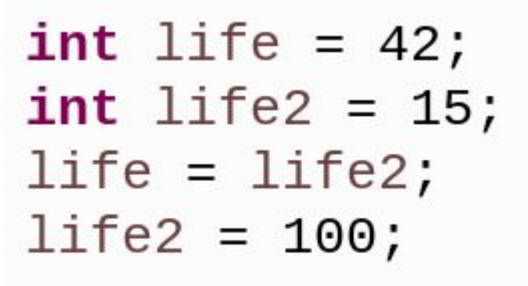
“Before you marry a person you should first make them use a computer with slow internet to see who they really are.”

– Will Ferrell

METHODS

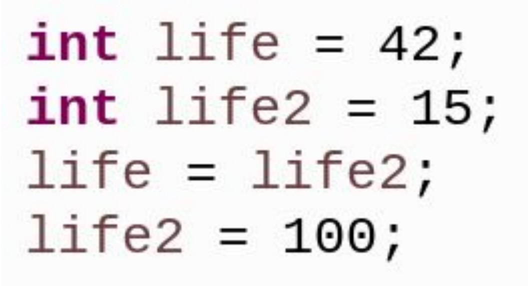
But First...

what do you think this does?

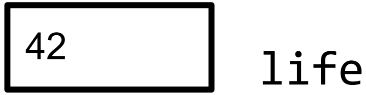


```
int life = 42;
int life2 = 15;
life = life2;
life2 = 100;
```

what do you think this does?



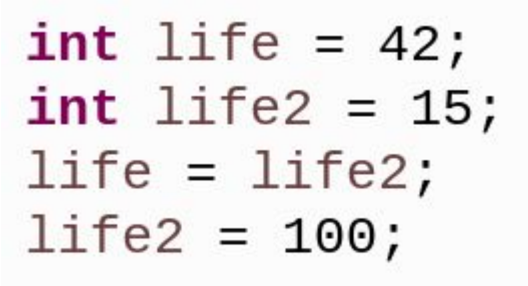
```
int life = 42;
int life2 = 15;
life = life2;
life2 = 100;
```



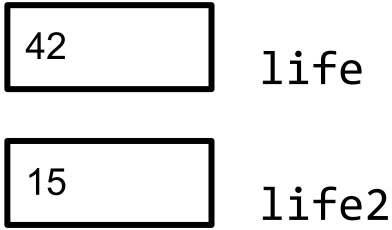
42

life

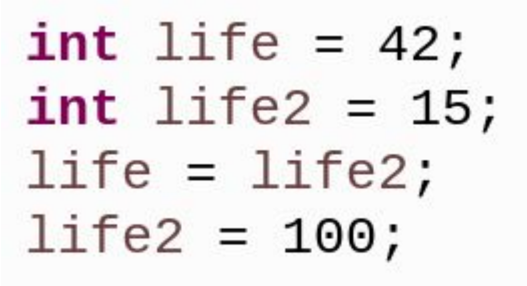
what do you think this does?



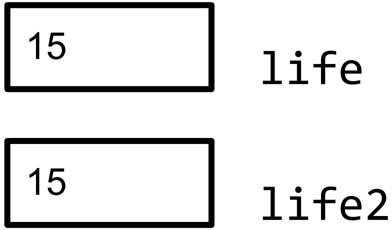
```
int life = 42;
int life2 = 15;
life = life2;
life2 = 100;
```



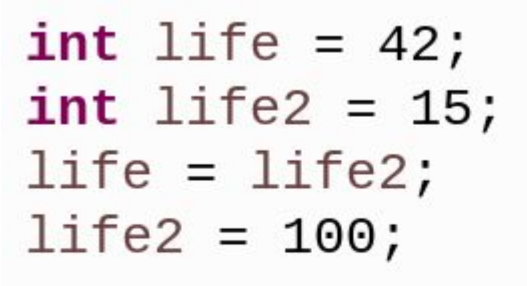
what do you think this does?



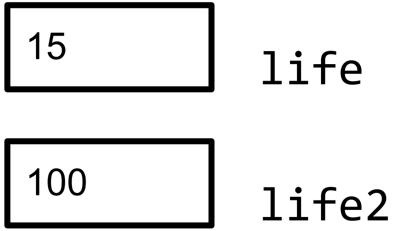
```
int life = 42;
int life2 = 15;
life = life2;
life2 = 100;
```



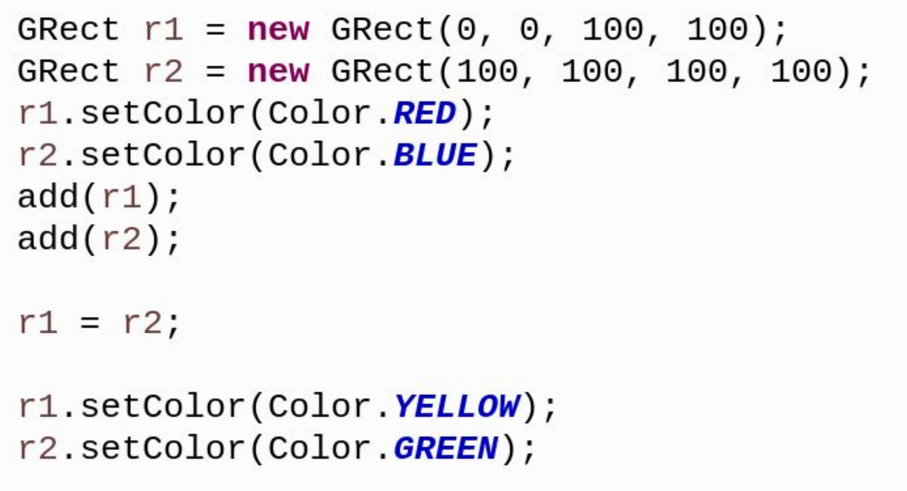
what do you think this does?



```
int life = 42;
int life2 = 15;
life = life2;
life2 = 100;
```



