School of Information Systems

SMU SINGAPORE MANAGEMENT UNIVERSITY

Object Oriented Application Development

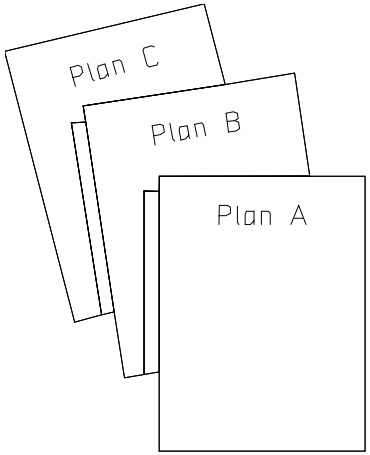
Inheritance and Packaging -Part 2-

I do not know anyone who has gotten to the top without hard work. That is the recipe. It will not always get you to the top, but it will get you pretty near.

Margaret Thatcher

Overview

- Objective
  - To be able to use packages and various access modifiers
- Content
  - Environment variables
  - Packages
  - Access Modifier
- After this module, you should be able to
  - Understand environment variables
  - Organize your classes into packages
  - Use the appropriate modifier for access control



Environment Variables

http://support.microsoft.com/kb/310519

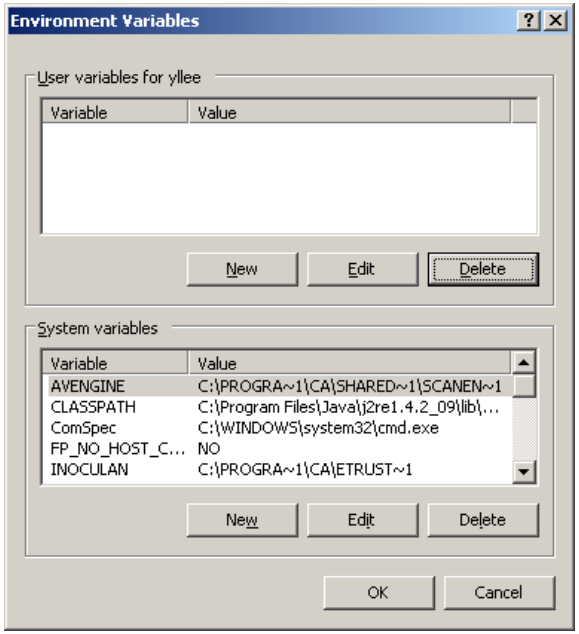
- String that contain information about the environment for the system, and the currently logged on user.
- Method 1: Command Prompt
  - Display

```
C:\>echo %path%
```

