


The OO Solution

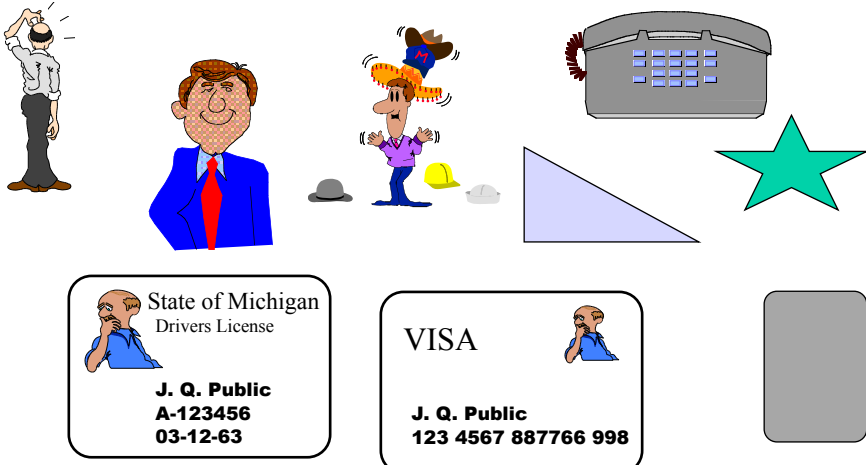
- The OO model closely resembles the problem domain
 - Base your model on the objects in the problem domain
- Iteratively refine the high-level model until you have an implementation
 - Attempt to avoid big conceptual jumps during the development process

CSE870: UML Classes

1

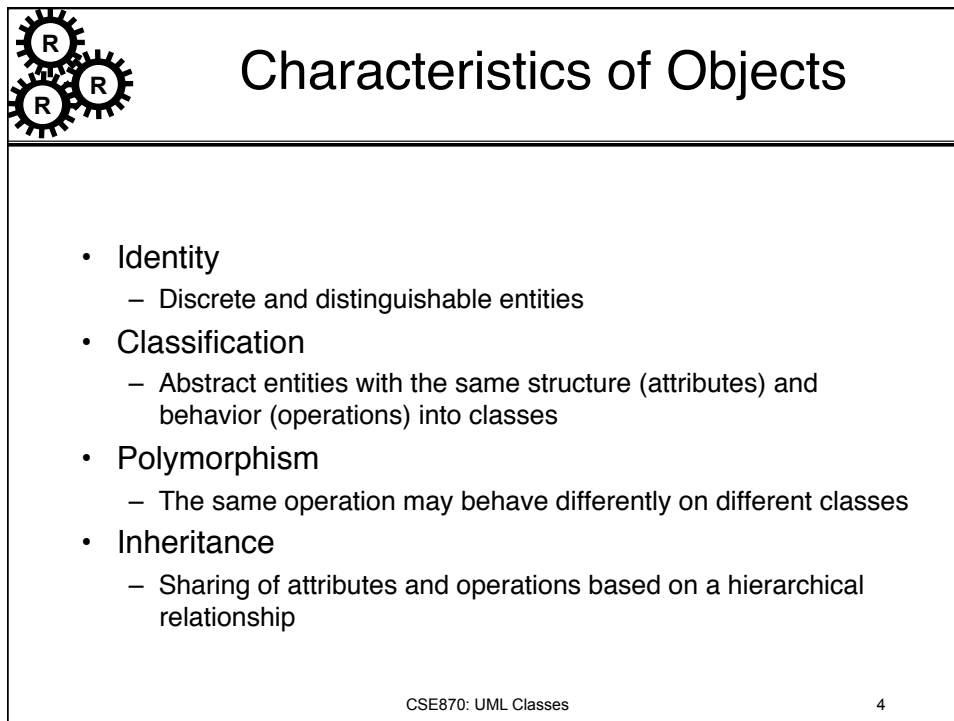
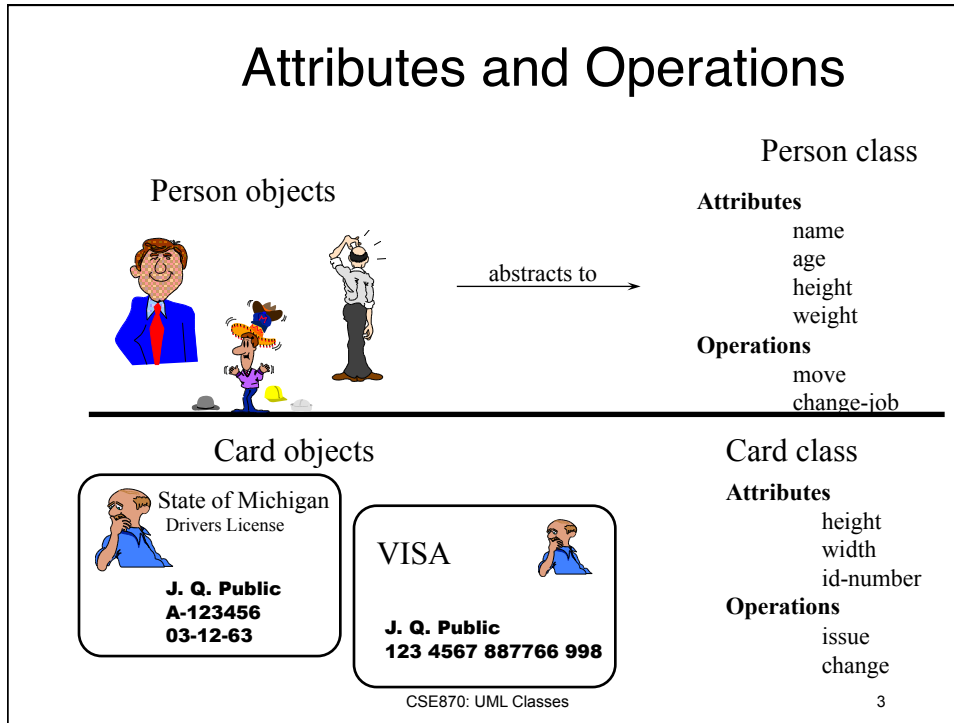


