Static

Methods, Attributes

YEGOR BUGAYENKO

Lecture #2 out of 8 90 minutes

All visual and text materials presented in this slidedeck are either originally made by the author or taken from public Internet sources, such as website. Copyright belongs to their respected authors.

Methods Attributes

Methods Attributes 3/11



What static methods are for?

```
class Circle {
  public float radius;
}

class Circle {
  public float radius;
  float square() {
    return radius * radius * 3.14;
    return c.radius * c.radius * 3.14;
  }
}
```

Most notable Java examples: FileUtils, IOUtils, and StringUtils from Apache Commons; Files from JDK7; Iterators from Google Guava. Read this.

Methods Attributes

[Purpose Problems Coupling Eagerness Cohesion]

What's wrong with "Utils"?

- 1) They are unbreakable dependencies
- 2) They are eager, not lazy
- 3) They are not cohesive

Tight Coupling

```
void paintIt(Circle c) {
  float s = GeometryUtils.calcSquare(c);
  float p = s * 5.55;
  // paint it using the "p"
  }
  void paintIt(Circle c) {
    float s = c.square();
    float p = s * 5.55;
    // paint it using the "p"
  }
}
```

Which snippet is easier to test? Try to write a test for the first one, expecting s to be equal to 42.0. Read this.

Imperative, not Declarative

```
void paintIt(Circle c) {
  float s = GeometryUtils.calcSquare(c);
  if (t) { return; }
  float p = s * 5.55;
  // paint it using the "p"
  }

void paintIt(Circle c) {
  float s = new SquareOf(c);
  if (t) { return; }
  float p = s * 5.55;
  // paint it using the "p"
  }

// paint it using the "p"
```

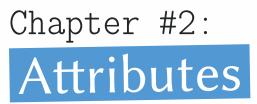
Which snippet is more eager to calculate the square of the circle? Which one does it when it's <u>really</u> necessary? Read <u>this</u>.

Low Cohesion

```
class GeometryUtils {
    static float calcSquare(Circle c);
    static float calcPerimeter(Circle c);
    static float calcSinus(Angle a);
    static float calcCosinus(float s);
    // and many more...
    }
    class Circle {
        float square();
        float perimeter();
    }
    class Angle {
        float sinus();
    }
    class Float {
        float cosinus();
    }
}
```

Which class looks more cohesive to you, the utility class GeometryUtils or the Circle?

Methods Attributes 9/11



Public literals

We must solve the problem of functionality duplication, not just data duplication. Read this.

Singletons

```
class Canvas {
  public static Canvas INSTANCE =
    new Canvas();
  private Canvas() {}
  public void addCircle(Circle c);
}
Canvas.INSTANCE.addCircle(c1);
Canvas.INSTANCE.addCircle(c2);
```

Read this.

```
c = new Canvas();
c.addCircle(c1);
c.addCircle(c2);
```