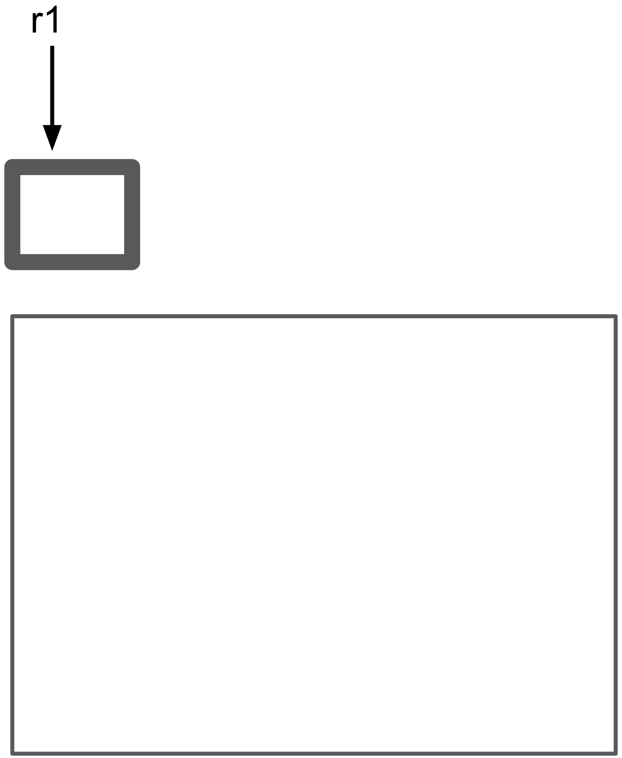
what do you think this does?



```
GRect r1 = new GRect(0, 0, 100, 100);
GRect r2 = new GRect(100, 100, 100, 100);
r1.setColor(Color.RED);
r2.setColor(Color.BLUE);
add(r1);
add(r2);

r1 = r2;

r1.setColor(Color.YELLOW);
r2.setColor(Color.GREEN);
```

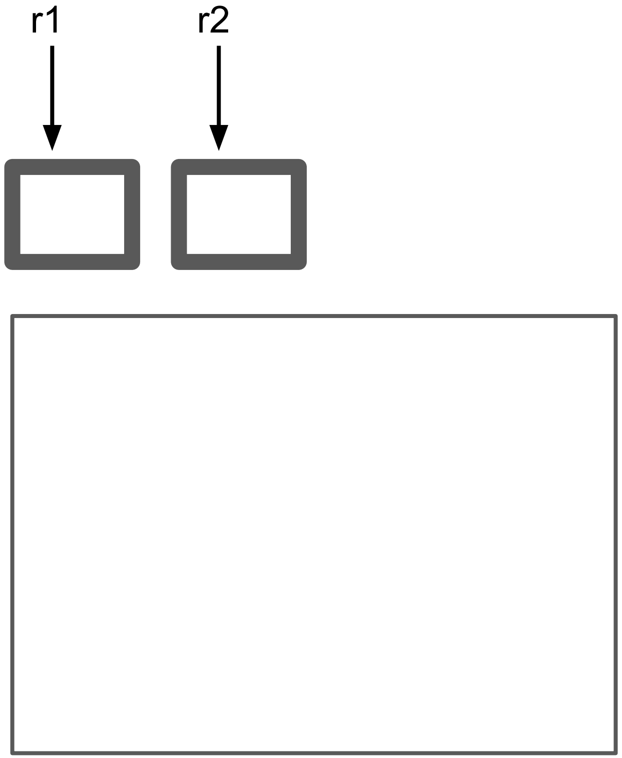
what do you think this does?

```
GRect r1 = new GRect(0, 0, 100, 100);
GRect r2 = new GRect(100, 100, 100, 100);
r1.setColor(Color.RED);
r2.setColor(Color.BLUE);
add(r1);
add(r2);

r1 = r2;

r1.setColor(Color.YELLOW);
r2.setColor(Color.GREEN);
```



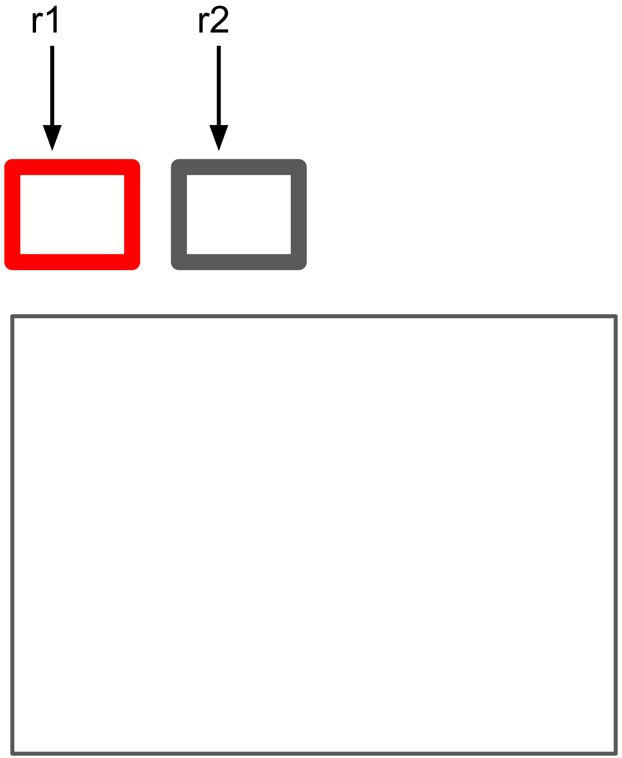
what do you think this does?

```
GRect r1 = new GRect(0, 0, 100, 100);
GRect r2 = new GRect(100, 100, 100, 100);
r1.setColor(Color.RED);
r2.setColor(Color.BLUE);
add(r1);
add(r2);

r1 = r2;

r1.setColor(Color.YELLOW);
r2.setColor(Color.GREEN);
```