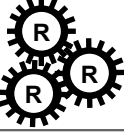
Objects



CSE870: UML Classes

2






The Class Diagrams

CSE870: UML Classes

5




Objects

- Something that makes sense in the application context (application domain)
 - J.Q. Public
 - Joe's Homework Assignment 1
 - J. Q. Public's drivers license
- All objects have identity and are distinguishable
- NOT objects
 - Person
 - Drivers license

CSE870: UML Classes


6



Classes

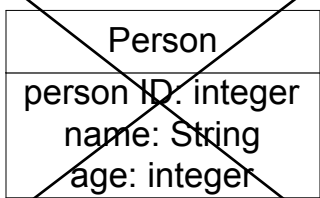
- Describes a group of objects with:
 - similar properties (attributes),
 - common behavior (operations),
 - common relationships to other classes, and
 - common semantics
- Person
 - J. Q. Public
 - Joe Smith
 - D. Q. Public
 - Card
 - Credit card
 - Drivers license
 - Teller card

CSE870: UML Classes 7



Class Diagrams

Class diagram	Instance diagram	
<div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;">Person</div> <div style="border: 1px solid black; padding: 5px;">age: integer</div>	<div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;">D. Q. Public:Person</div> <div style="border: 1px solid black; padding: 5px;">age= 32</div>	<div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;">J. Q. Public: Person</div> <div style="border: 1px solid black; padding: 5px;">age= 35</div>
Class with attributes	Objects with values	

