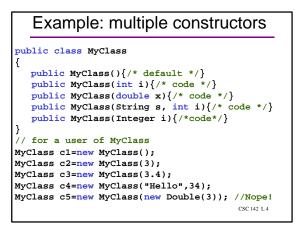


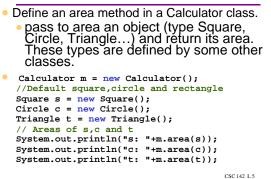
Method overloading

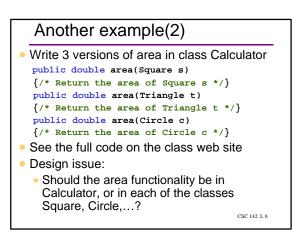
- In a class, you can define multiple methods with the same name.
- The compiler picks the right method based on the arguments passed to the method.
- Advantage: give the illusion that one method works on many types.
- Compiler gives an error if there is more than one possible match. If the match is not exact, the compiler does some automatic conversion.
- Complete matching algorithm rather complex

CSC 142 L 3



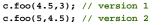
Another example (1)





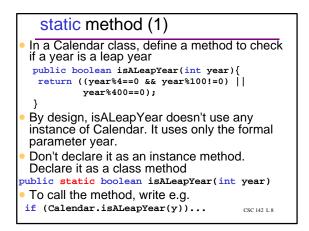
Ambiguity

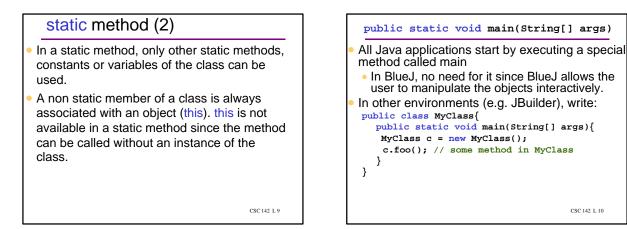
```
public class SomeClass{
  public void foo(double x, int i)
  {/*version 1*/}
  public void foo(int i,double x)
   {/*version 2*/}
}
// for a user of SomeClass
SomeClass c = new SomeClass();
```



- c.foo(1,2); // which one?
- The compiler will tell you.

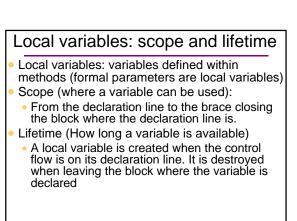
CSC 142 L 7





What about String[] args?

- When executing a java program,
- arguments can be provided to the program via the command line
- Write 'java className parm1 parm2 ...' (e.g. in a DOS window).
- parm1, parm2, ... are stored in the String array args
- Example: Salutations.java (available as a sample program).
- interactive mode: enter the name using a dialog box
- command line mode: write 'java Salutations Valerie' CSC 142 L 11



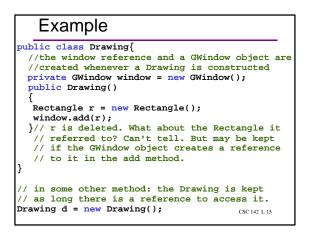
CSC 142 L 12

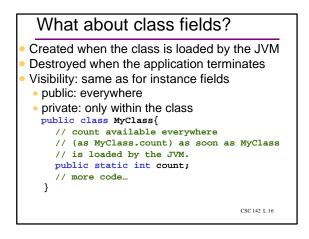
CSC 142 L 10

Example	
<pre>public void foo(int i) //i is created and initialized with the actua //parameter value every time foo is executed {</pre>	.1
<pre>// print the digits of i one by one do{</pre>	
<pre>int j; //created at every iteration</pre>	
<pre>System.out.println(j); i/=10;</pre>	
<pre>}while(i>0);//j is destroyed // create and initialize s String s="Thanks for using foo";</pre>	
<pre>System.out.println(s); }// s and i are destroyed CSC 142 L1 CSC 142 L1</pre>	13

Object: scope and lifetime

- An object can be used wherever a reference to the object is available.
- Visibility of the instance fields:
 - everywhere if declared public within the class of the object
 - only within the class of the object if declared private
- Lifetime of an object
- Created by the call to the class constructor (using new)
- Destroyed when all references to the object are destroyed (the memory is reclaimed by the garbage collector).

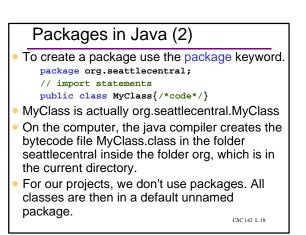




Packages in Java (1)

- A package is a named collection of classes. It defines a namespace for the classes that it contains.
- For large projects, packages prevent name collision.
- e.g., if you create a JOptionPane class, the compiler won't confuse it with the usual JOptionPane class, which is actually the javax.swing.JOptionPane class.

CSC 142 L 17



Program organization

- Write each public class in its own java file • public class Date is written in the file Date.java
- Can have more than one class in a java file. But only one of them is public public class Class1{ /* code */}

class Class2 {/* code */}

Class1 and Class2 are in the same package. If no package is specified, that would be the default package. But, only Class1 is visible to another class importing the package.

CSC 142 L 19

More on scope

- An instance field may be declared without any access modifier (public or private). Its visibility is restricted to the package of the class.
- Inside a package, any class has access to the data and method members of other classes as long as the members are not declared private.
- But only public members of public classes are visible outside the package. public class Class1{

```
int i; /* package access */
private int j; /* Class1 only */
public int k; /* everybody */ }
class Class2 {
     public int a; /* package access */
int b; /* package access */
private int c; /* Class2 only */} csc142 L20
```