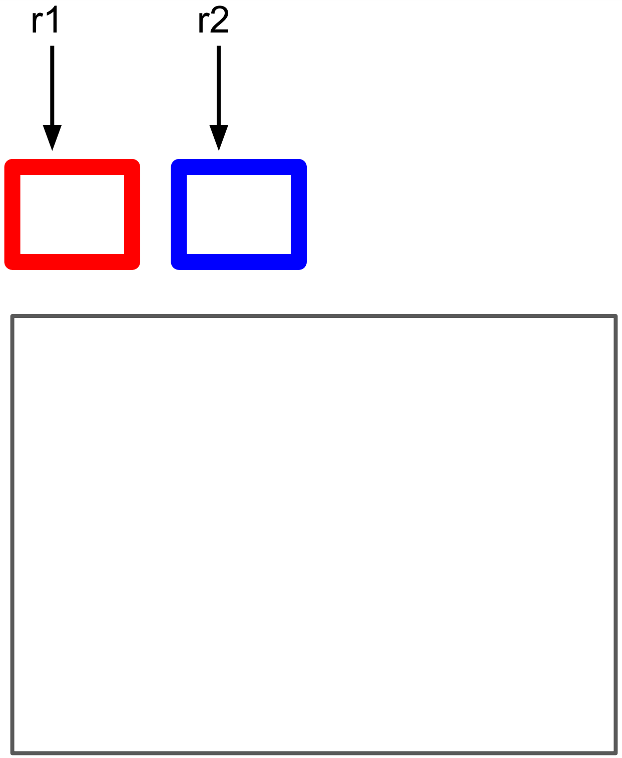
what do you think this does?

```
GRect r1 = new GRect(0, 0, 100, 100);
GRect r2 = new GRect(100, 100, 100, 100);
r1.setColor(Color.RED);
r2.setColor(Color.BLUE);
add(r1);
add(r2);

r1 = r2;

r1.setColor(Color.YELLOW);
r2.setColor(Color.GREEN);
```

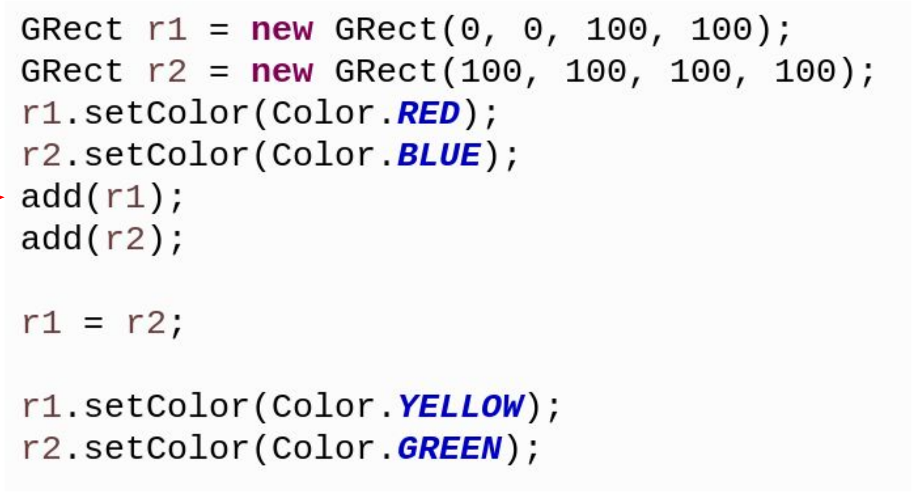


what do you think this does?

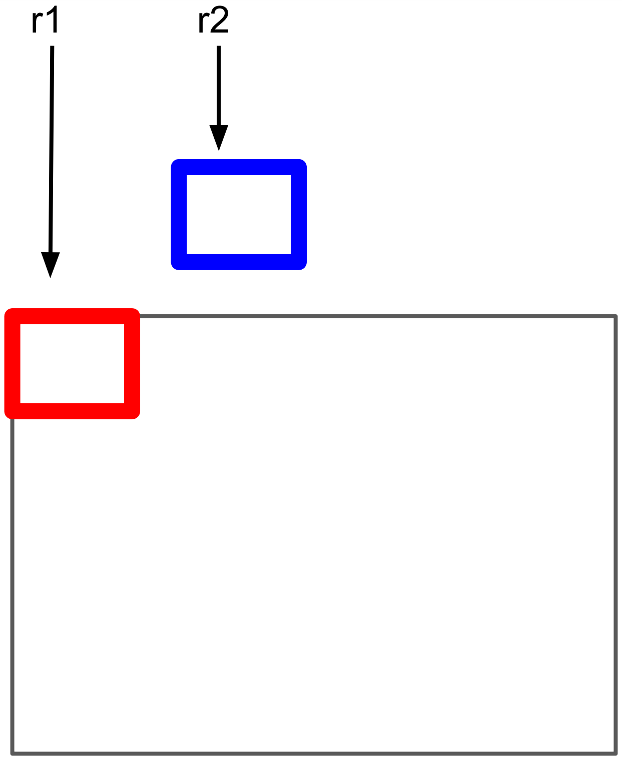
```
GRect r1 = new GRect(0, 0, 100, 100);
GRect r2 = new GRect(100, 100, 100, 100);
r1.setColor(Color.RED);
r2.setColor(Color.BLUE);
add(r1);
add(r2);

r1 = r2;

r1.setColor(Color.YELLOW);
r2.setColor(Color.GREEN);
```



what do you think this does?

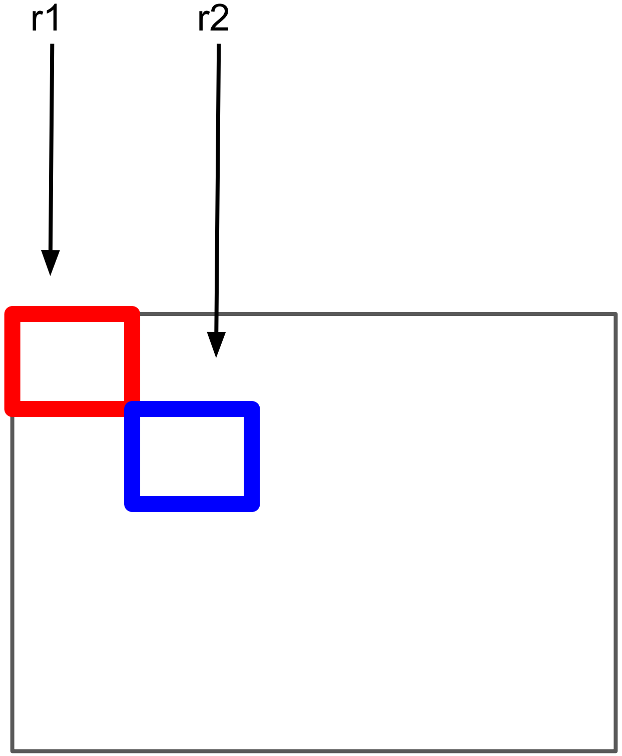


```
GRect r1 = new GRect(0, 0, 100, 100);
GRect r2 = new GRect(100, 100, 100, 100);
r1.setColor(Color.RED);
r2.setColor(Color.BLUE);
add(r1);
add(r2);

r1 = r2;

r1.setColor(Color.YELLOW);
r2.setColor(Color.GREEN);
```



