

Static Methods

By: Megan Avery
Updated July 2018



Before We Begin

- Make sure you have Java downloaded on your computer
 - can run **java -version** in terminal to check
- Make sure you have IntelliJ downloaded on your computer
- Download figure.txt onto your Desktop

- *Suggested:* Watch previous Java tutorials

Static Methods

Vocab Alert!

algorithm: A list of steps for solving a problem

structured algorithm: a solution split into coherent tasks

static method: A named group of statements

procedural decomposition: process of dividing a problem into methods

Algorithm Exercise

Take a few minutes and write an algorithm for baking and icing a cake

Example Solution: Baking a Cake

1. Mix the dry ingredients.
2. Cream the butter and sugar.
3. Beat in the eggs.
4. Stir in the dry ingredients.
5. Preheat the oven.
6. Grease cake pans
7. Pour cake batter into pans
8. Put cake in oven
9. Set the timer.
10. Allow the cake to bake.
11. Mix ingredients for frosting.
12. Take cake out of oven
13. Put icing on cake

Problems with Algorithms

lack of structure: many small steps; tough to remember

redundancy: consider making 2 cakes in sequence. You wouldn't want to have to resay all the steps over again

Making Our Algorithm Structured

1. Make Cake Batter
 - a. Mix the dry ingredients.
 - b. Cream the butter and sugar.
 - c. Beat in the eggs.
 - d. Stir in the dry ingredients.
2. Prepare Cake to Bake
 - a. Preheat the oven.
 - b. Grease cake pans
 - c. Pour cake batter into pans
3. Bake the Cake
 - a. Put cake in oven
 - b. Set the timer.
 - c. Allow the cake to bake.
4. Decorate Cake
 - a. Mix ingredients for frosting.
 - b. Take cake out of oven
 - c. Put icing on cake

How Structure Removes Redundancy

1. Make Cake Batter
- 2a. Prepare Cake to Bake (*cake #1*)
- 2b. Prepare Cake to Bake (*cake #2*)
3. Bake the Cake
4. Decorate Cake

Structure Leads to Methods

If we were going to write a program to bake a cake each place we split the algorithm to create structure would become its own method.

So, we would end up with a methods for:

1. Make Cake Batter
2. Prepare Cake to Bake
3. Bake the Cake
4. Decorate Cake

Syntax for a static method

```
// method declaration  
public static void methodName() {  
    // code goes here  
}
```

```
public static void main(String[] args) {  
    // calling method  
    methodName();  
}
```

Note on Method Names

Method names should follow the same convention as variable names:

- start with lowercase letter
- can contain letters and numbers
- follow camel case style
 - ex: `methodName`

Steps for Using Methods

1. Design the algorithm.
 - a. Look at the structure, and which commands are repeated.
 - b. Decide what are the important overall tasks.
 - c. Good programmers do this BEFORE writing any code
2. Declare (write down) the methods.
 - a. Arrange statements into groups and give each group a name.
3. Call (run) the methods.
 - a. The program's main method executes the other methods to perform the overall task.

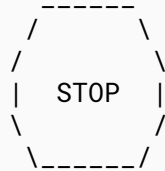
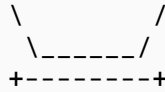
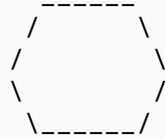
Important Points About Methods

Methods can call other methods

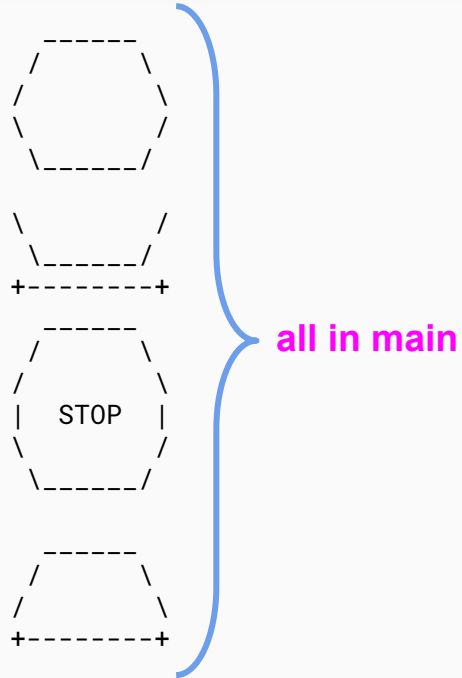
Methods can be called as many times
as needed/wanted

Static Method Walkthrough

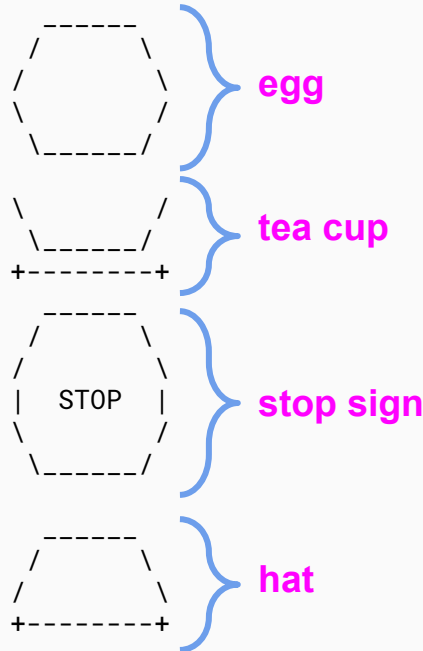
Write a program to print out this figure



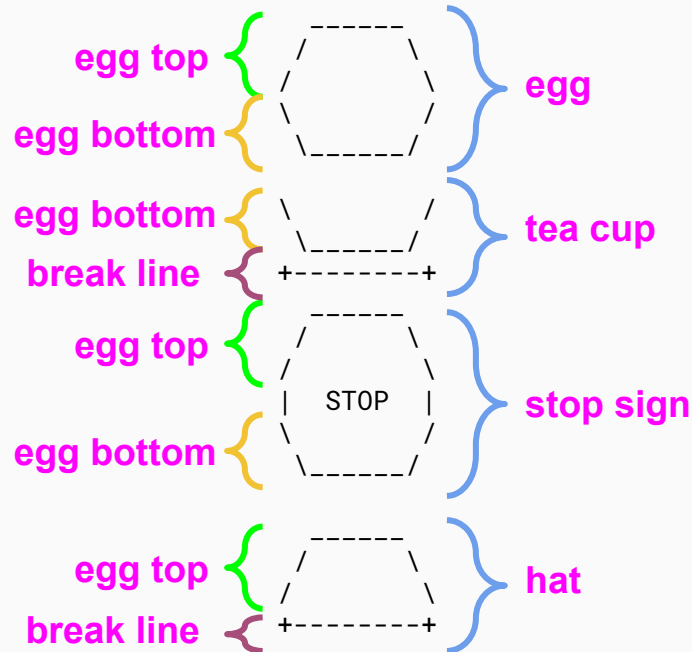
Unstructured Solution



Structured Solution With Redundancy



Structured Solution Without Redundancy



Practice for Later:

12 Days of Christmas Program

Write a program that uses static methods to print out the “[The 12 Days of Christmas](#)” song in such a way that no line is printed using a `System.out.println` call more than once.

Things to consider:

- What to make a method
- When to call the methods

The End