  - Modify

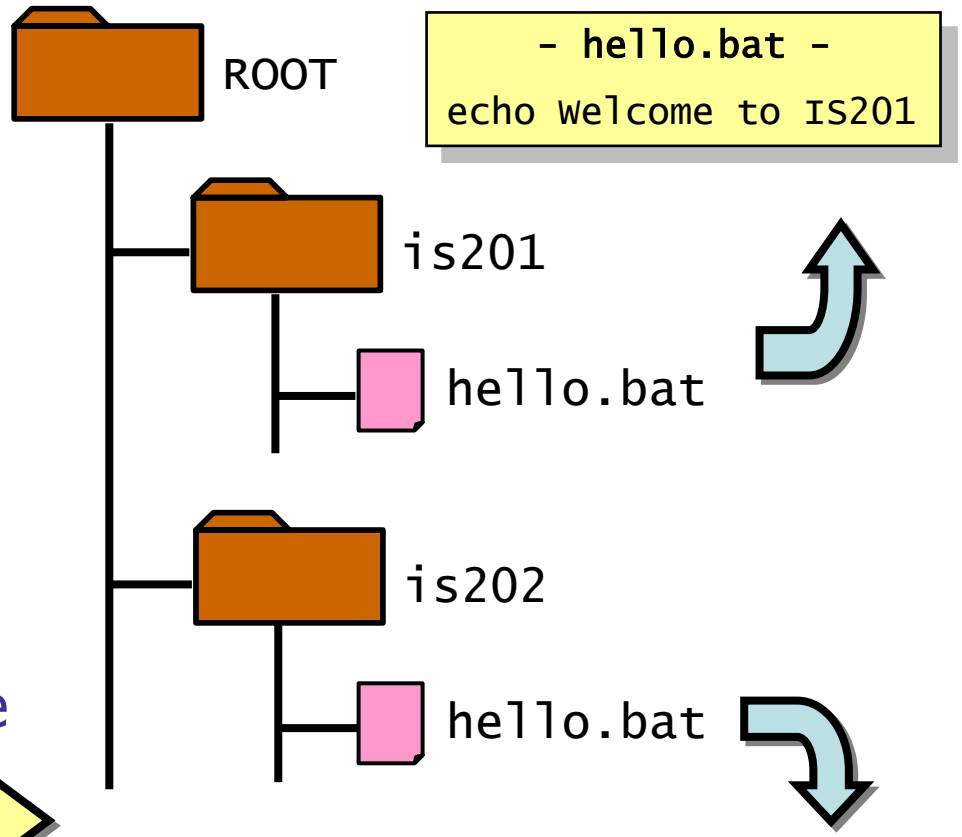
```
C:\>set path=%path%;c:\is201
```

- Method 2: Control Panel > System & Security > System
- Click Advanced System Settings
  - Click Environment Variables



Environment Variables

- PATH
  - A string containing folder locations where executables will be searched for by the Operating System
  - Executables are searched for in the folders specified in the PATH variable in order (left-to-right) for the first occurrence



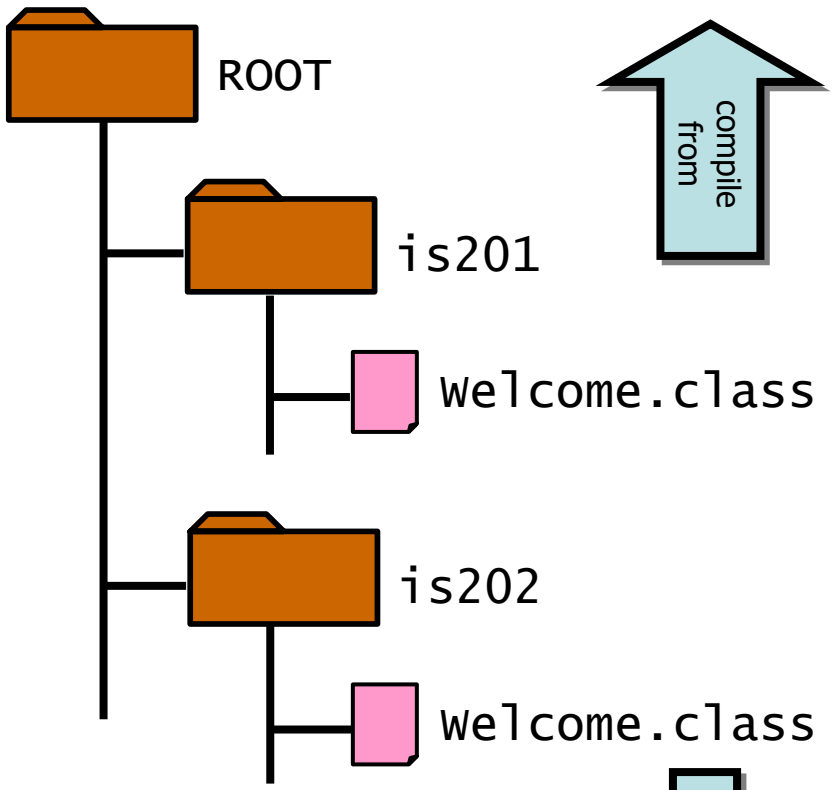
Search order

```
C:\>set path=c:\is201;c:\is202
C:\>hello

C:\>echo Welcome to IS201
Welcome to IS201
```

Environment Variables

- CLASSPATH
  - A string containing folder locations where class files will be searched for by the Java virtual machine (java.exe)
  - Executables are searched for in the folders specified in the CLASSPATH variable in order (left-to-right) for the first occurrence

```
- Welcome.java -
public class Welcome {
  public static void main(String[] args) {
    System.out.println("Welcome to IS201");
  }
}
```



Search order

```
C:\>set CLASSPATH=c:\is201;c:\is202
C:\>java Welcome
Welcome to IS201
```

```
- Welcome.java -
public class Welcome {
  public static void main(String[] args) {
    System.out.println("Welcome to IS202");
  }
}
```

Exercise 12: Path

- Locate the folder where your notepad++ executable is stored
  - Usually somewhere in C:\Program Files
- Add a reference in your path to the folder where notepad++ is stored
- Start notepad++ directly from the command prompt