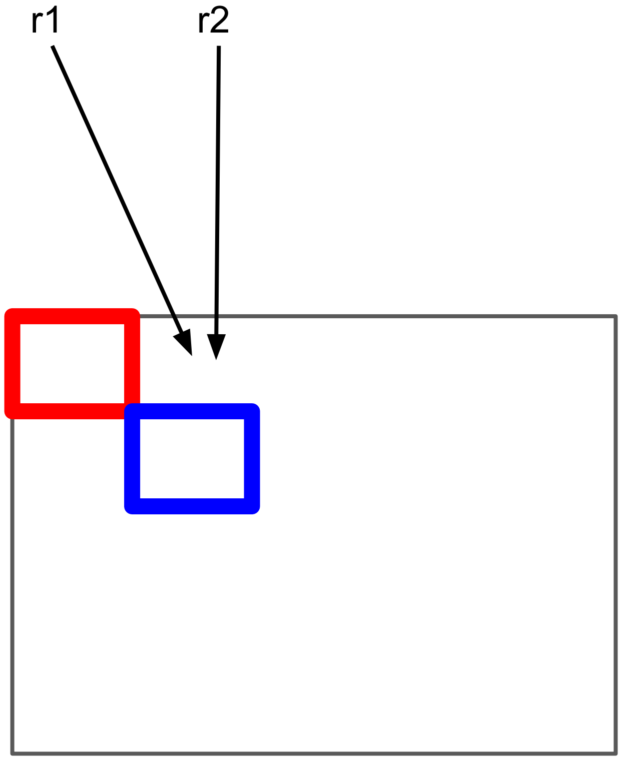
what do you think this does?

```
GRect r1 = new GRect(0, 0, 100, 100);
GRect r2 = new GRect(100, 100, 100, 100);
r1.setColor(Color.RED);
r2.setColor(Color.BLUE);
add(r1);
add(r2);

r1 = r2;

r1.setColor(Color.YELLOW);
r2.setColor(Color.GREEN);
```



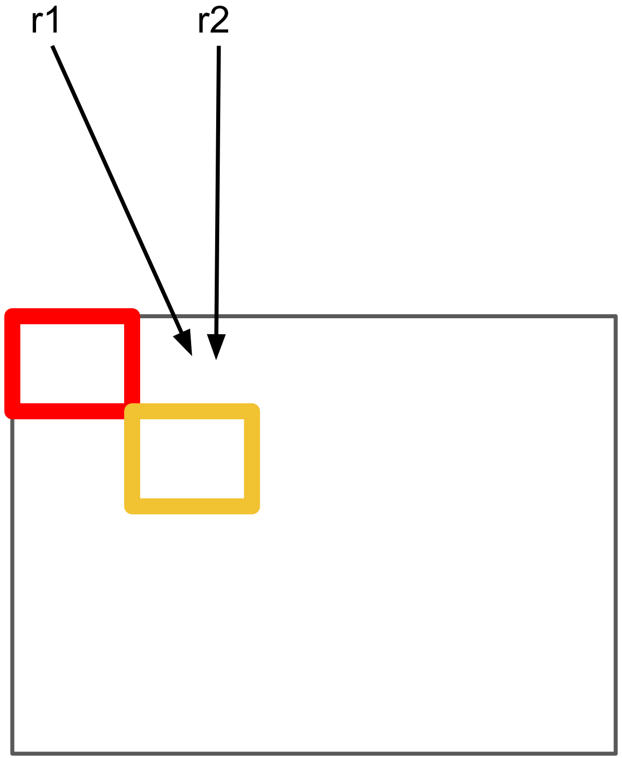
what do you think this does?

```
GRect r1 = new GRect(0, 0, 100, 100);
GRect r2 = new GRect(100, 100, 100, 100);
r1.setColor(Color.RED);
r2.setColor(Color.BLUE);
add(r1);
add(r2);

r1 = r2;

r1.setColor(Color.YELLOW);
r2.setColor(Color.GREEN);
```



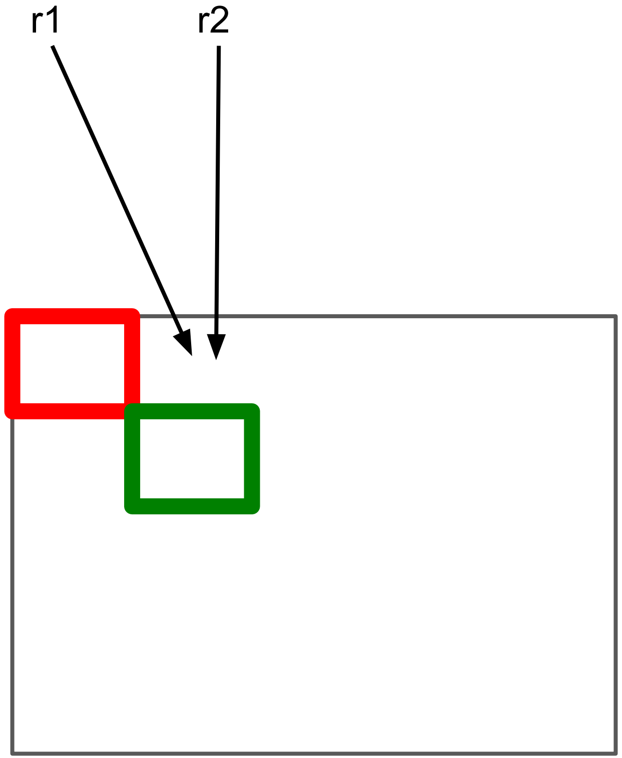
what do you think this does?

```
GRect r1 = new GRect(0, 0, 100, 100);
GRect r2 = new GRect(100, 100, 100, 100);
r1.setColor(Color.RED);
r2.setColor(Color.BLUE);
add(r1);
add(r2);

r1 = r2;

r1.setColor(Color.YELLOW);
r2.setColor(Color.GREEN);
```



what do you think this does?

```
GRect r1 = new GRect(0, 0, 100, 100);
GRect r2 = new GRect(100, 100, 100, 100);
r1.setColor(Color.RED);
r2.setColor(Color.BLUE);
add(r1);
add(r2);

r1 = r2;

r1.setColor(Color.YELLOW);
r2.setColor(Color.GREEN);
```



