

CISC 3115 TY2

# The protected Visibility Modifier and Java Package

Hui Chen

Department of Computer & Information Science

CUNY Brooklyn College

# Notice

- The slides are always subject to change.
- The slides posted before the lecture are for preview only, and they are a draft and their content can change significantly.

# Outline

- Discussed
  - The Object class and its methods
    - toString()
    - equals and its contract
- The protected visibility modifier

# Visibility Modifiers

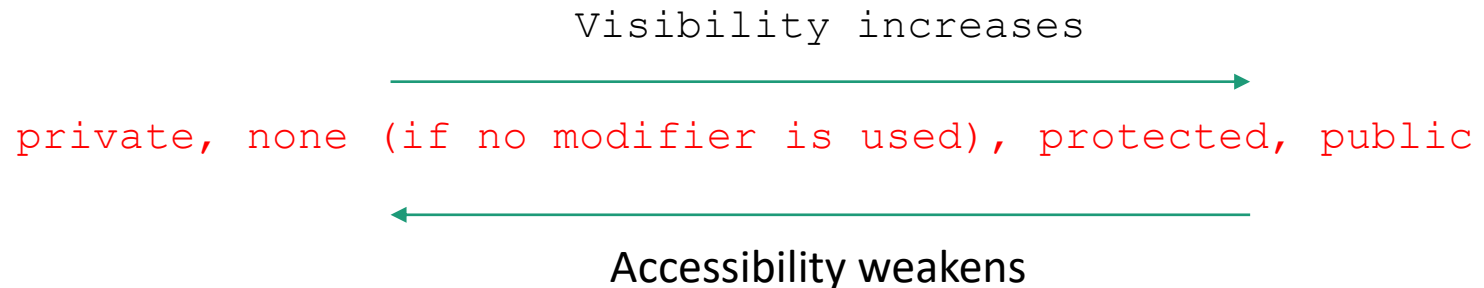
- Default (No visibility modifier)
- public
- private
- protected

# The protected Visibility Modifier

- It can be applied to data fields and methods in a class.
- A protected data field or method in a public class can be accessed by
  - any class in the same package, or
  - its subclasses, even if the subclasses are in a different package

# Comparing Visibility Modifiers

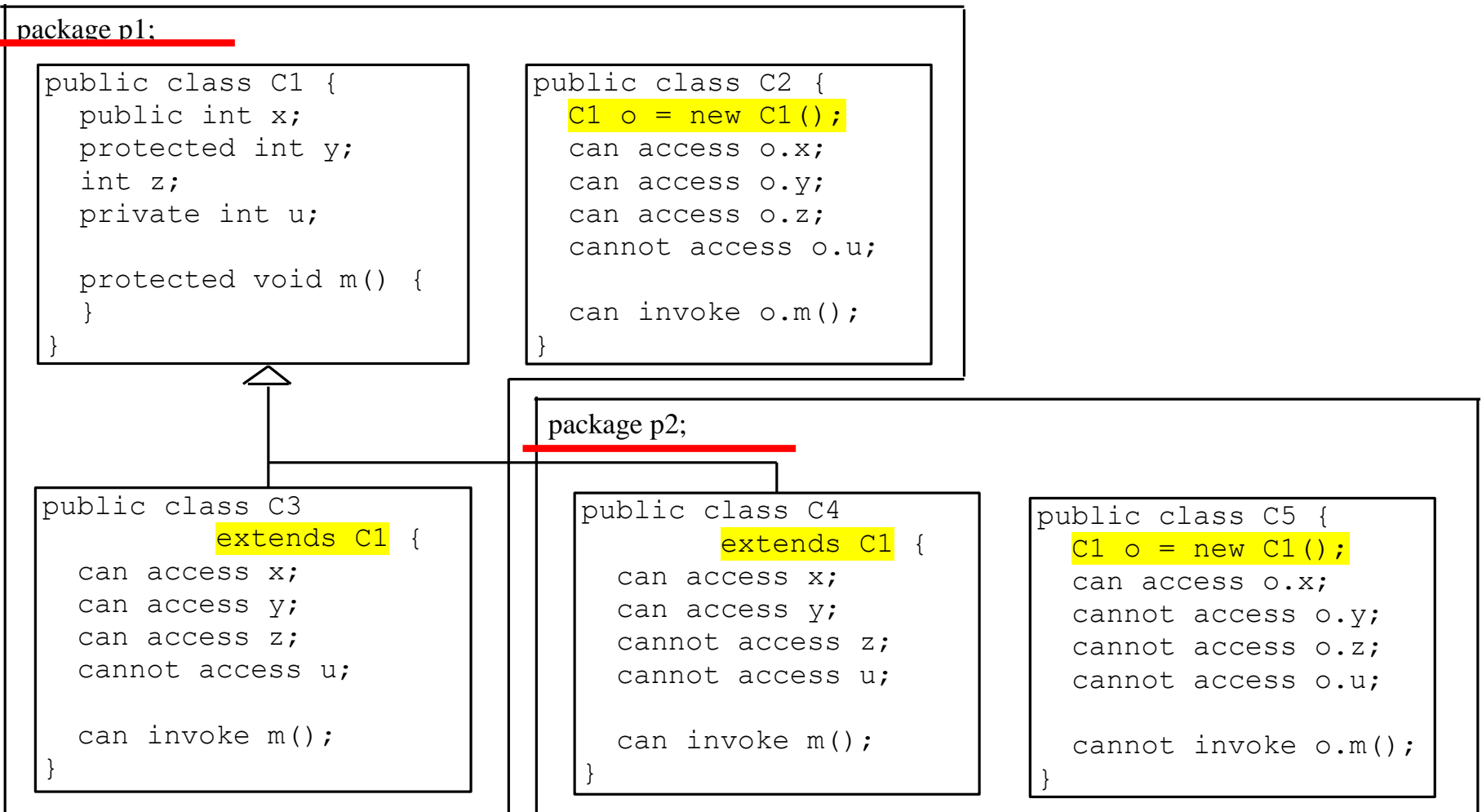
- private, default, protected, public



# Visibility Summary

Modifier on members in a class	Accessed from the same class	Accessed from the same package	Accessed from a subclass	Accessed from a different package
public	✓	✓	✓	✓
protected	✓	✓	✓	-
default	✓	✓	-	-
private	✓	-	-	-

# Example: Visibility Modifiers





# Remark: Package and Directory Structure

- Java package maps to a directory structure in a file system
- Example
  - package p1; // → directory p1
    - Classes in package p1 are in directory p1
  - package edu; // → directory edu
    - Classes in package edu are in directory edu
  - package edu.cuny; // → directory edu/cuny
    - Classes in package edu.cuny are in directory cuny that is in the edu directory

# Remark: Package and Directory Structure

- But where should you issue javac or java to compile or run the programs, respectively?
  - At the directory where the directories correspond to the package resides
  - package p1;
    - Directory p1's parent directory
- Use native terminal
  - Windows Command Prompt on Windows
  - Terminal from Mac OS X

# A Subclass Cannot Weaken the Accessibility

- A subclass may override a protected method in its superclass and change its visibility to public.
- However, a subclass cannot weaken the accessibility of a method defined in the superclass.
- For example, if a method is defined as public in the superclass, it must be defined as public in the subclass.

# Questions

- The protected visibility modifier
- Compare the accessibility /visibility of 4 different visibility modifiers
  - private, (none), protected, public

# Exercise

- Create subdirectory/subfolder in today's journal
- With your own addition/design, complete the code in [Slide 8](#) to demonstrate the visibility modifiers (public, protected, (default/package), and private)
- Submit your journal