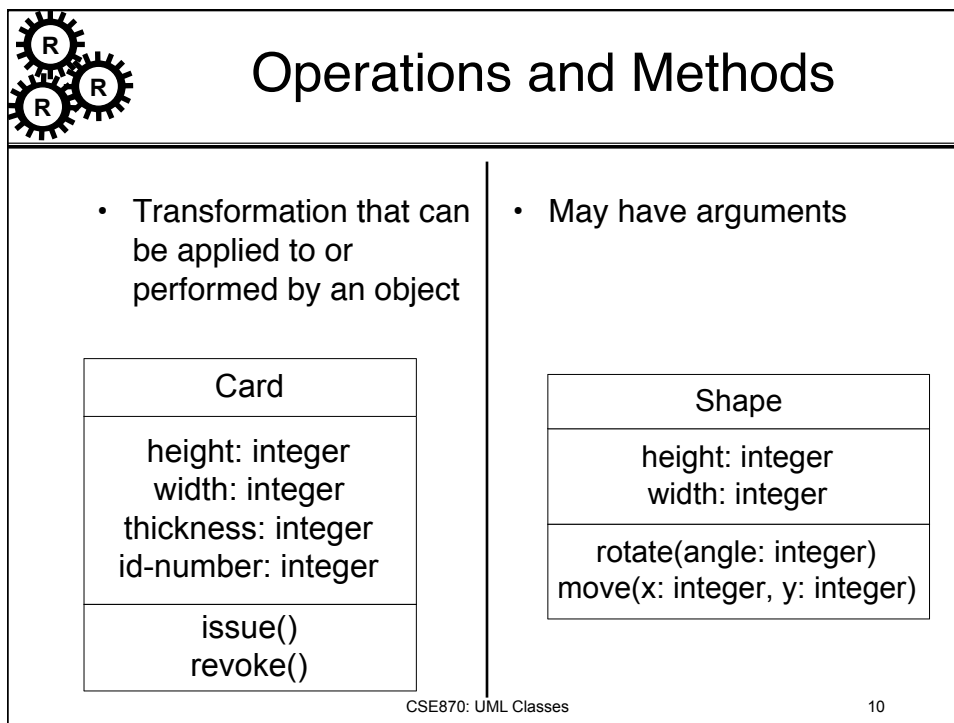
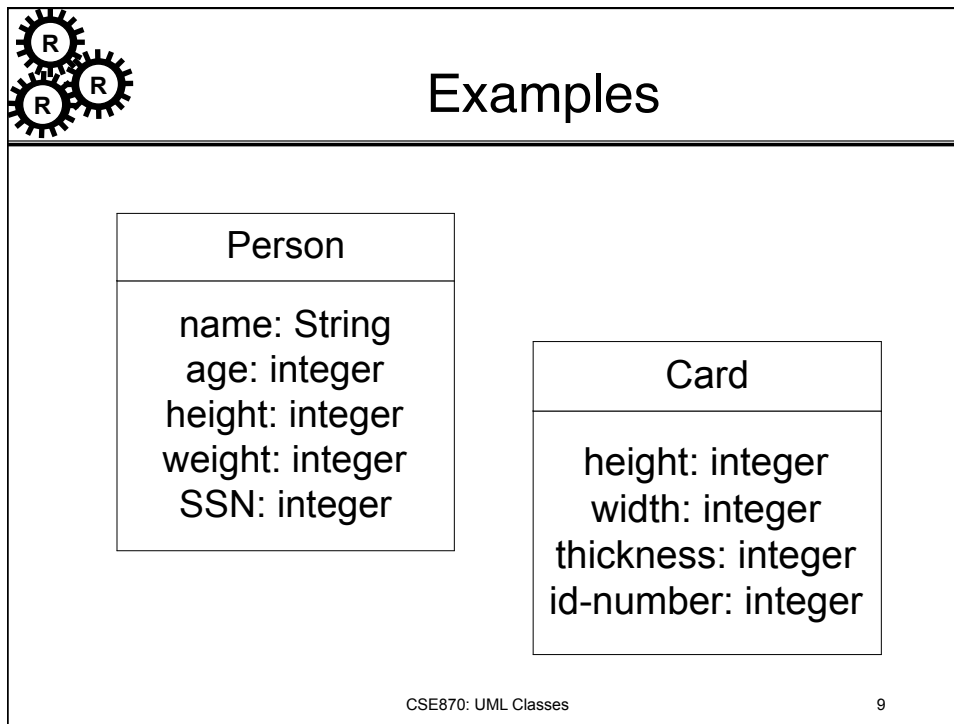


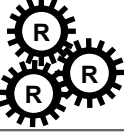
Objects have an identity

Do not explicitly list object identifiers

SSN OK!

CSE870: UML Classes 8






Object Notation - Summary

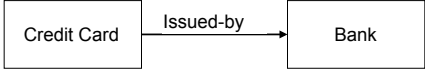
Class name
attribute-1 : data-type-1 = default-value-1 attribute-2 : data-type-2 = default-value-2 attribute-3 : data-type-3 = default-value-3
operation-1(argument-list-1) : result-type-1 operation-2(argument-list-2) : result-type-2 operation-3(argument-list-3) : result-type-3

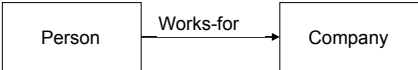
CSE870: UML Classes 11



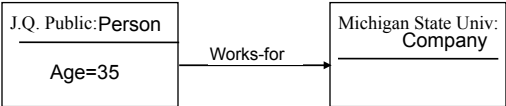
Associations

- Conceptual connection between classes
 - A credit card is issued-by a bank
 - A person works-for a company



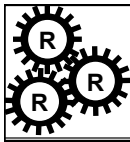


Class diagrams



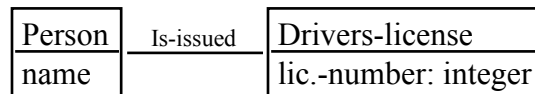
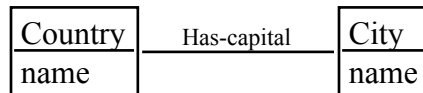
Instance diagram