METHODS

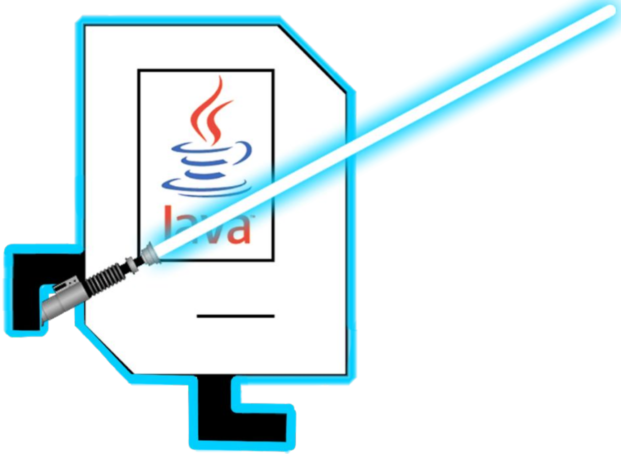
Let's feel the force

You have seen methods before

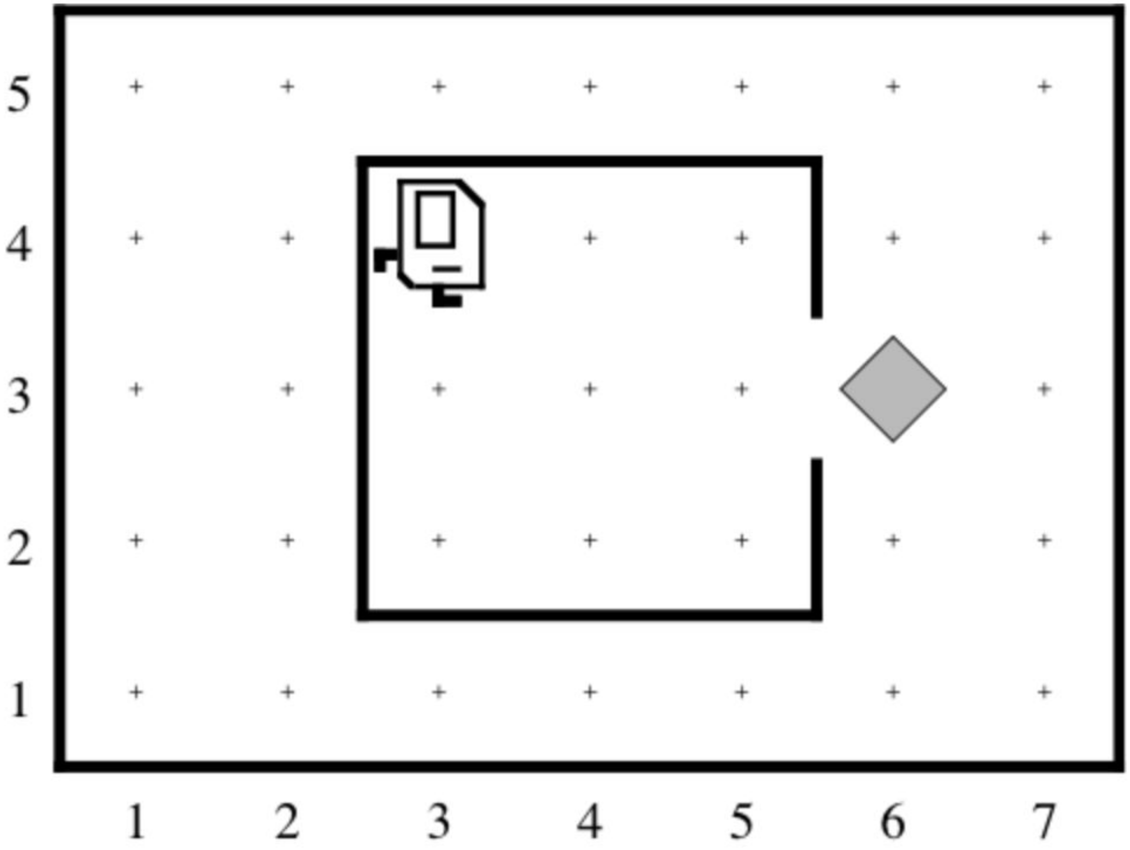
```
turnRight();

move();
readInt(“Int please! ”);

println(“hello world”);
rect.getX();

getWidth()

rect.setLocation(10, 20);
```

Karel and his methods

```
private void turnRight() {
   turnLeft();
   turnLeft();
   turnLeft();
}
```