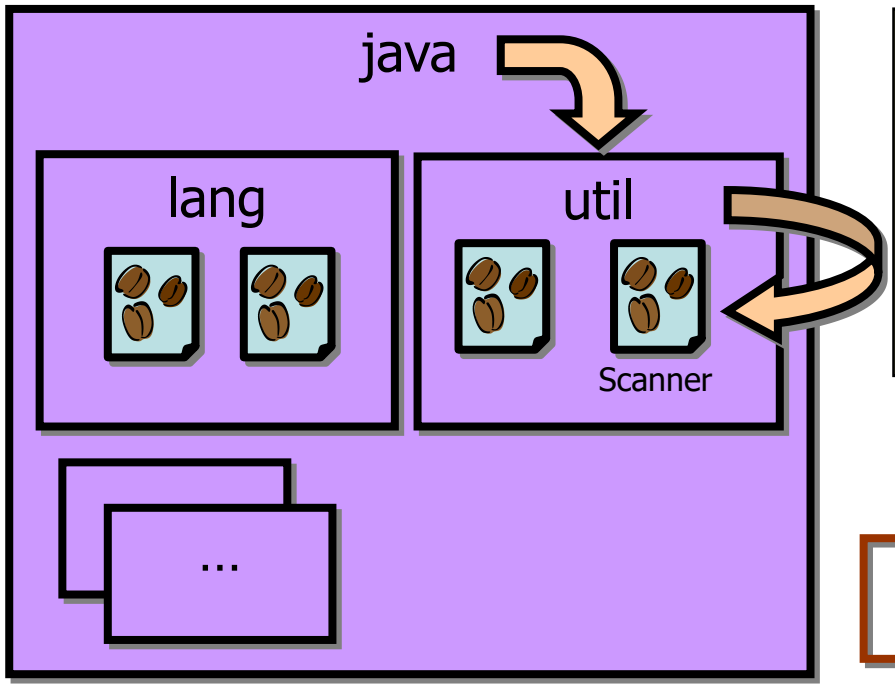
Exercise 13: Classpath

- Create two Greetings.java classes
  - The first one prints "Good Morning !"
  - The second one prints "Good Afternoon !"
- Create two folders: morning and afternoon
- Open a command prompt
- Set the classpath so that "Good Morning !" is printed whenever one types: java Greetings

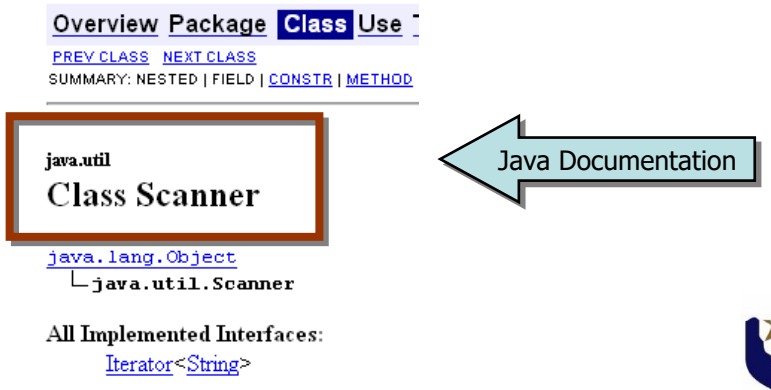
Packages

Java 5.0 Program Design P74

- To logically group classes.
- To access a class in another group, we use the import keyword.
- The character * refers to all classes in the package.




```
import java.util.*;
public class ScannerDemo {
  public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);
    System.out.println("Enter your name: ");
    String name = sc.next();
    System.out.println("Your name is " + name);
  }
}
```



Packages

- Allow classes to be collected together into a single grouping

```
package entity;

public class Employee {
  // ...
}
```

Employee will be added to the entity package

```
import entity.*;

public class EmployeeTest {
  public static void main(String[] args) {
    Employee e = new Employee("Lily");
  }
}
```

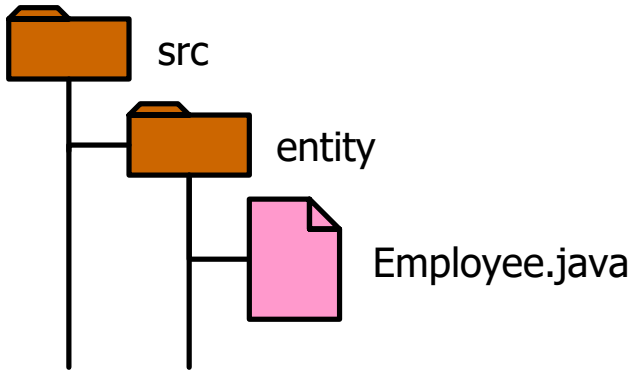
Import it to use it.