CSE870: UML Classes 12



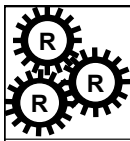
Associations are Bi-directional

- There is no direction implied in an association (Rumbaugh - OMT)



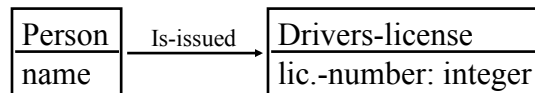
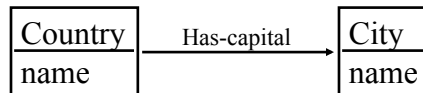
CSE870: UML Classes

13



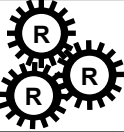
Associations Can Have Direction

- Unified adds a direction indicator
 - Inconsistently used



CSE870: UML Classes

14



Multiplicity

One person holds one credit card

```

classDiagram
    class Person {
        name: String
    }
    class Credit-card {
        card-number: integer
    }
    Person --> Credit-card : Holds
    
```


- One object can be related to many objects through the same association

One person can hold zero or more credit cards

```

classDiagram
    class Person {
        name: String
    }
    class Credit-card {
        card-number: integer
    }
    Person --> "0..*" Credit-card : Holds
    
```

CSE870: UML Classes 15



Multiplicity (Cont.)

- One person can hold zero or more credit cards (0..*)
- Each card has zero or one holder (0..1)


```

classDiagram
    class Person {
        name: String
        age: integer
    }
    class Credit-card {
        card-number: integer
    }
    Person --> "0..*" Credit-card : Holds
    Credit-card --> "0..1" Person : Holds
    
```

```

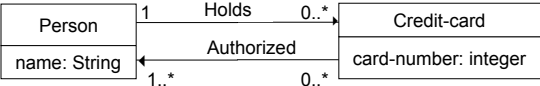
classDiagram
    class JQPublicPerson {
        name= J. Q. Public
        age=35
    }
    class DQPublicPerson {
        name= D. Q. Public
        age=32
    }
    class Card789CreditCard {
        card-number= 123 456 789
    }
    class Card123CreditCard {
        card-number= 111 222 333
    }
    class Card456CreditCard {
        card-number= 444 555 666
    }
    JQPublicPerson --> Card789CreditCard : Holds
    JQPublicPerson --> Card123CreditCard : Holds
    
```

CSE870: UML Classes 16

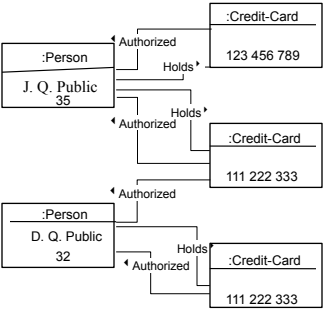


Multiplicity (Cont.)


- One person can hold zero or more credit cards (0..*)
- Each card has one holder (no indication or 1)
- Each card has one or more authorized users (1..*)
- One person can be authorized to use zero or more cards



Explicit enumeration is also possible (2, 3, 2..5, etc.)

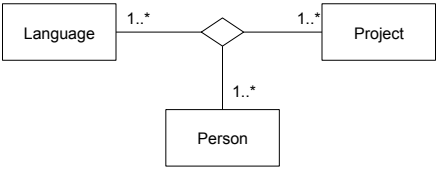


CSE870: UML Classes
17

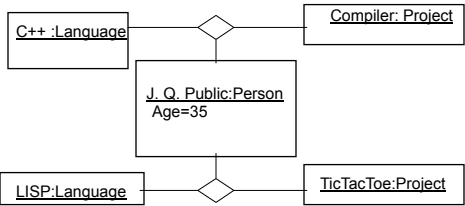


Higher order associations

- Ternary association
 - Project, language, person
- Seldom needed (and should be avoided)



Instance of ternary association



CSE870: UML Classes
18

Link Attributes

- Associations can have properties the same way objects have properties

Person

name: String
age: integer
SSN: integer
address: String

0..* Works-for →

Company

name: String
address: String

How to represent salary and job title?

Person

name: String
age: integer
SSN: integer
address: String

0..* Works-for →

Company

name: String
address: String

salary: integer
job-title: String

Use a link attribute!

CSE870: UML Classes 19

Folding Link Attributes

Person

name: String
age: integer
SSN: integer
address: String
salary: integer
job-title: String

0..* Works-for →

Company

name: String
address: String

Why not this?