Decomposition



Collect Newspaper

=

Exit House

then

PickUpPaper

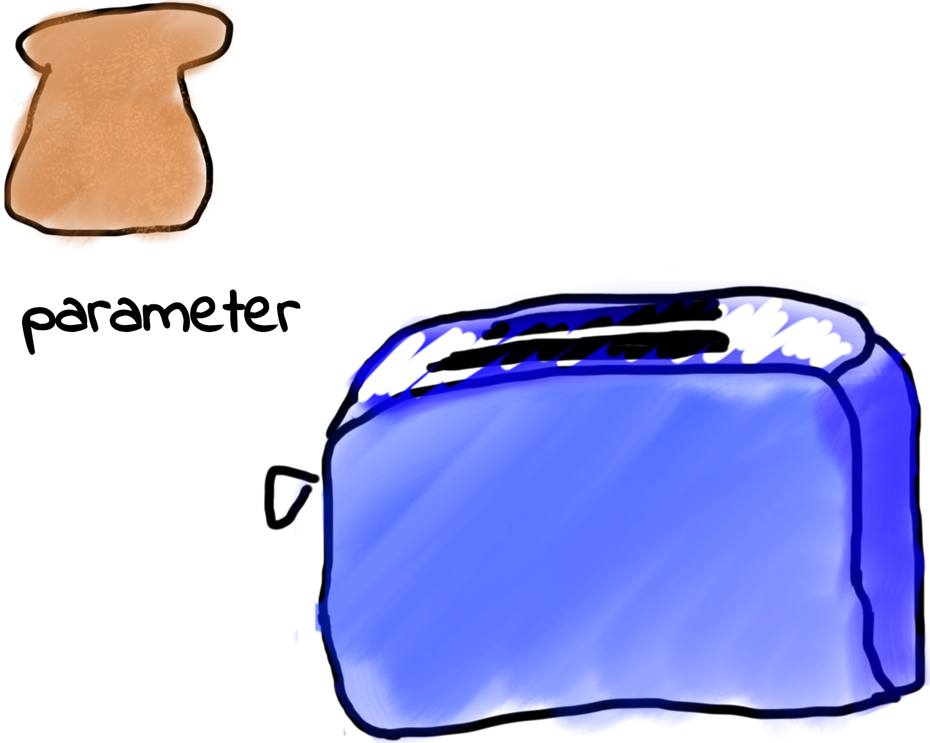
then

ReturnHome

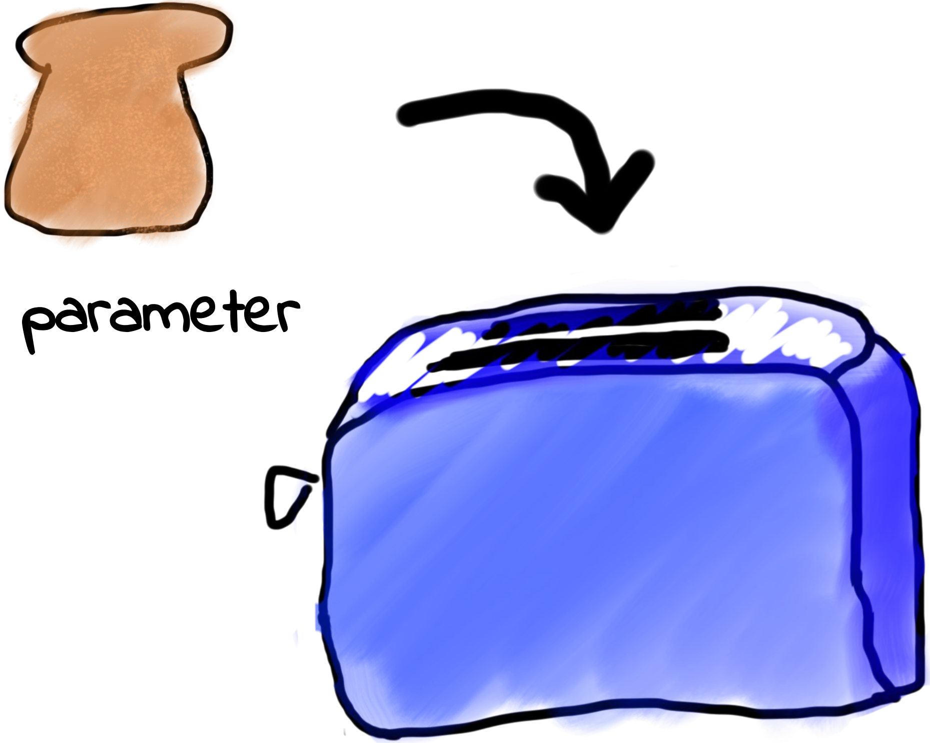
what is method?

