Interacting Classes and Object-Oriented Design

zyBook Chap 9.6

Be Creative and Reasonable

Four Main Principles of OOP

1. **Abstraction**

   To simplify reality and focus only on the properties and external behaviors rather than inner details

2. **Encapsulation**

   Hiding the implementation details (data and the programs that manipulate the data) of an object from the clients of the object

3. Inheritance
4. Polymorphism

Design a Program/Application

- Design Process
  - Determine the classes
  - Determine the responsibilities of each class
  - Determine the interactions and collaborations among the classes

Determine the Classes

- "Just as a **noun** is a person, place, or thing, so is an object."
- Begin by noting the **nouns** in the problem statement.
  - These nouns give us a good starting point for considering possible classes.
  - Not all nouns will become classes
  - Not all classes will correspond to nouns of the problem statement.

Task – Design a Student Class

- How?

Determine Responsibilities of Each Class

- "As the nouns indicate classes, the **verbs** of the problem statement help determine class responsibilities."
- Consider the following:
  - What service does the class provide?
  - What is each class's responsibility?
  - What are the actions and behaviors of each class?
  - What attributes/fields?

Brainstorm – Design a Student Class

- Fields for a Student?
  - Name
  - Preferred Name
  - Major
  - …
- Methods for a Student?
  - Generally, getter and setter methods
  - …

Design a Class

- Do not provide any functionality that does not have a clear use
- Limit object creation to the constructor
- Classes should have cohesion
  - The extent to which the code for a class represents a single abstraction
  - Allows for reusability of the class in other programs
- Classes should not have unnecessary dependencies
  - Coupling is the degree to which one part of a program depends on another
  - Related data and behavior should be in the same place (same class)

Brainstorm – Interacting with Student Class

- Objects that Student can interact with?
  - Name
  - Course
  - Dorm
  - Calendar
  - Faculty
  - Hometown
  - Restaurants
  - …

Coding Practice

- Step 1: Complete the TODOs in the classes
- Step 2: Complete the TODOs in the client program
- Step 3: Add more fields and methods to Student Class
- Step 4: Try to come up with objects that Student can interact with, such as Course, Dorm, Calendar, Faculty…