Northeastern University

Lecture 4: Java Review - 4

Prof. Chen-Hsiang (Jones) Yu, Ph.D.

College of Engineering

Materials are edited by Prof. Jones Yu from

Data Structures and Abstractions with Java, 5th edition. By Frank M. Carrano and Timothy M. Henry.
ISBN-13 978-0-13-483169-5

Outline

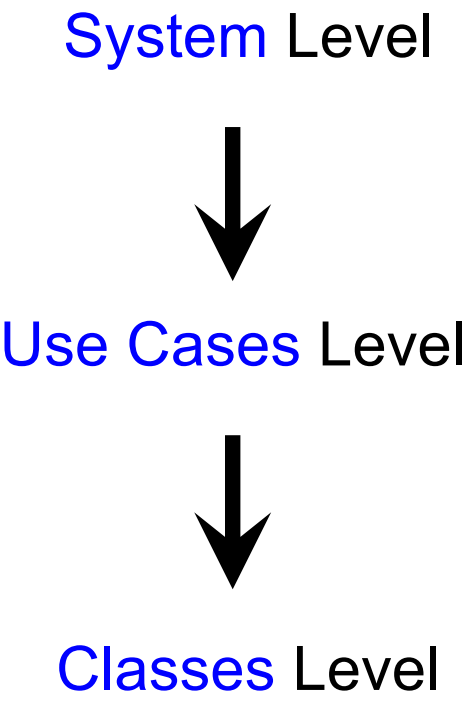
- System vs. Classes

System vs. Classes

Choosing Classes

- Considering a registration system for your school
- Issues:
  - » Who will use the system?
  - » What can each actor do with the system?
  - » Which scenarios involve common goals?

