Class Diagram

Dr. Rodrigo Spínola

Last time

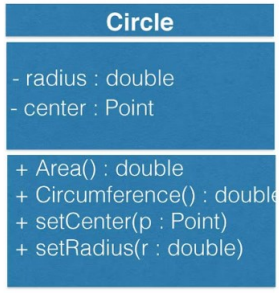
- Use case diagram is behavioral
- Maps to user stories or functional requirements
  - Describes the outside view of the system
    - From the point of view of a set of actors
  - Models system actions that yield an observable result
  - Simple, but effective for several purposes

Agenda

- Class diagrams
  - Class
  - Encapsulation
  - Association
  - Reflexive association
  - Aggregation
  - Association class
  - Inheritance
- Hands on activities

Class Diagram

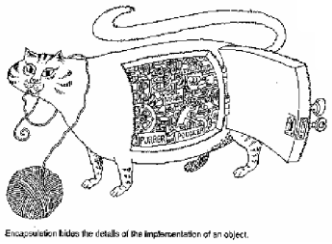
- Probably the most popular diagram in UML (structural diagram)
- Encodes classes and relationships between them
- This is a class
  - Attributes and operations
    - Can be of several types
      - + public
      - - private
      - #protected



Encapsulation

Describes the idea of bundling data and methods that work on that data within one unit. This concept is also often used to **hide the internal representation**, or state of an object **from the outside**. This is called **information hiding**.

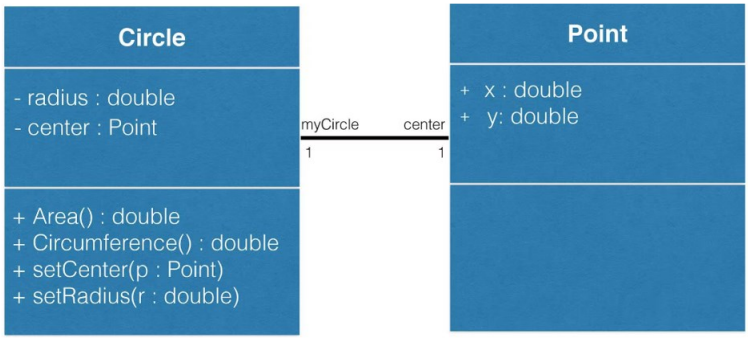
- For example, you have an attribute that is not visible from the outside of an object. You bundle it with methods that provide read or write access (getter and setter methods)
- Encapsulation allows you to hide specific information and control access to the internal state of the object



Encapsulation hides the details of the implementation of an object.

