

ICS 141

Programming with Objects

Jessica Maistrovich
Metropolitan State University

Seeing objects

hidden-3d.com hidden-3d.com hidden-3d.com hidden-3d.com hidden-3d.com hidden-3d.com hidden-3d.com hidden-3d.com hidden-3d.com hidden-3d.com



Stereogram by 3Dimka

“Seeing” objects

- Almost everything in Java is an object.
- These objects have fields and methods.
- We use the objects, fields, and methods to accomplish our goals.
- Sooner see objects, easier it is to code in Java. Not memorizing anything, using an objects abilities.

Dot Notation

Dot notation

`instance.field` or `instance.method()`
`Class.field` or `Class.method()` (for static)

For example, if we have a class `Student`, and we made the following student:

```
Student jessica = new Student();
```

Then we could write:

```
jessica.goToClass()
```

```
jessica.gpa
```

or

```
Student.numStudents
```

If variable is private, can't use `instance.field`

- That's why we have getters and setters



```
1 public class McDonaldsDriver {
2
3
4     public static void main(String[] args) {
5         //Notice no privacy modifiers because we are inside a method.
6         McDonalds store1 = new McDonalds(true, 0);
7         McDonalds store2 = new McDonalds(true, 0);
8         McDonalds store3 = new McDonalds(true, 0);
9         McDonalds store4 = new McDonalds(true, 0);
10
11         double earnings = store1.annualEarnings; //wrong for private variable - fine for public
12
13         double earnings5 = store1.getAnnualEarnings(); //right
14
15         store1.annualEarnings = 1000; //wrong for private variable - fine for public
16
17         store1.setAnnualEarnings(1000); //right
18     }
19
20 }
21
```


Object oriented thinking

```
# Check the state of the Coffee Pot object
if coffeePot.isEmpty
```

```
# Perform method on the Grinder object
grounds = grinder.grind(bean)
```

```
# Check state of the Coffee Machine object
if coffeeMachine.waterLevel.isEmpty
```

```
# Call a method that changes the water level state
coffeeMachine.add(water)
```

```
# Add the results of the Grinder object to the Coffee Machine object
coffeeMachine.add(grounds)
```

```
coffeeMachine.brew()
```

```
while coffeeMachine.isBrewing()
# Pun-intended
sleep
```

```
mug = cupboard.findCoffeeCup()
while not mug.isFull
coffeePot.pour(mug)
```

Example from prior class

```
If teeth.isDirty
    toothpaste.grab()
    toothbrush.grab()
    toothpaste.squeeze(paste)
    if toothpaste.pasteLevel is empty
        toothpaste.throwAway(garbage)
    system.out.println("note to self: buy more toothpaste")
end

toothpaste.putAway(counter)
toothbrush.brush(upperJawTeeth)
toothbrush.brush(bottomJawTeeth)
facet.turnOn(water)
mouth.rinse(water)
if teeth.isClean
    toothbrush.away(counter)
    mirror.smile()
```

Try it!

- Think of something you do everyday – write it in pseudocode using objects and dot notation.