Different Levels



Choosing Classes

System Level

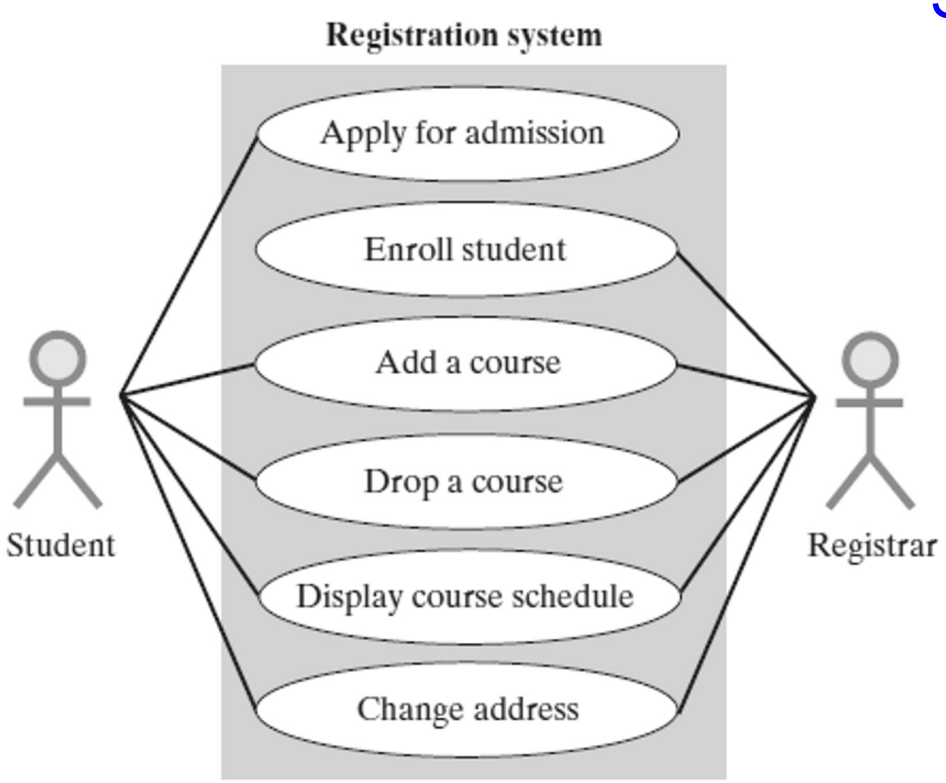


Figure P-4: A use case diagram for a registration system

Choosing Classes

System Level

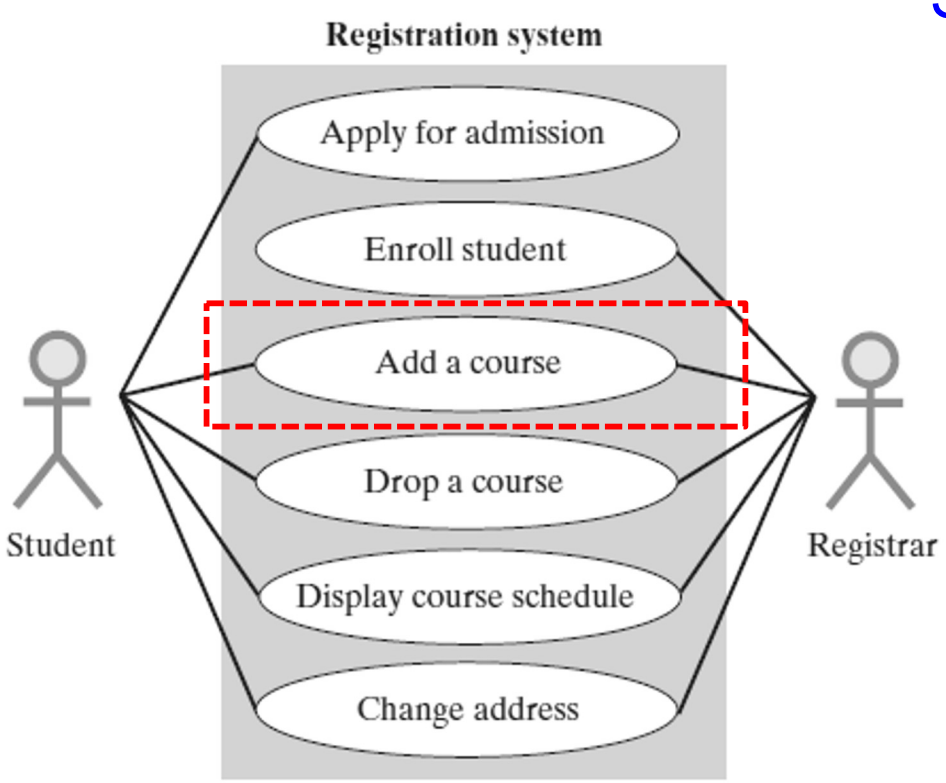


Figure P-4: A use case diagram for a registration system

Note: The use case diagram is part of a UML, which will be explained later.

Identifying Classes

Use Case Level

System: Registration
Use case: Add a course
Actor: Student
Steps:

1. Student enters identifying data.
2. System confirms eligibility to register.
   a. If ineligible to register, ask student to enter identification data again.
3. Student chooses a particular section of a course from a list of course offerings.
4. System confirms availability of the course.
   a. If course is closed, allow student to return to Step 3 or quit.
5. System adds course to student's schedule.
6. System displays student's revised schedule of courses.

Figure P-5: A description of a use case for adding a course

Interpretation of the Description

- Nouns: classes
  - a student (Student)
  - a course (Course)
  - a list of courses offered (CourseList)
  - a student's schedule of courses (CourseSchedule)
- Verbs: actions
  - confirm whether a student is eligible to register
  - see whether a course is closed
  - add a course to a student's schedule
- One way to assign actions to classes: class-responsibility-collaboration (CRC) cards