Creating a Package - 1

- Step 1: Include the package statement as the first statement

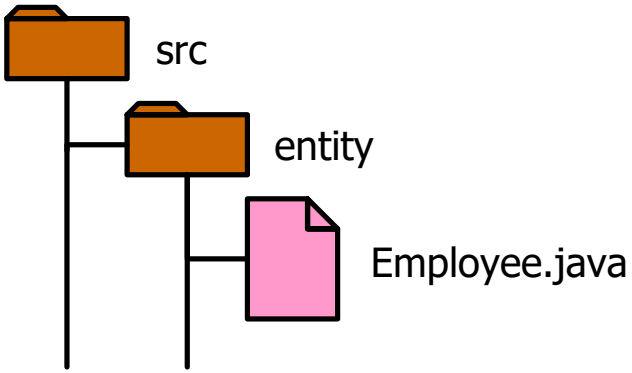
```
package entity;

public class Employee {
}
```

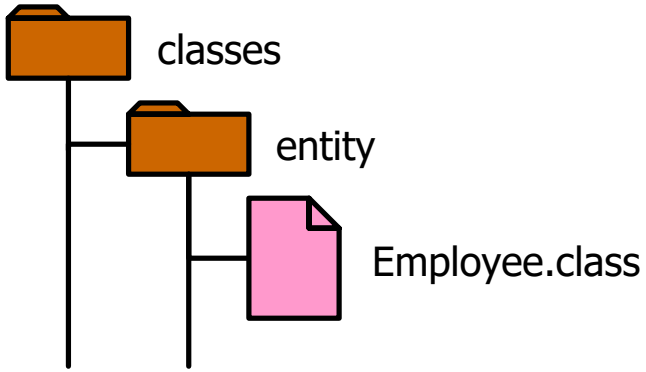
- Step 2: Create a folder named `entity`, the same name as the package name.

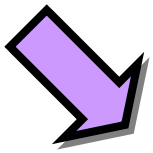


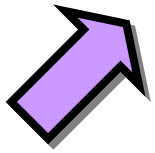
Creating a Package - 2

- Step 3: Compile the java file









```
D:\is201>javac -d classes src/entity/*.java
```

Note: The classes folder need to be created first

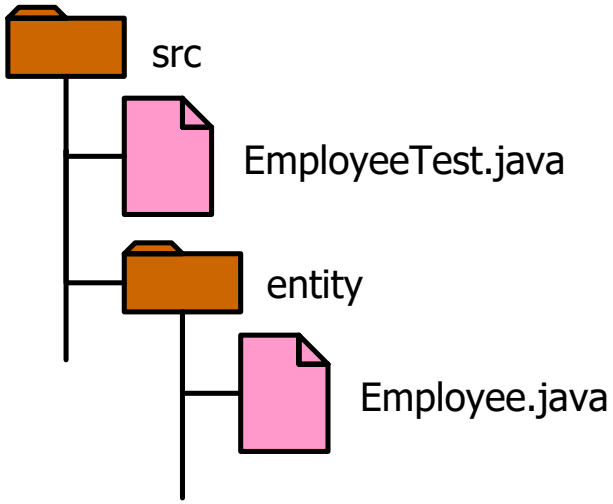
Creating a Package - 3

- Step 4: Import the package in your class and use it.

```
import entity.*;

public class EmployeeTest {
  public static void main(String[] args) {
    Employee e = new Employee("Lily");
    System.out.println(e.getName());
  }
}
```

Creating a Package - 4

- Step 5: Save the file in the right directory.



- Step 6: Compile and run it.

```
D:\is201>javac -cp classes -d classes src\EmployeeTest.java
```

```
D:\is201>java -cp classes EmployeeTest
```

