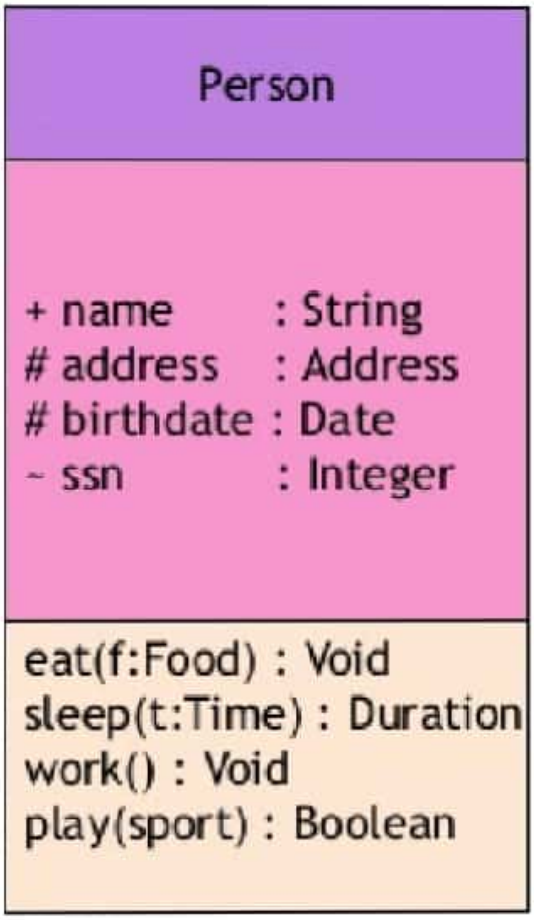
Class Diagram

- A *class* is a description of a set of objects that share the same attributes, operations, relationships, and semantics.

- Graphically, a class is rendered as a rectangle, usually including its name, attributes, and operations in separate, designated compartments.

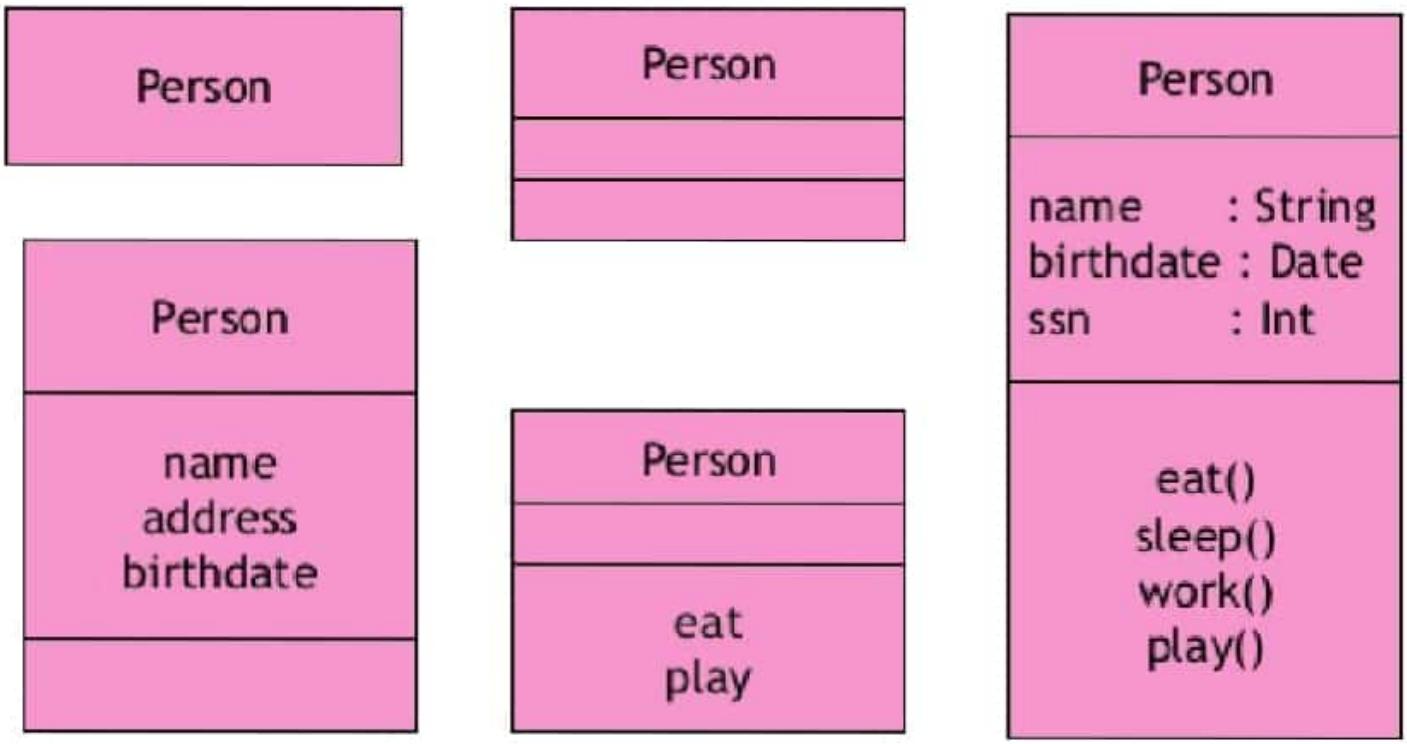
Class Diagram (cont.)