CRC Cards

Class Level

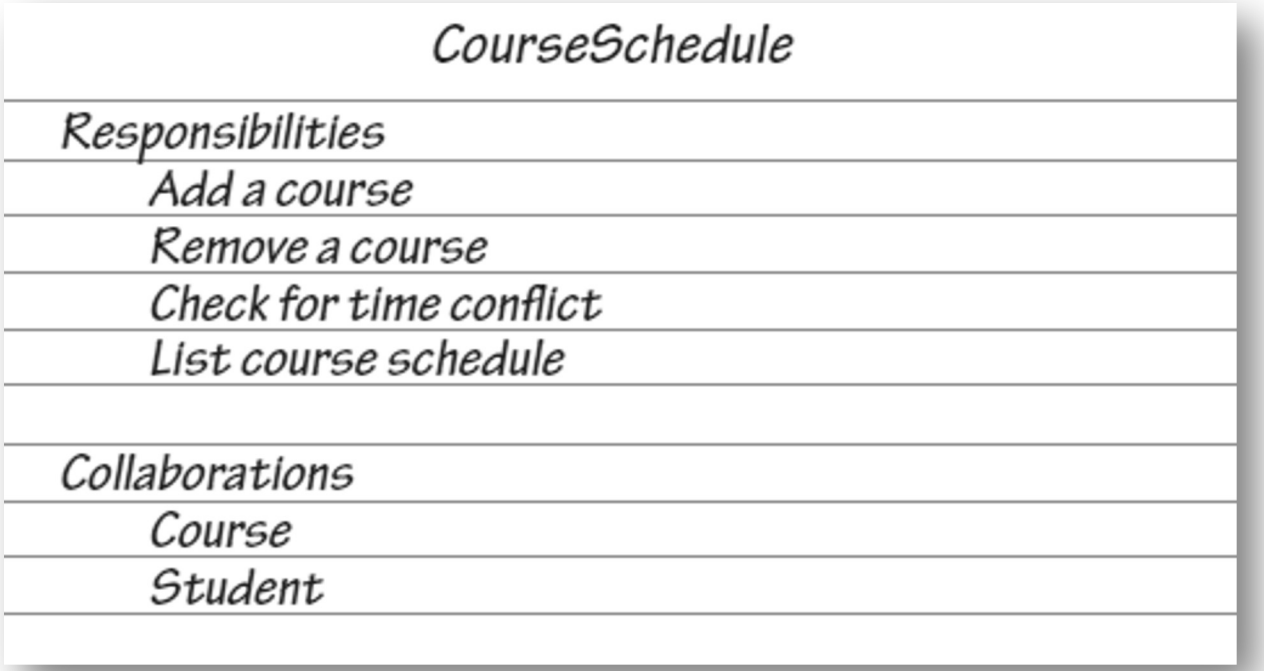


Figure P-6: A class-responsibility-collaboration (CRC) card

Exercise

- Write a CRC card for the class `Student`

Answer

Student
Responsibilities Set name and ID Set name Set ID Get name Get ID Get a string that represents a student
Collaborations String Name

The Unified Modeling Language (UML)

- Designers use the UML to Illustrate a software system's necessary classes and their relationship.
- Class diagram: place each class description into a box analogous to a CRC card
  - » class name
  - » attributes (data fields)
  - » operations (methods)
  - » Omit: constructors, get methods and set methods

The Unified Modeling Language (UML)

