Model 1 Object Methods

In addition to providing constructors, getters, and setters, classes often provide `equals` and `toString` methods. These methods make it easier to work with objects of the class.

As a team, review the provided *Color.java* and *Point.java* files. Run each program to see how it works. Then answer the following questions using the source code (don't just guess).

Questions (15 min) Start time:

1. Based on the output of *Color.java*, what is the value of each expression below?

```
Color black = new Color();
Color other = new Color(0, 0, 0);
Color gold = new Color(255, 215, 0);
```

a) `black == other`

b) `black == gold`

c) `black.toString()`

d) `black.equals(other)`

e) `black.equals(gold)`

f) `gold.toString()`

2. What is the purpose of the `toString` method?

3. Based on the output of *Point.java*, what is the value of each expression below?

```
Point p1 = new Point();
Point p2 = new Point(0, 0);
Point p3 = new Point(3, 3);
```

a) `p1 == p2`

b) `p1.toString()`

c) `p3.toString()`

d) `p1.equals(p2)`

e) `p1.equals("(0, 0)")`

f) `p3.equals("(3, 3)")`

4. What is the purpose of the `equals` method?

5. Examine *Point.java* again. What is the purpose of the `if`-statement in the `equals` method?

6. How could you modify the `equals` method to cause both #3e and #3f to return `true`?