This Week

- P2 Final Milestone Due Thursday
- Midterm 2 next Thursday
  - Exam conflict arrangements emailed this week.
  - See Piazza for Review Questions
- Review & More Object Concepts, OO Design
Class vs Instance

Sponge Painted Owls
with Owl Template

How are these terms related?

field, attribute, member variable
class, related data and methods, template
object, instance
instance variable, non-static variable
class field, static field, static variable
class Picture {
    private boolean hasFrame;
    public Picture(boolean hasFrame) {
        this.hasFrame = hasFrame;
    }
}
Does this print true or false?

class Person {
    private boolean something = false;
    boolean getThing(boolean something) {
        return this.something;
    }
}

public static void main(String []args) {
    Person p = new Person();
    System.out.println( p.getThing( true));
}

| true     |
| false    |
| error/other |
Does this print 0, 1, other or error?

```java
public class Person {
    static int count = 0;
    private boolean something = false;
    Person(boolean something) {
        this.something = something;
        count++;
    }
}
```

`System.out.println(Person.count);` // in other class in package
Visibility Modifiers

For members of a class:
- public
- private
- protected
- <package>

Demo
Can methodA call methodB?

//classes in different files in same package

class A {
    public void methodA() {
        B b = new B();
        b.methodB();
    }
}

class B {
    void methodB() {
    }
}

<table>
<thead>
<tr>
<th></th>
<th>yes</th>
<th>no</th>
<th>depends</th>
<th>error</th>
</tr>
</thead>
</table>

Can a method outside the package call methodA()?

//classes in different files in same package

class A {
    public void methodA() {
        B b = new B();
        b.methodB();
    }
}

class B {
    void methodB() {
    }
}
class Employee {
    private static int employeeCount = 0;
    private final int id;
    Employee() {
        this.id = ++employeeCount;
    }
    public static void main(String[] args) {
        Employee anEmployee = new Employee();
        System.out.println(anEmployee.id);
    }
}
Default Constructor

class Person {
    private String name;
}

Can we create an instance?
Person p = new Person();
class Person {
    private String name;

    Person() {
        this("no name");
    }

    Person(String name) {
        this.name = name;
    }
}
Constructors

If there is no constructor in a class is a constructor automatically provided by the compiler?

| yes - the default, no argument constructor |
| yes - a really good one. |
| no |
| error |
Will the default constructor be provided in this case?

```java
class Cup {
    private String contents;
    Cup(String contents) {
        this.contents = contents;
    }
}
```
Will toString return results of Graphic toString()?

class Bird {
    private Graphic graphic;
    Bird(String name) {
    }
    public String toString() {
        return this.graphic.toString();
    }
}

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>yes</td>
<td>no</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
|     |      |NullPointerException
|     |      | other
Which are true?

class Light {
}

// in some method
Light aLight = new Light();
System.out.println(aLight);

<table>
<thead>
<tr>
<th>error - no constructor</th>
</tr>
</thead>
<tbody>
<tr>
<td>Object classes' toString() will be called</td>
</tr>
<tr>
<td>an instance of Light has nothing in it</td>
</tr>
<tr>
<td>error</td>
</tr>
</tbody>
</table>
How many Bug instances in list?

ArrayList<Bug> list;
list = new ArrayList<Bug>();
list.add( new Bug());
Bug aBug = new Bug();
list.add( aBug);
list.add( 0, aBug);

<table>
<thead>
<tr>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 copies of reference to 1 bug</td>
</tr>
<tr>
<td>none, error, no list</td>
</tr>
<tr>
<td>3</td>
</tr>
</tbody>
</table>
Circle Class

Design a Circle Class
- Field: radius
- Constructor: radius is the argument
- Methods: getArea(), getCircumference(), toString()
- Recall: Area = π * radius * radius; Circumference = π × diameter

Draw a UML Class diagram

Create TestCircle Class
- Create circles with radius 3.5 and 34.1
- Print out area, circumference, and radius
Rectangle

1. Design a Rectangle class
   - **Fields**: width & height as double with default of 1.0 and private
   - **Constructors**: no-arg constructor & a constructor with specified width and height, public
   - **Methods**: getArea() and getPerimeter(), public

2. Draw a UML diagram for the class then implement the class.

3. Write a TestRectangle program that:
   - Creates 2 rectangles (4 by 10) and (3.5 by 25.4)
   - Display width, height, area and perimeter
Bike

Design a bike class.

- **Instance Fields**: numWheels, Color, unique id
- **Class Field**: numBikesCreated, used to assign unique id’s to each bike.
- **Constructor**: numWheels and Color, automatically sets the unique identifier.
- **Instance Methods**: Number of Wheels and id can be accessed but not changed. Color can be changed. Add a toString() method to return all instance field values in String form.
- **Class Method**: returns the number of bikes created.

Draw the UML diagram and then write the code.

Create a BikeShop class that creates 10 bikes and stores in an array.

Print out each bike’s number of wheels, color and id using the toString method.