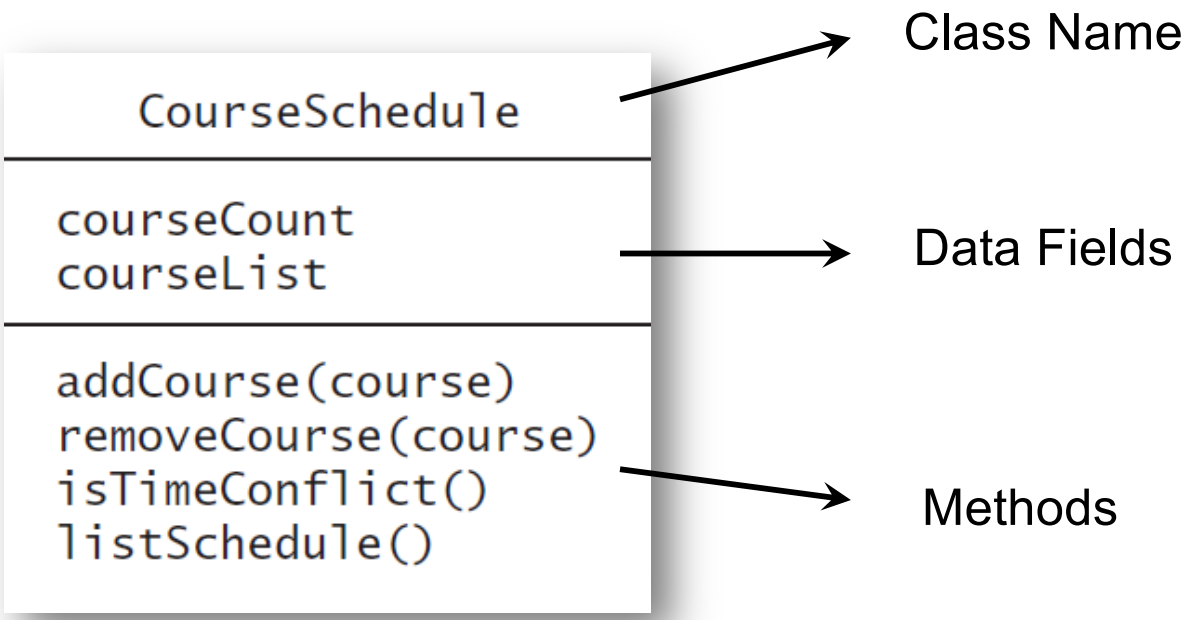


Figure P-7: A class representation that can be a part of a class diagram

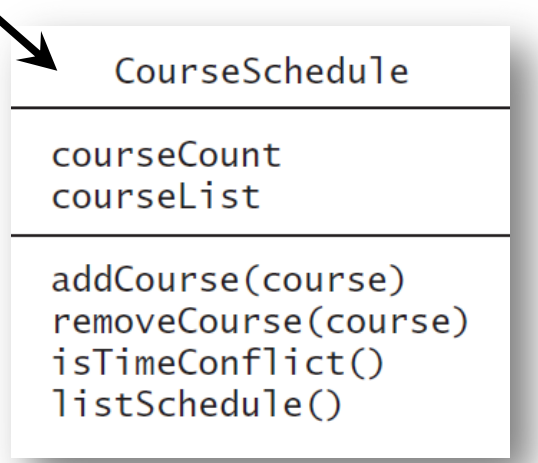
The Unified Modeling Language (UML)

- The visibility of a field or method:
  - +: public
  - -: private
  - #: protected
- The data type of a field, parameter, or return value are placed after a colon
  - data field: define data type
    - - courseCount: integer
  - method: define parameter and return value
    - + addCourse(course: Course): void

The Unified Modeling Language (UML)

- You can provide more details for Fig P-7:
  - » Data fields:

```
-courseCounter: integer
-coursList: List
```



  - » Methods:

```
+addCourse(course: Course): void
+removeCourse(course: Course): void
+isTimeConflict(): boolean
+listSchedule(): void
```

The Unified Modeling Language (UML)

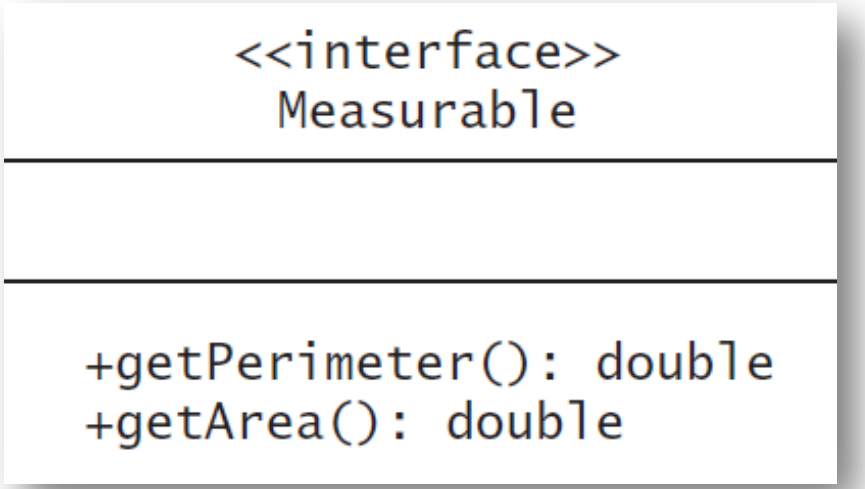


Figure P-8: UML notation for the interface **`Measurable`**

The Unified Modeling Language (UML)