- Attributes/Operation can be:
  - \+ public
  - # protected
  - - private
  - ~ default



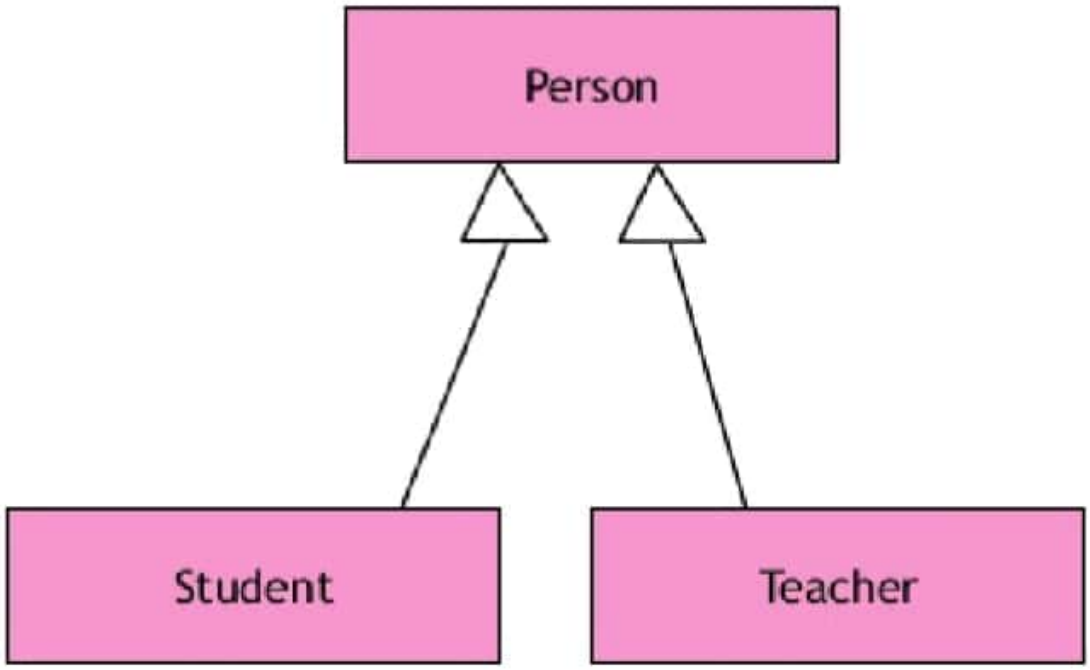
Depicting Classes

- When drawing a class, you needn't show attributes and operation in every diagram.



Generalization Relationships

- A *generalization* connects a subclass to its superclass. It denotes an inheritance of attributes and behavior from the superclass to the subclass and indicates a specialization in the subclass of the more general superclass.

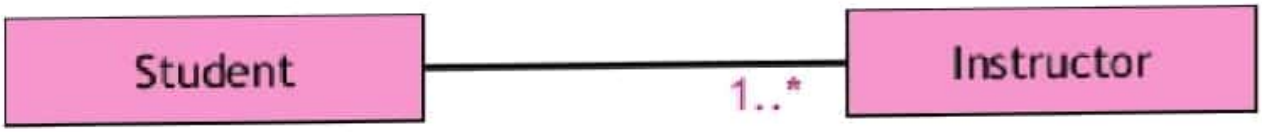


Association Relationships

- If two classes in a model need to communicate with each other, there must be link between them.
- An *association* denotes that link.

