Inheritance

COMP53
Sept 5 2007
Inheritance

- Inheritance allows us to define new classes by extending existing classes.
- A child class inherits all members from the parent class.
- Parent class = base class or super-class.
- Child class = derived class or sub-class.
Unified Modeling Language

- UML is a graphical language for designing classes
- Shape is the base class
- Circle and Rectangle are derived classes
UML Class Declaration

- UML class declarations show fields, methods and their modifiers

<table>
<thead>
<tr>
<th>protected fields</th>
</tr>
</thead>
<tbody>
<tr>
<td>#center: Point2D</td>
</tr>
<tr>
<td>#shape_type: String</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>public methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>+Shape(_center:Point2D)</td>
</tr>
<tr>
<td>+getArea(): double</td>
</tr>
<tr>
<td>+contains(_p:Point2D): boolean</td>
</tr>
<tr>
<td>+getShapeType(): String</td>
</tr>
<tr>
<td>+toString(): String</td>
</tr>
</tbody>
</table>
Subclass Design

- A Circle has all the fields and methods defined by Shape, and some additional members
Another Subclass

```java
public class Shape {
    #center: Point2D
    #shape_type: String

    +Shape(_center:Point2D)
    +getArea(): double
    +contains(_p:Point2D): boolean
    +getShapeType(): String
    +toString(): String
}

public class Circle {
    -radius: double
    +Circle(_center:Point2D, _radius:double)
    +getArea(): double
    +contains(_p:Point2D): boolean
}

public class Rectangle {
    -width: double
    -height: double
    +Rectangle(_center:Point2D, _width:double, _height:double)
    +getArea(): double
    +contains(_p:Point2D): boolean
}
IS-A Relationship

• We say that an object of a sub-class IS-A object of the super-class
• A Circle IS-A Shape
• A Rectangle IS-A Shape

• This means that a Circle or a Rectangle can be used anywhere that a Shape is expected.
Using Derived Class Objects

// array of Shape references
Shape[] shapes = new Shape[10];

// use a Circle as a Shape
shapes[0] = new Circle(new Point2D(1,1), 10);

// use a Rectangle as a Shape
shapes[1] = new Rectangle(new Point2D(2,3), 10, 20);
Using Derived Class Objects

// print area of a Circle
System.out.println(shapes[0].getArea());

// print area of a Rectangle
System.out.println(shapes[1].getArea());

Through dynamic dispatch, Java knows the true class of each object and can call the appropriate getArea() method for that class.