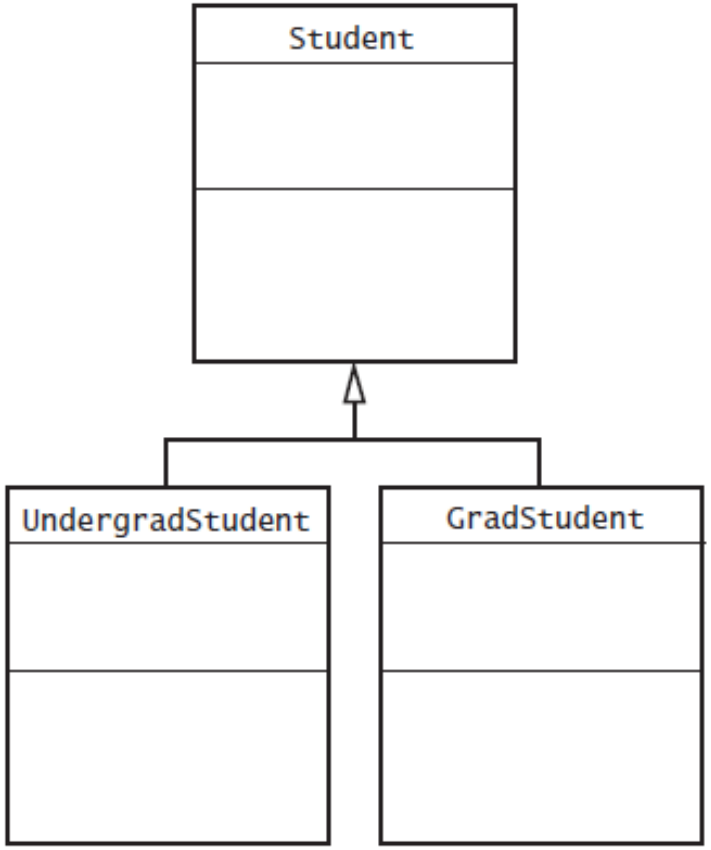
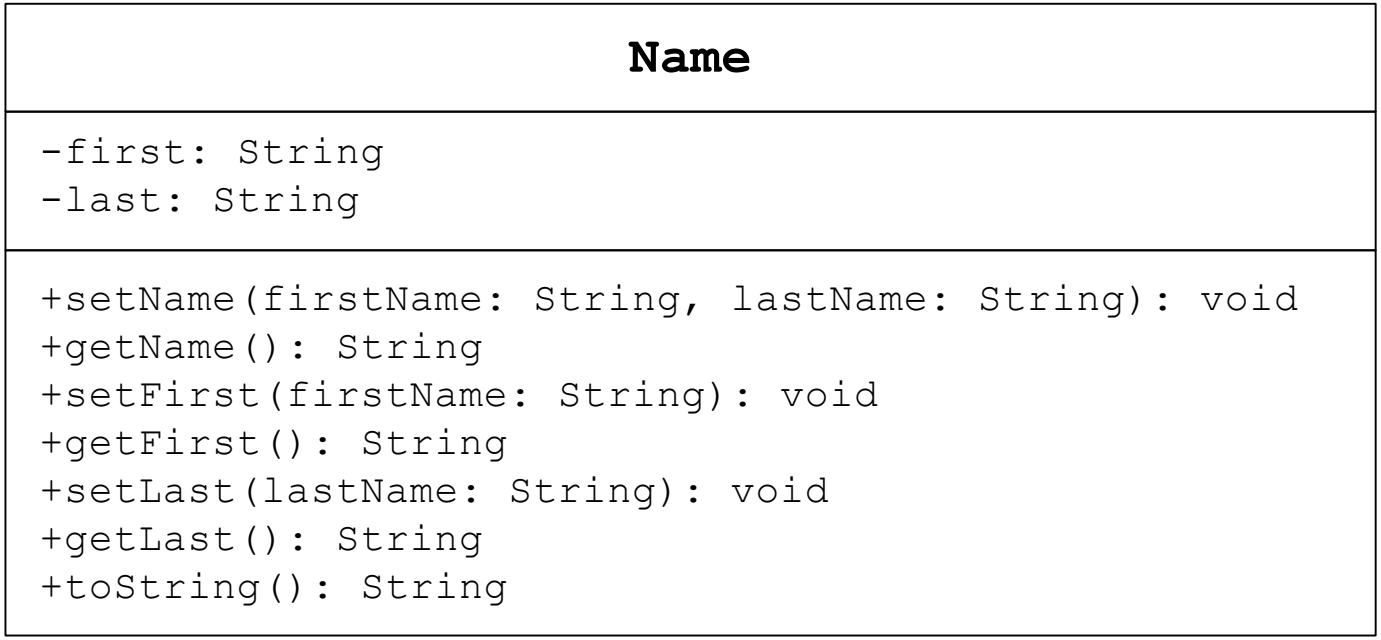