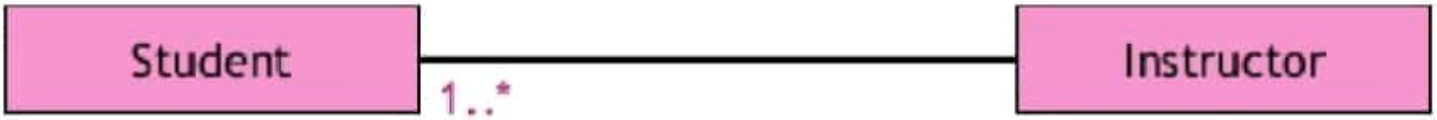
Association Relationships (Cont.)

- We can indicate the *multiplicity* of an association by adding *multiplicity adornments* to the line denoting the association.
- The example indicates that a *Student* has one or more *Instructors*:

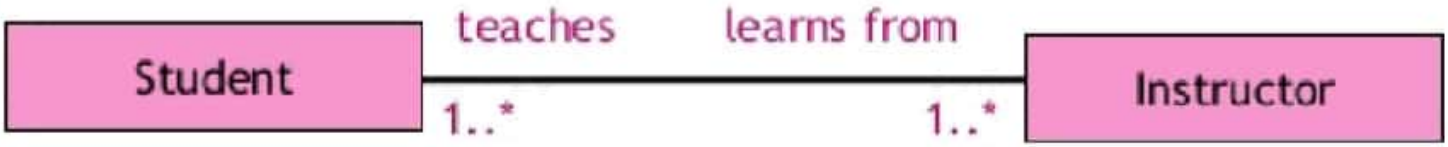


Association Relationships (Cont.)

- The example indicates that every *Instructor* has one or more *Students*:



- We can also indicate the behavior of an object in an association (*i.e.*, the *role* of an object) using *rolenames*.



- We can also name the association.

Association Relationships (Cont.)

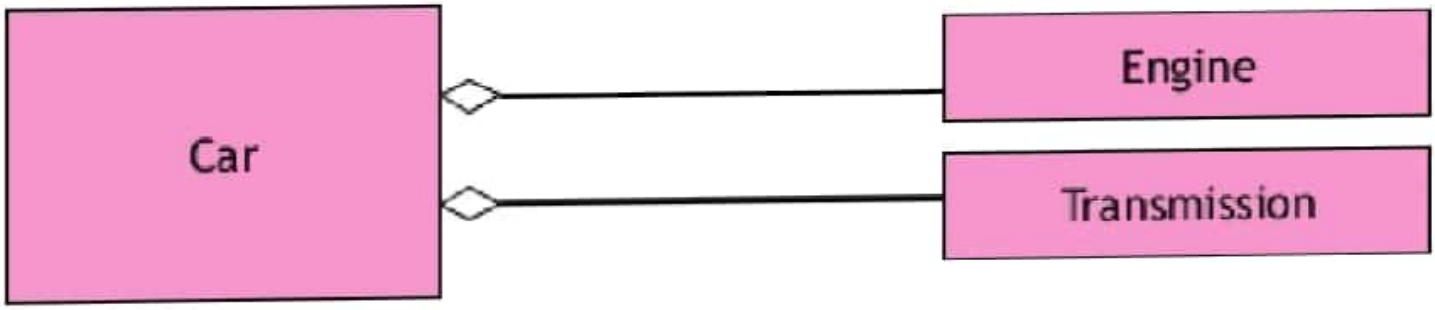
- We can constrain the association relationship by defining the *navigability* of the association. Here, a *Router* object requests services from a *DNS* object by sending messages to (invoking the operations of) the server. The direction of the association indicates that the server has no knowledge of the *Router*.

Association Relationships:

Aggregations and Compositions

- We can model objects that contain other objects by way of special associations called *aggregations* and *compositions*.

- An *aggregation* specifies a whole-part relationship between an aggregate (a whole) and a constituent part, where the part can exist independently from the aggregate. Aggregations are denoted by a hollow-diamond adornment on the association.



Association Relationships:

Aggregations and Compositions (cont.)

- A *composition* indicates a strong ownership and coincident lifetime of parts by the whole (*i.e.*, they live and die as a whole). Compositions are denoted by a filled-diamond adornment on the association.