Salary and job title are properties of the job **not** the person

Person

name: String
age: integer
SSN: integer
address: String

0..* Works-for →

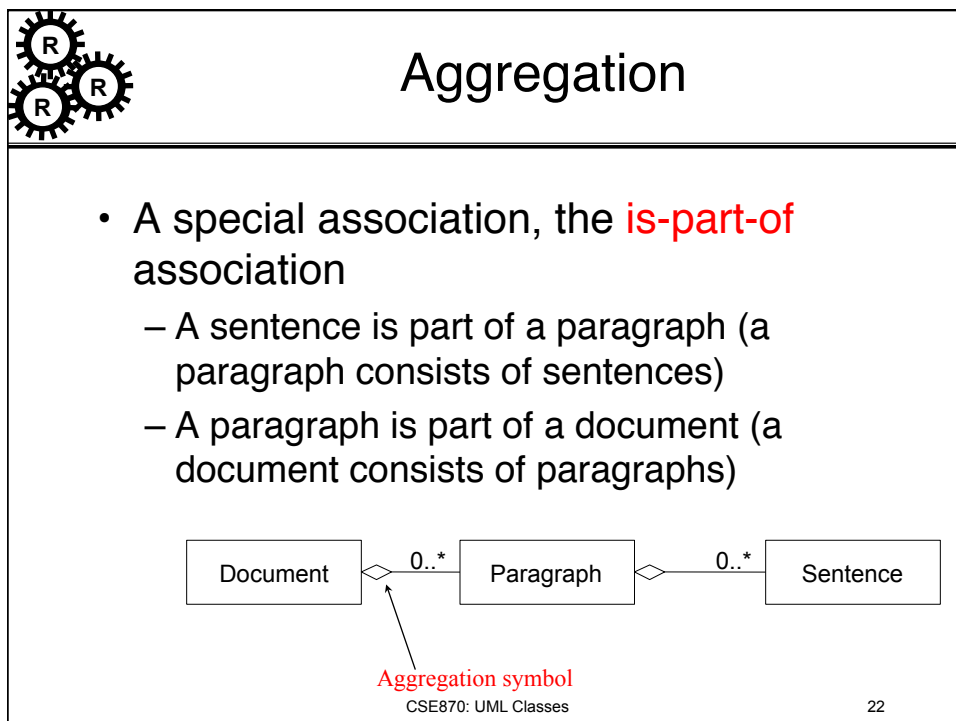
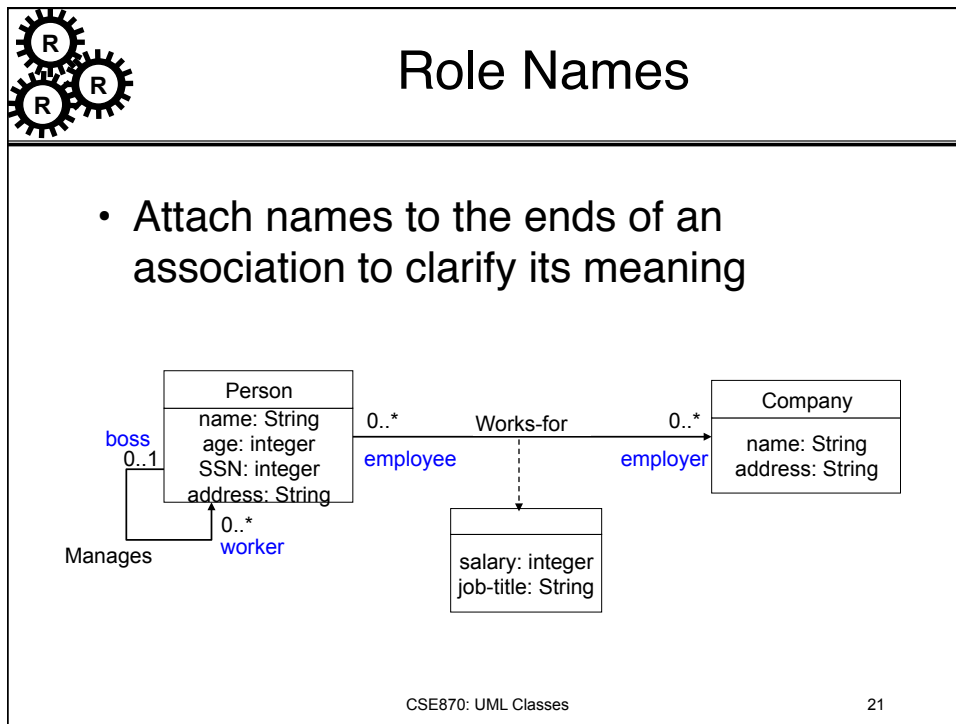
Company