Exercise 14: Packages

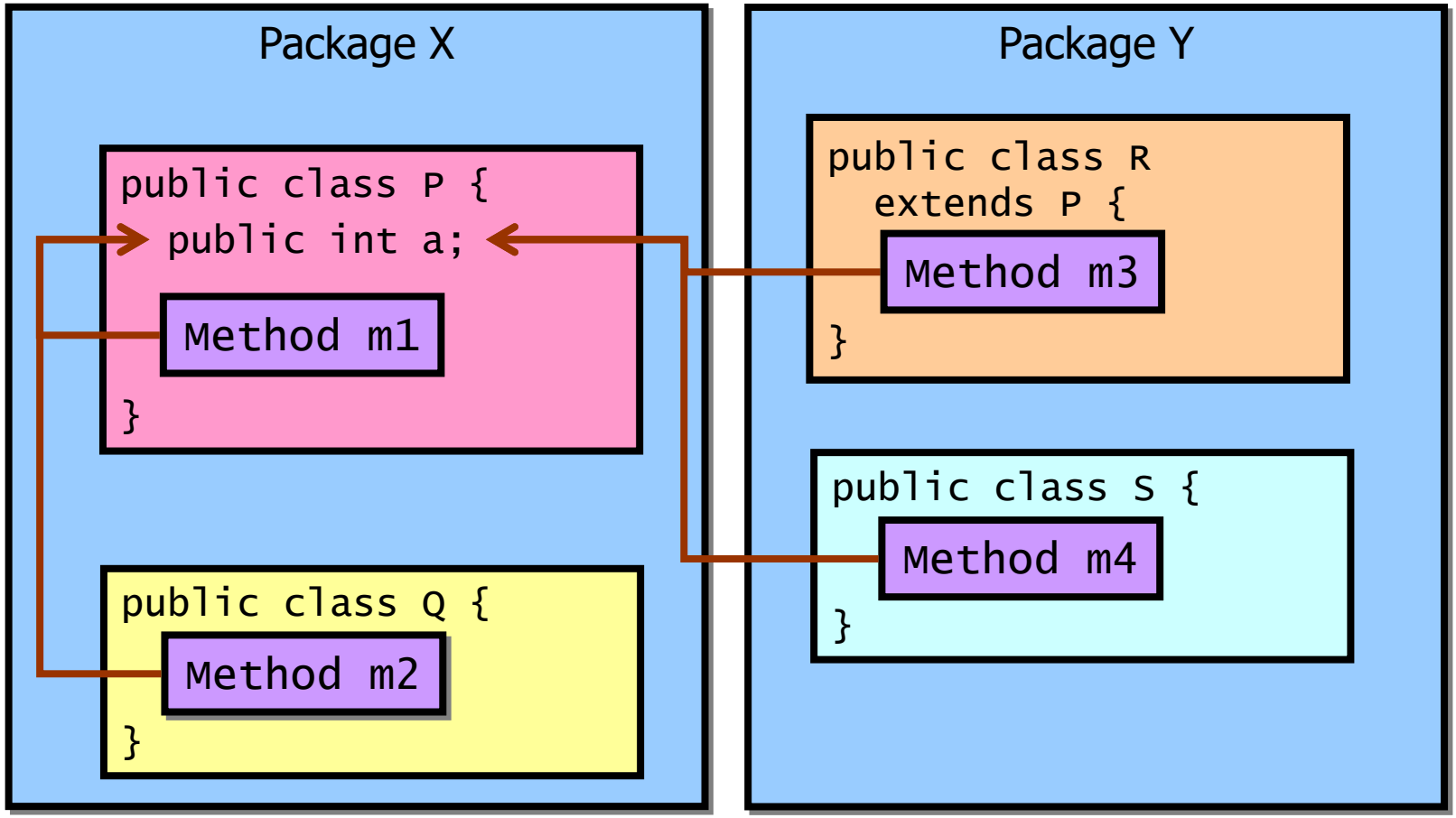
- Implement the Employee class.
  - The class has two attributes
    - `name` (of type `String`)
    - `salary` (of type `double`)
  - Implement getters and setters
  - Implement the `toString` method
- Place the `Employee` class in the `ooad` package.
- Implement the `EmployeeTest` class in the default package (i.e., no package is declared).
  - Declares and creates an instance of `Employee`
  - Displays the textual representation of the `Employee` object to the screen.