Methods are Like Toasters

Methods are Like Toasters

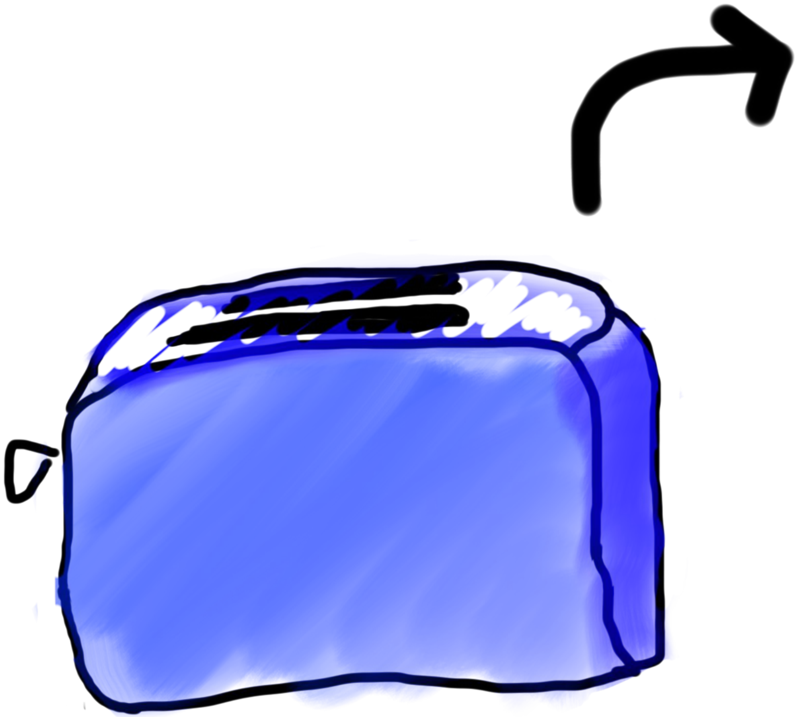


Methods are Like Toasters

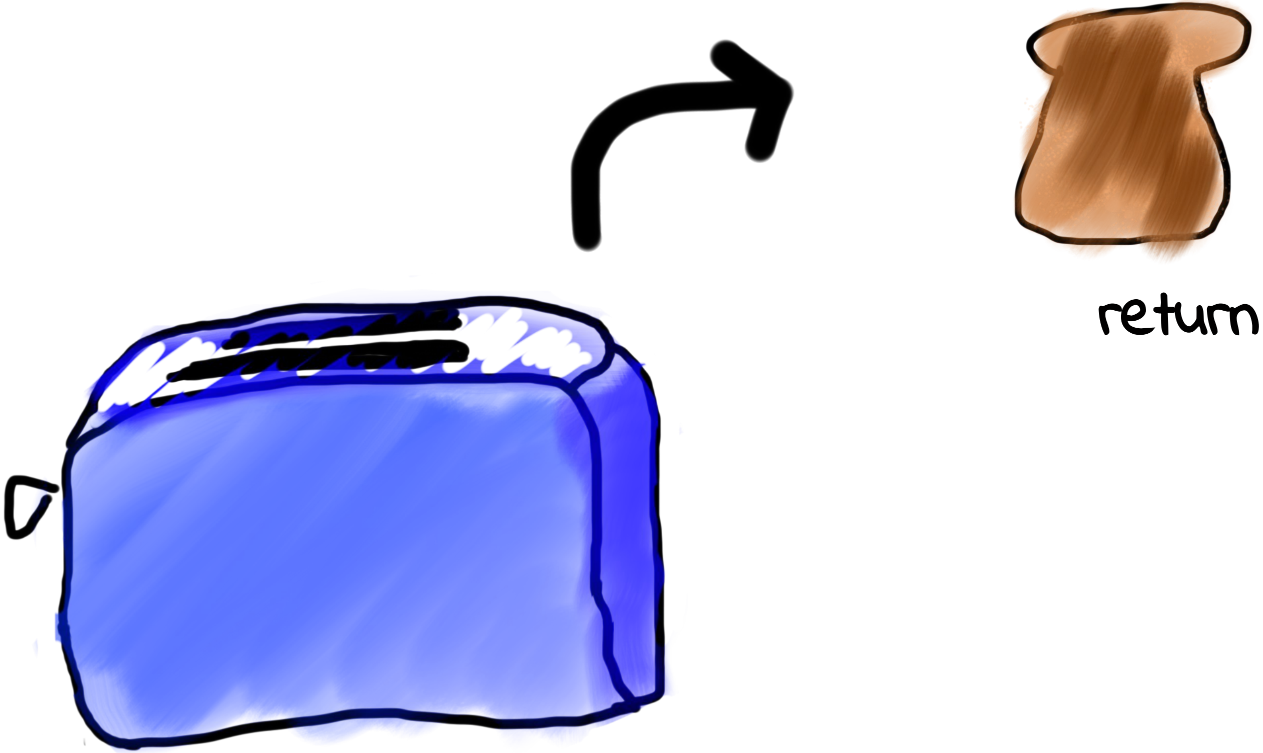


Methods are Like Toasters

Methods are Like Toasters



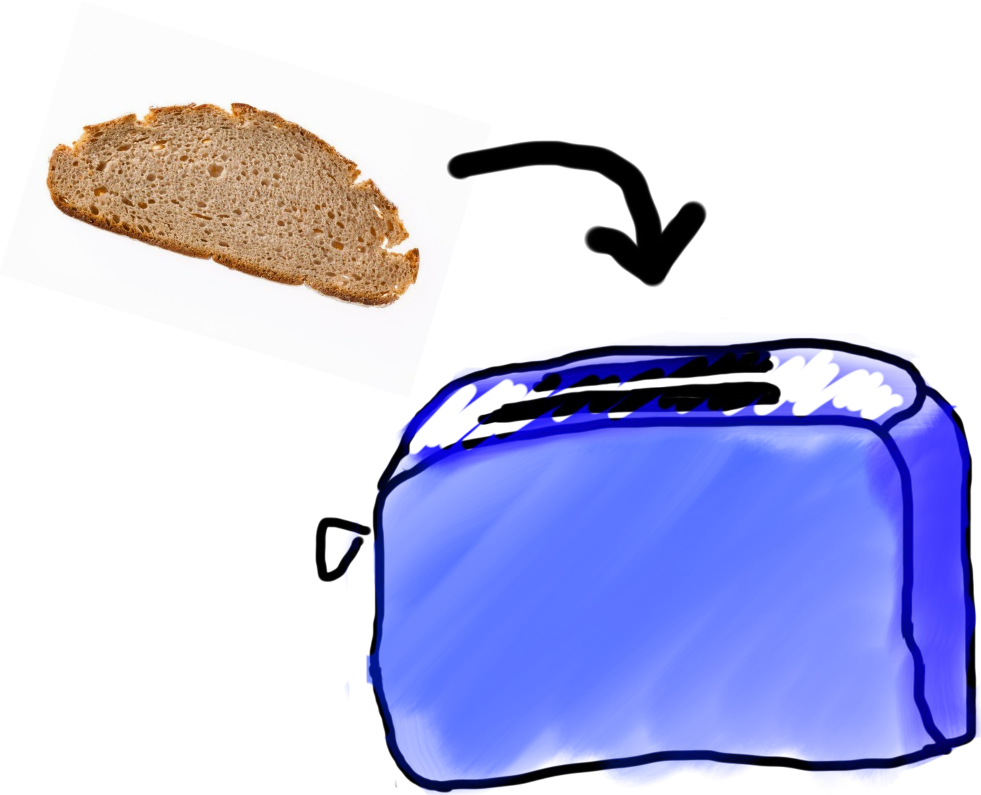
Methods are Like Toasters



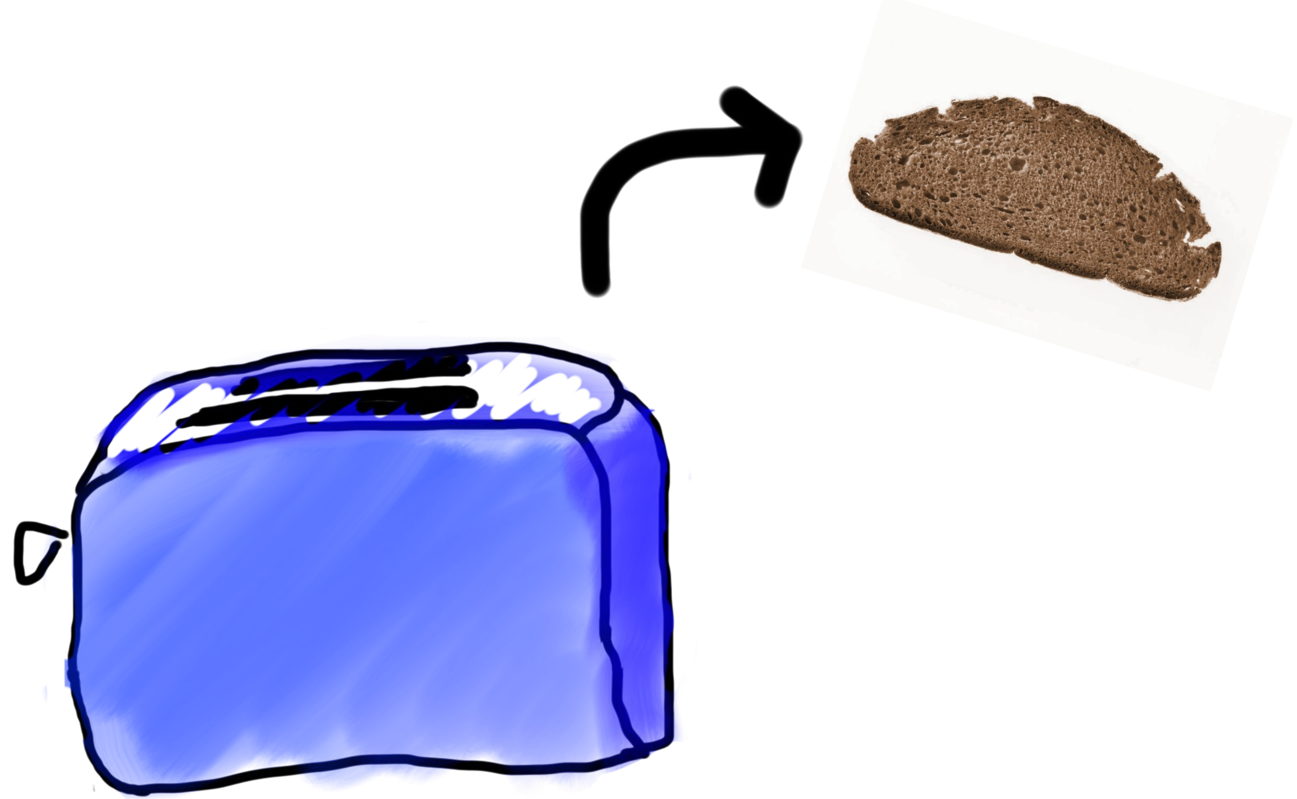
Methods are Like Toasters

Methods are Like Toasters