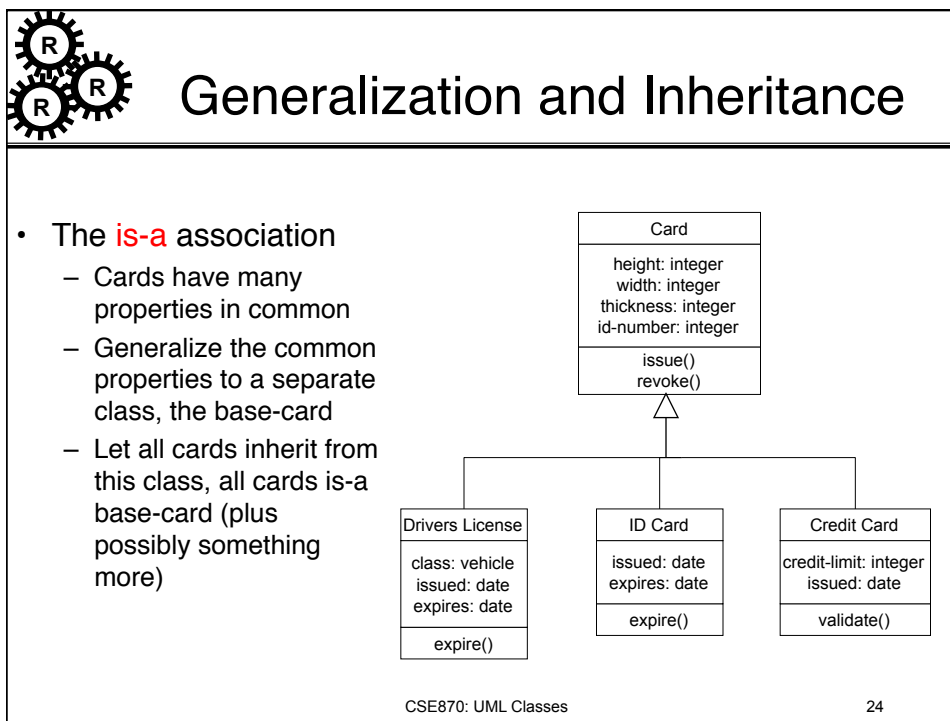
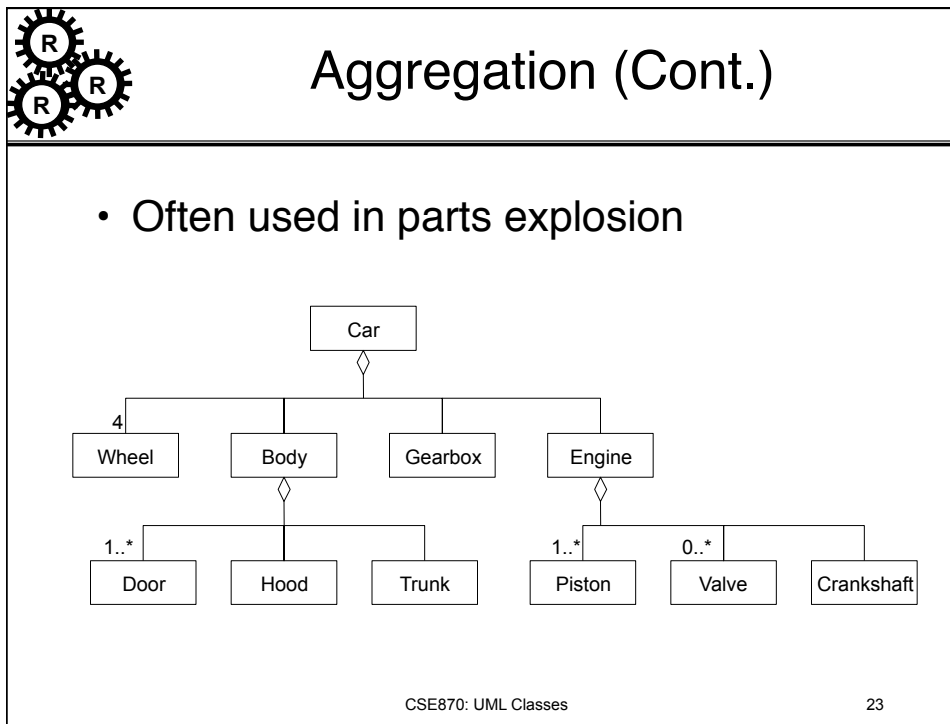
name: String
address: String