Controlling Access

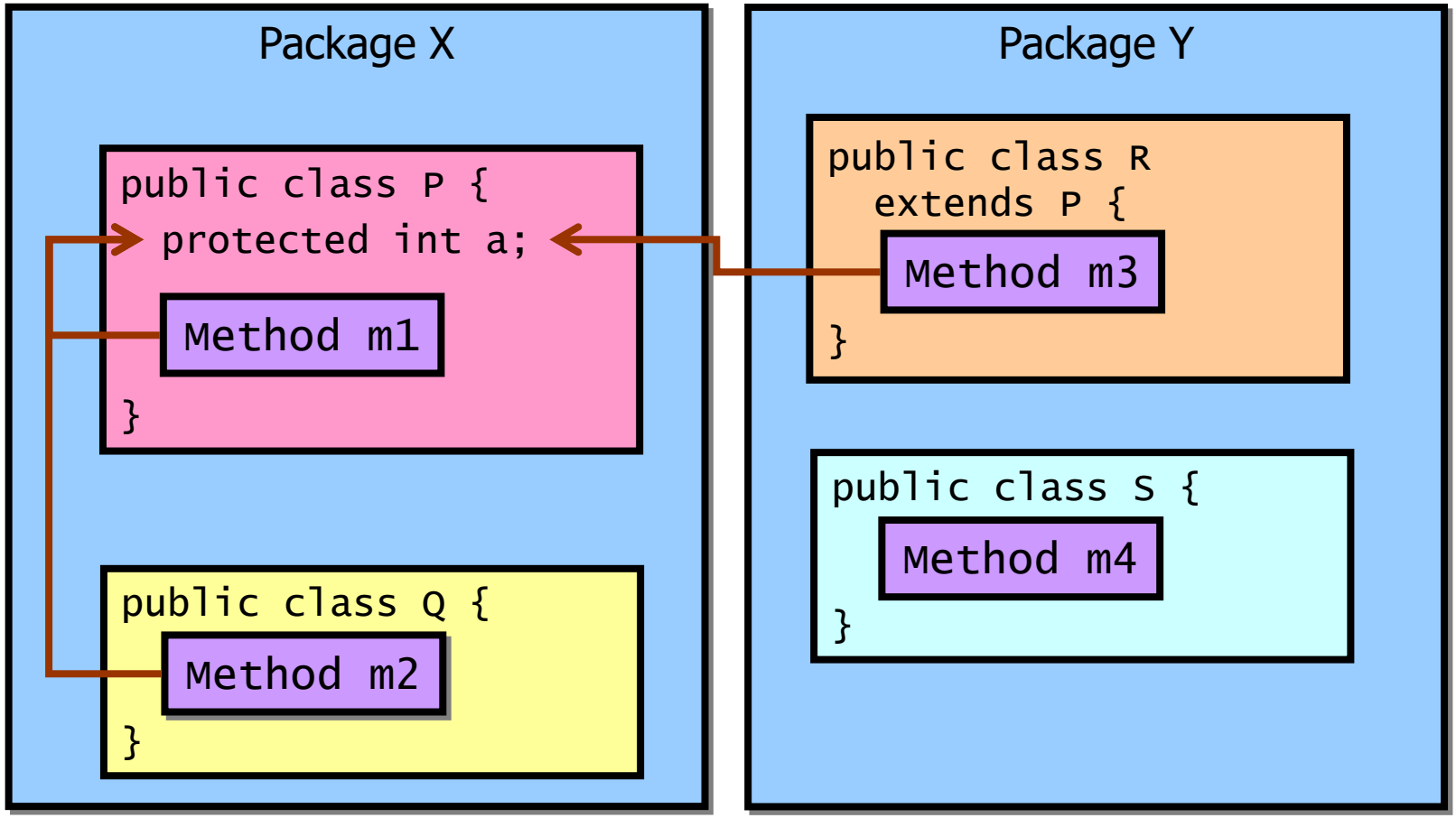
Java 5.0 Program Design P477

Modifier	Same Class	Same Package	Subclass	Universe
private	Yes			
(default)	Yes	Yes		
protected	Yes	Yes	Yes	
public	Yes	Yes	Yes	Yes

