INHERITANCE

Generalization

“IS A”
• “Is A” that reads in the arrow direction
• A sedan inherits all of the features (member variables and functions) that a car has

• “A sedan is a car”
• “A sedan is a kind of car”
• “A sedan is like a car”
CLASS ROLES

- Parent / Child
- Superclass / Subclass
- Base class / Derived class
- Ancestor / Descendant

- The arrow head always attaches to the superclass
DIRECTIONALITY

- **Unidirectional**
  - From subclass to super class
  - Subclass can send messages to the superclass
  - Subclass “knows about” superclass
  - Can navigate from the subclass to the superclass
STRONG BINDING

LIFETIME
• Coincident or simultaneous
  • The related objects are created and are destroyed at the same time
  • When an instance of the subclass is created, an instance of the superclass is created at the same time
  • The superclass object may be initialized by the subclass constructor calling the superclass constructor

SHARING
• Exclusive
  • The subclass object does not share its superclass object with other objects
IMPLEMENTATION

• Dedicated syntax
• class sedan : public car {...}
• sedan(x, y, z) : car(x, y) {}
• inheritance must be the first element in the initializer list

IDENTIFYING CODE PATTERN

class Super
{
};

class Sub : public Super
{
};
• The subclass object has a superclass object embedded inside it

• The embedding makes
  • the binding strong
  • changing the relationship difficult
  • sharing the superclass object difficult

• as depicted, the instance of the superclass is at the beginning of the subclass object