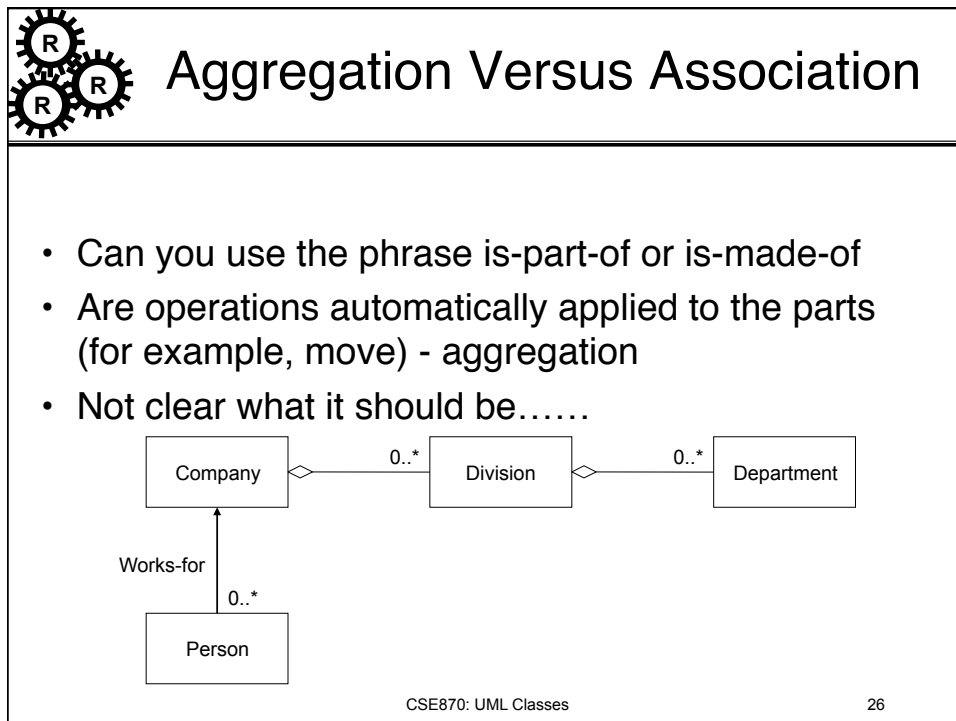
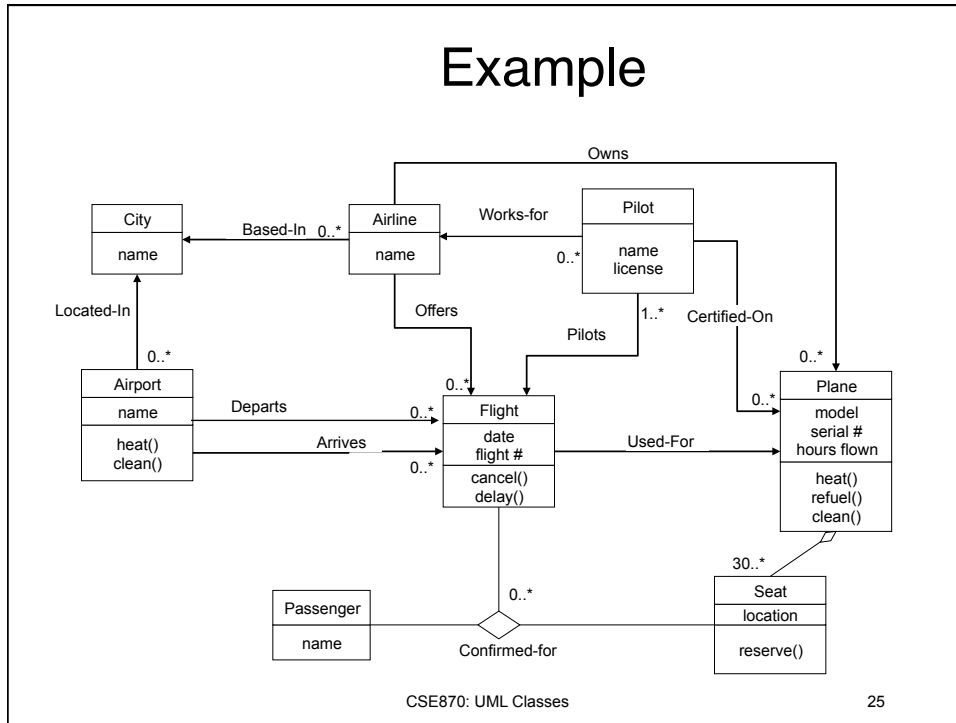
salary: integer
job-title: String