Public Access Modifier

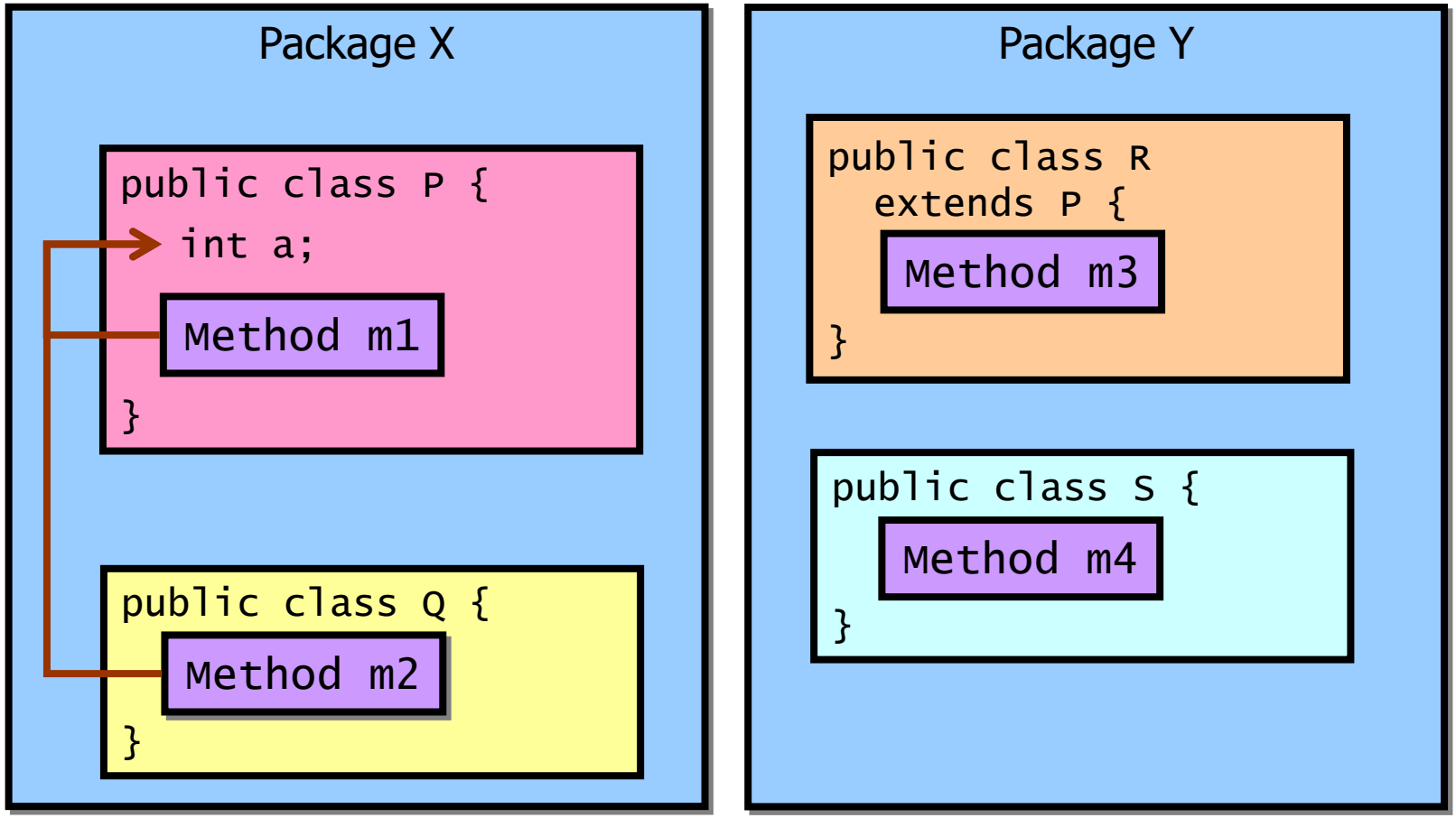


Protected Access Modifier

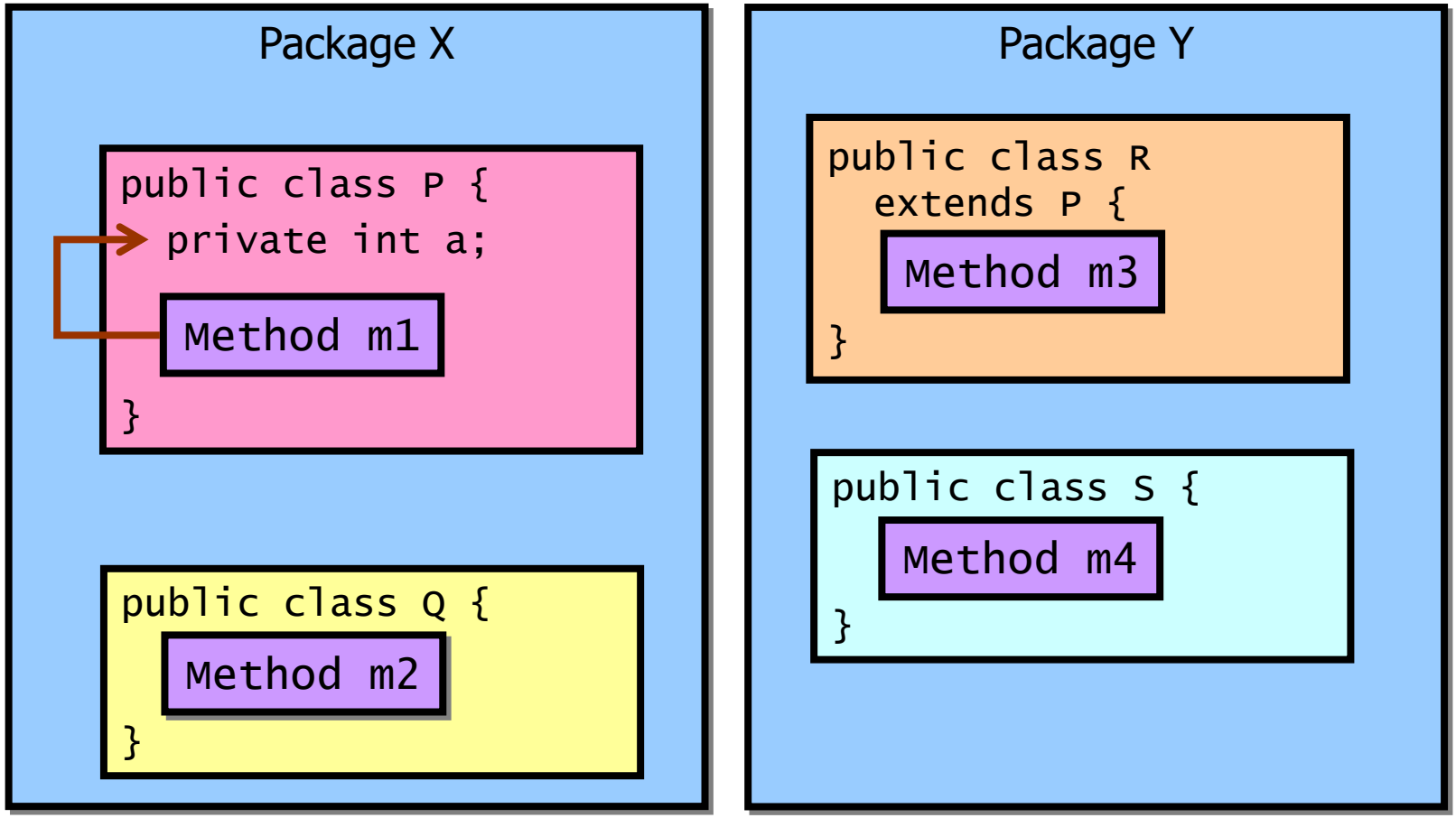


Default Access Modifier

- No keyword needs to be specified.

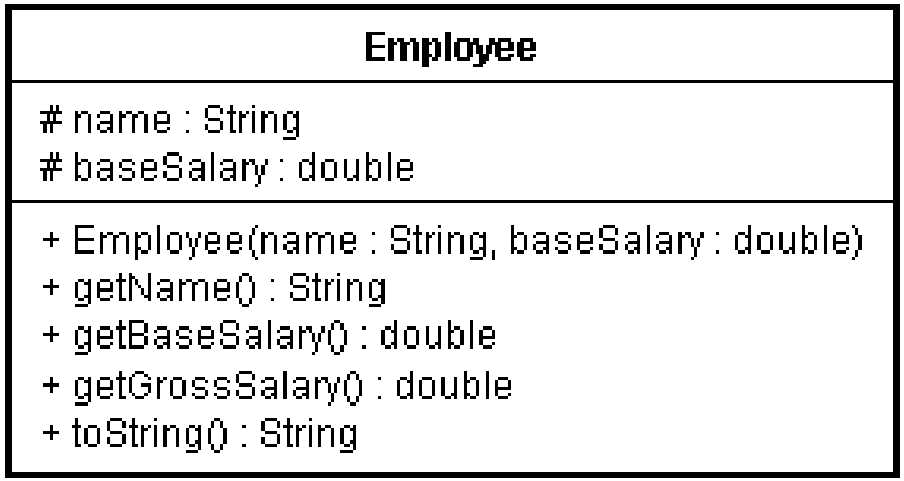


Private Access Modifier



Exercise 15: Access Modifiers

- Change the Employee class in Exercise 3 to follow the following class diagram.
  Note: # symbol represents protected access modifier



- Simplify the getGrossSalary() method of the Manager class.

Summary

- Environment variable
  - Path
  - Classpath
- Package allows classes to be collected together into a single grouping.
- Access Modifiers
  - public
  - protected
  - default
  - private