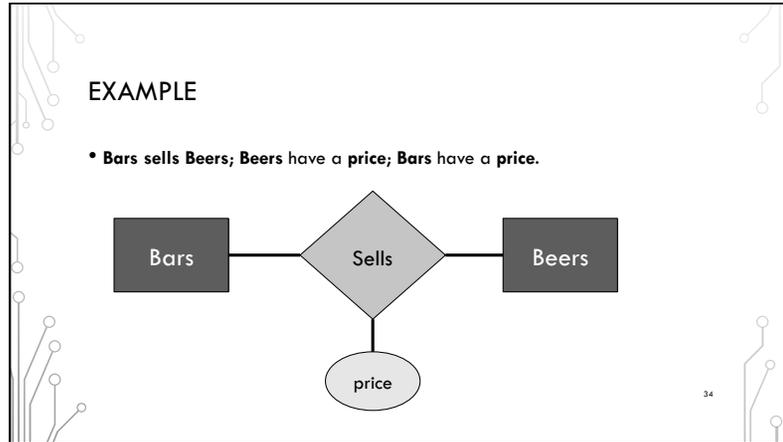
Rounded (open) Arrow = total function

28

### ALTERNATE/TEXTBOOK NOTATION

29





### ROLES

- Sometimes an entity set appears more than once in a relationship.
- Label the edges between the relationship and entity set with names called roles.

38

### EXAMPLE

Relationship Set	
Husband	Wife
John	Elizabeth
Warren	Alice
...	...

39

### EXAMPLE

Relationship Set	
Buddy1	Buddy2
Mike	Joe
Liz	Lisa
Jenny	Peter
Courtney	Moe
...	...

40

### SUBCLASSES

- Subclass = special case = more properties

e.g. Ales are a kind of beer.

- Not every beer is an ale, but some are.
- Let us suppose that in addition to all the *properties* (attributes and relationships) of beers also have the attribute **colour**.

41

### SUBCLASSES IN E/R DIAGRAMS

- **ISA** triangles indicate the subclass relationship.
  - Point to the superclass.
- Reasons for using **ISA**:
  - To add descriptive attributes specific to a subclass.
  - To identify entities that participate in a relationship.

42

### EXAMPLE

**Note:**  
Assume subclasses form a tree.

The diagram shows two entity sets, Beers and Ales, connected by an ISA triangle. Beers has two attributes: Name and Manf. Ales has one attribute: Colour.

43

### ISA ("IS A") HIERARCHIES

The diagram shows an Employees entity with attributes ssn, name, and lot. It has two subclasses: Hourly\_Emps (with attributes hourly\_wages and hours\_worked) and Contract\_Emps (with attribute contractid).

- As in C++, or other PLs, attributes are inherited.
- If we declare A **ISA** B, every A entity is also considered to be a B entity.
- Overlap constraints: Can two sub-classes contain the same entity?  
e.g. Can Mike be an Hourly\_Emps as well as a Contract\_Emps entity?
- Covering constraints: Does every Employees entity have to be an Hourly\_Emps or a Contract\_Emps entity?

44

### ISA ("IS A") HIERARCHIES

The diagram shows an Employees entity with attributes ssn, name, and lot. It has two subclasses: Hourly\_Emps (with attributes hourly\_wages and hours\_worked) and Contract\_Emps (with attribute contractid).

- Reasons for using **ISA**:
  - To add descriptive attributes specific to a subclass.
  - To identify entities that participate in a relationship.

45

### AGGREGATION

- Used when we have to model a relationship involving (entity sets and) a relationship set.
- Aggregation allows us to treat a relationship set as an entity set for the purposes of participation in (other) relationships.

46

### KEYS

- A key is a set of attributes for one entity set such that no two entities in this set agree on all the attributes of the key.
- We **must** designate a key for every entity set.

47

### EXAMPLE: MULTI-ATTRIBUTE KEY

**Note:** **hours** and **room** could also serve as a key, but we must select only one primary key (compound in this case).

48