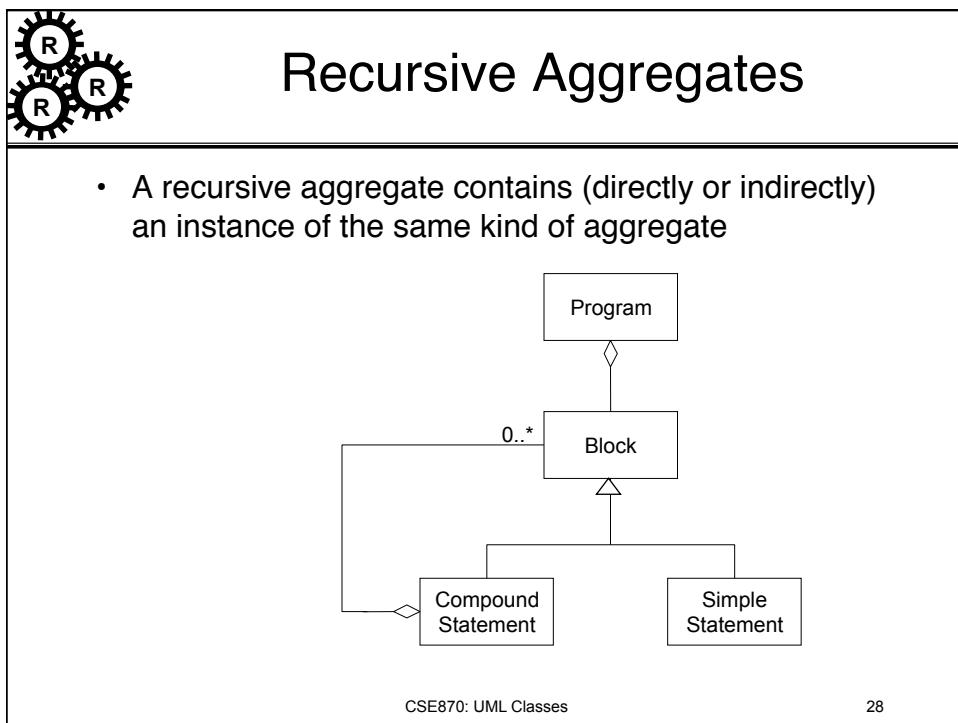
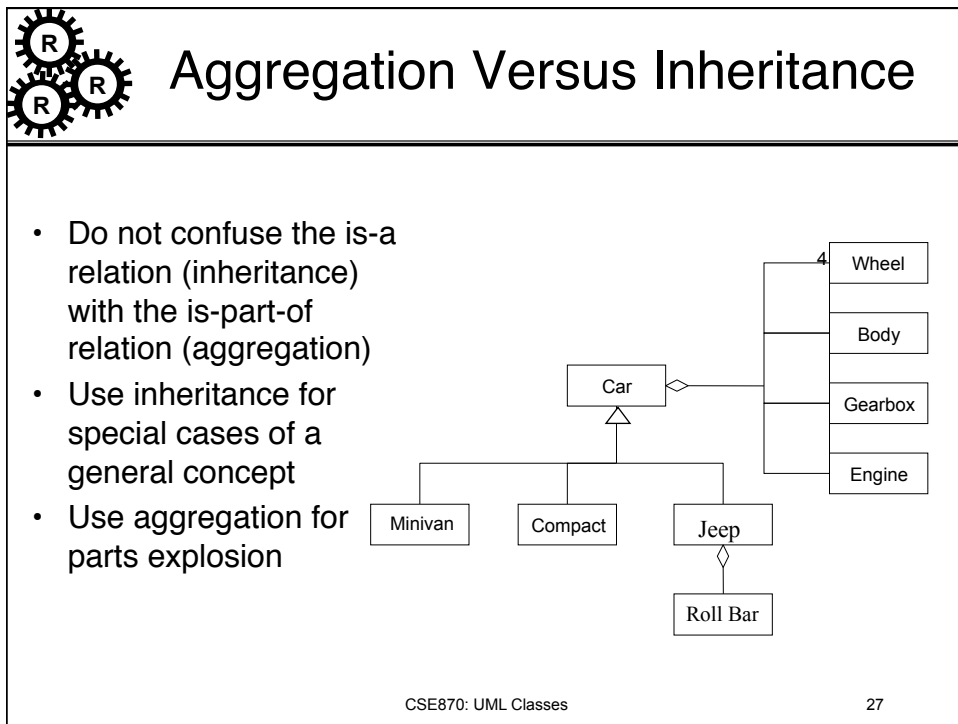
In this case, a link attribute is the only solution


CSE870: UML Classes 20











Object Modeling Summary

- Classes
 - Name
 - Attributes
 - Operations
- Associations
 - Roles
 - Link attributes
- Aggregation
- Inheritance

CSE870: UML Classes

29