Figure P-9 A class diagram showing the base class `Student` and two subclasses

Exercise

- How would the class `Name` (used in previous slide) appear in a UML class diagram?

Answer


Name
-first: String
-last: String
+setName(firstName: String, lastName: String): void
+getName(): String
+setFirst(firstName: String): void
+getFirst(): String
+setLast(lastName: String): void
+getLast(): String
+toString(): String

The Unified Modeling Language (UML)

- An association:
  - » a relationship between two objects of different classes
  - » the same as collaboration at CRC card

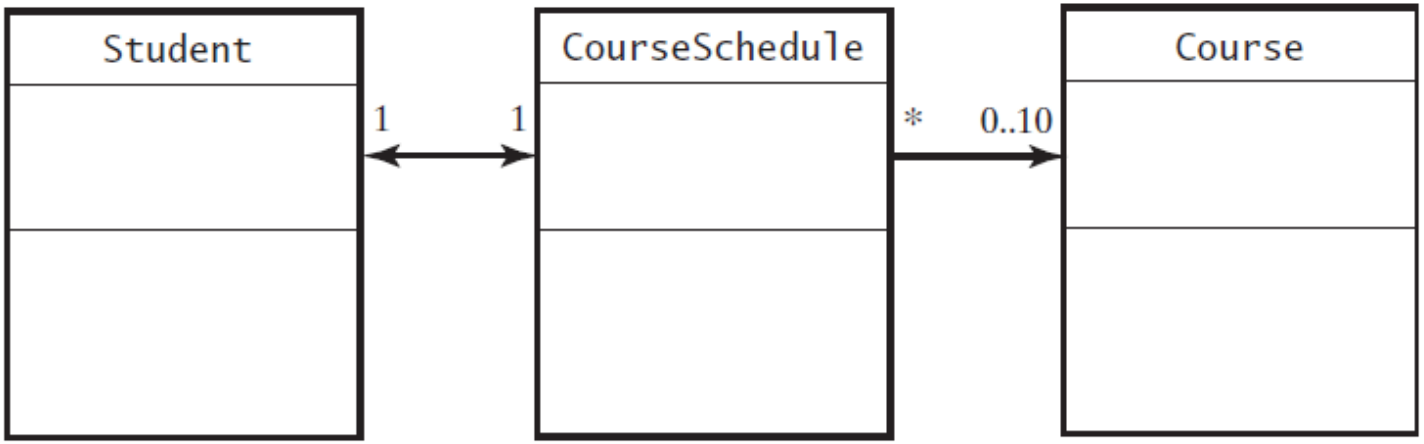


Figure P-10: Part of a UML class diagram with associations