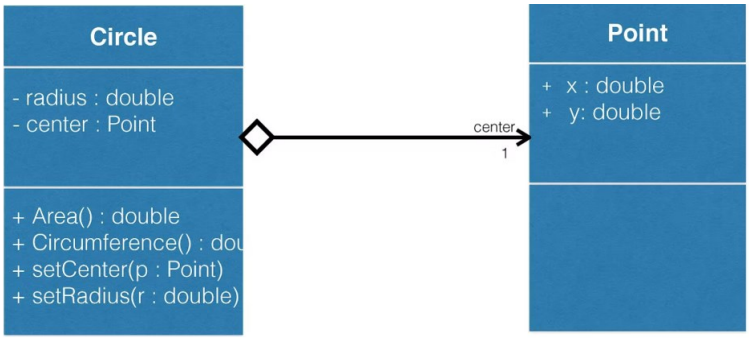
Class Diagram

- This is an association



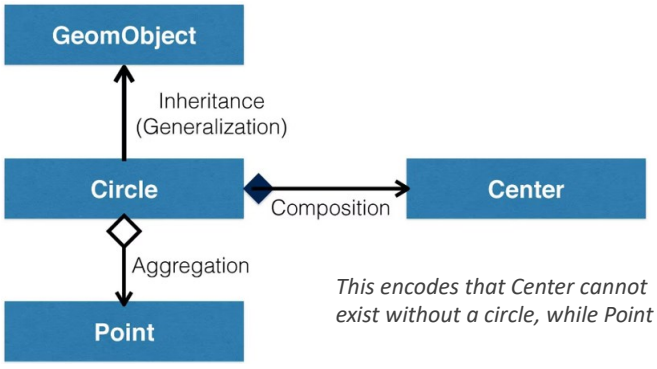
Class Diagram

- Aggregation makes more sense



Class Diagram

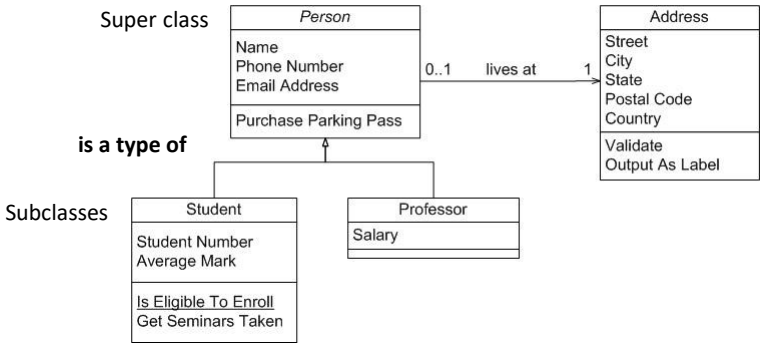
- More relationships



This encodes that Center cannot exist without a circle, while Point can

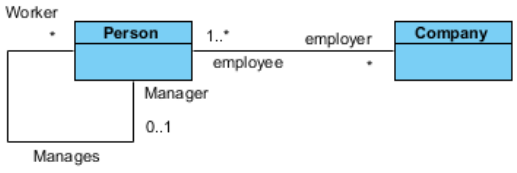
Inheritance (generalization)

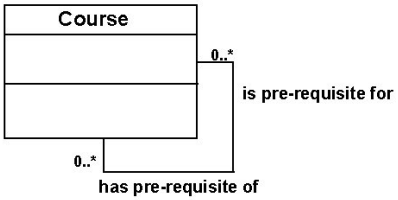
- Association ***is a type of***



Reflexive Associations

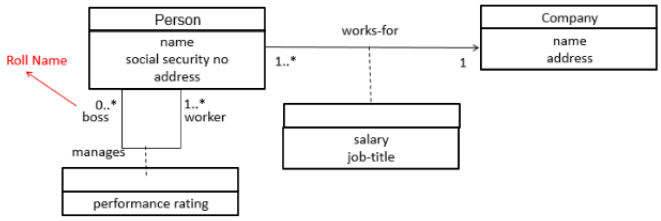
- Occurs when a class has an association with itself



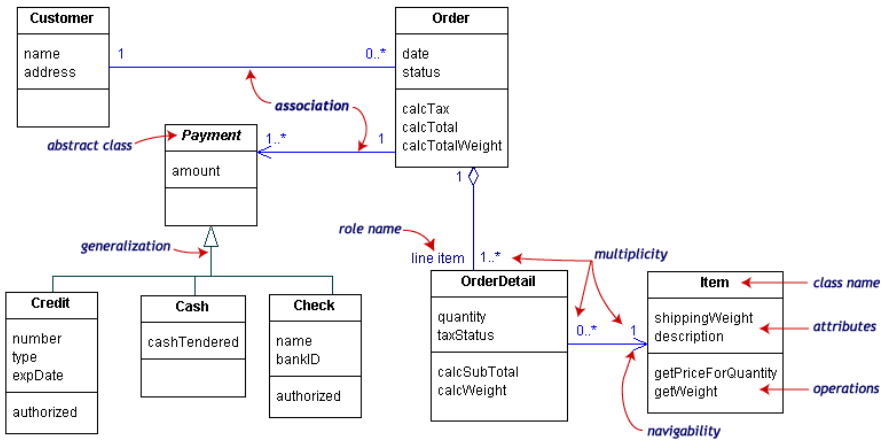


Association Class

- An association class is a class that is part of an association relationship between two other classes
  - You can attach an association class to an association relationship to provide additional information about the relationship
  - An association class is identical to other classes and can contain operations, attributes, as well as other associations

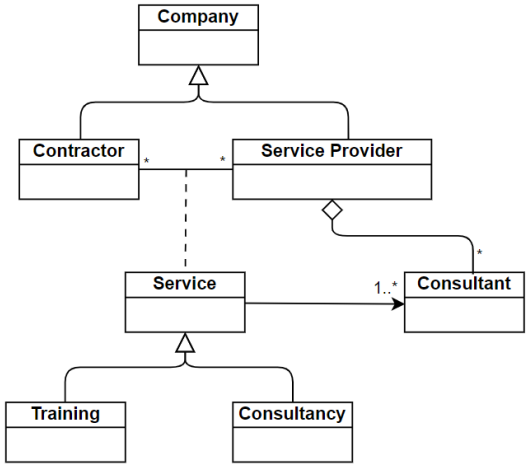


Another example



13

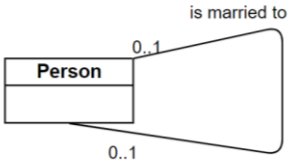
Another example

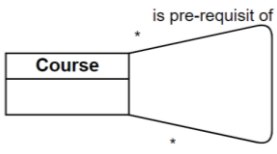


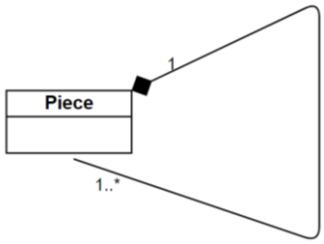
14

Class Diagram – Scenario 1

- Define a class diagram with relationships, role names, and multiplicities for the following situations:
  - A Person may be married to another Person
  - A Course can be prerequisite for one o more Courses
  - A Piece is composed of several other Pieces

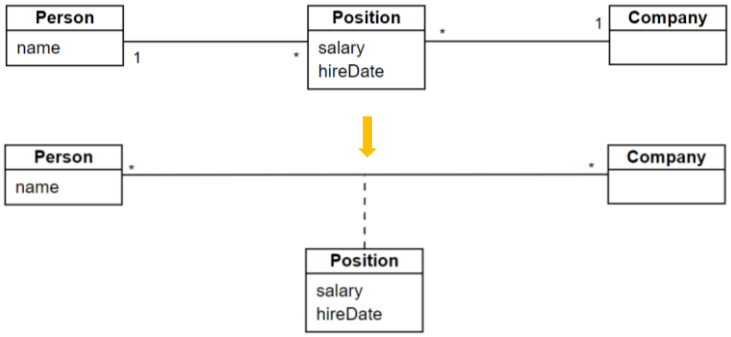






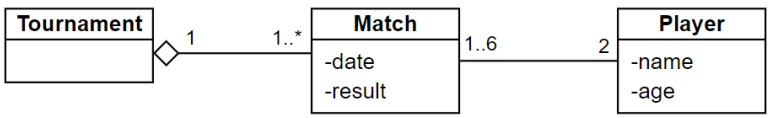
Class Diagram – Scenario 2

- Consider the following class diagram. Define an alternative to this diagram using an association class.



Class Diagram – Scenario 3

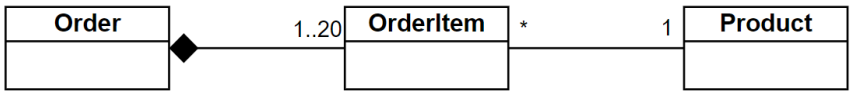
- Consider the following scenario concerning a tennis tournament management system: "In a tennis tournament, each match is played between 2 players. It is necessary to keep information about the name and age of the players; the date of the match and the assignment of the players to the matches. The maximum number of games a player can play is 6 and the minimum is 1."
- Define the corresponding class diagram.



17

Class Diagram – Scenario 4

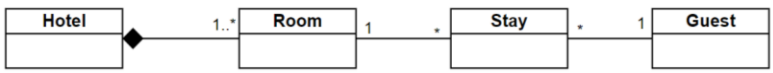
- Identify classes and their relationships considering the following business rules:
  - Orders are composed of several order items
  - An order item is related to one and exactly one product
  - An order can contain up to 20 items.



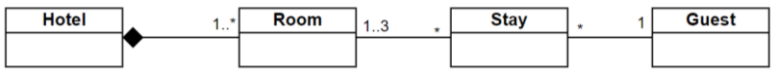
18

Class Diagram – Scenario 5

- Consider a hotel management system. Typically, one guest occupies one room per stay. But suppose a new rule was created: now, a guest can use up to three rooms. Define the class diagram for the two situations below:
  - a guest books a room

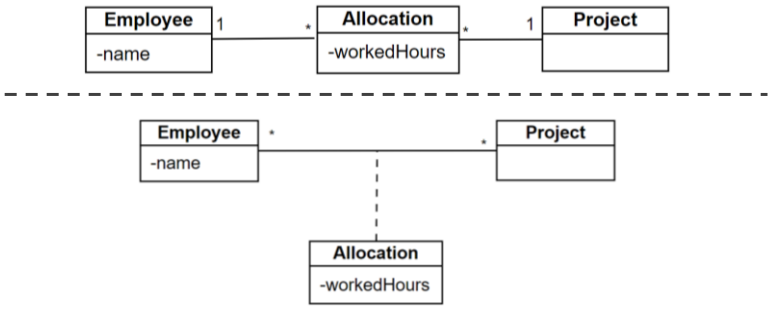


  - a guest books up to three rooms



Class Diagram – Scenario 6

- An employee may work on multiple projects. For purposes of calculating remuneration, it is necessary to know how many hours (s)he works on each project.



Class Diagram

Dr. Rodrigo Spínola

Lecture 22 - HoD: Class Diagram

TDresearchteam
Technical Debt Research Team

VCU Computer Science
College of Engineering