Methods are Like Toasters



Methods are Like Toasters

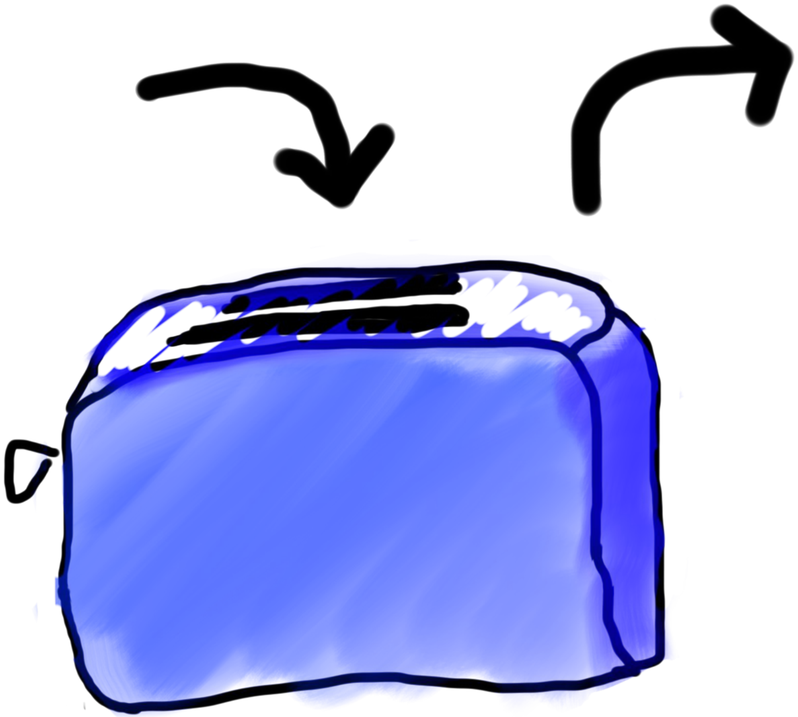


Methods are Like Toasters

THE ORIGINAL
BECHEROVKA
Jan Becher
CZECH REPUBLIC
18 07
HERBAL LIQUEUR
parameter

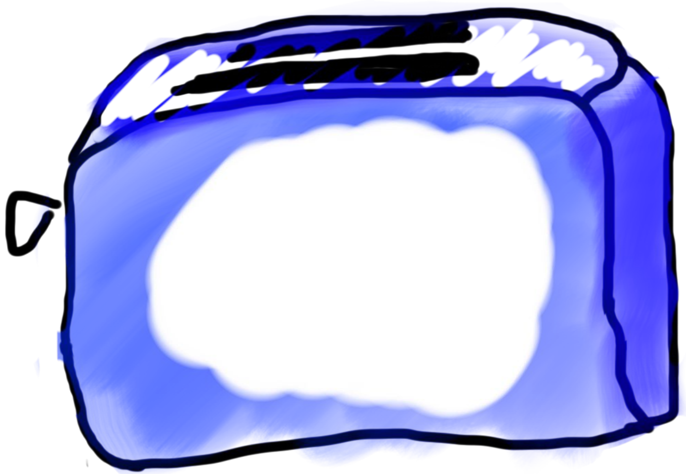
Methods are Like Toasters

parameter



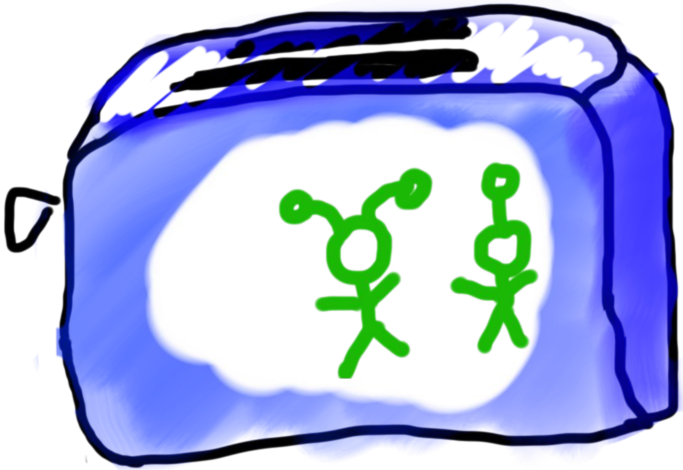
Not every input works

Methods are Like Toasters

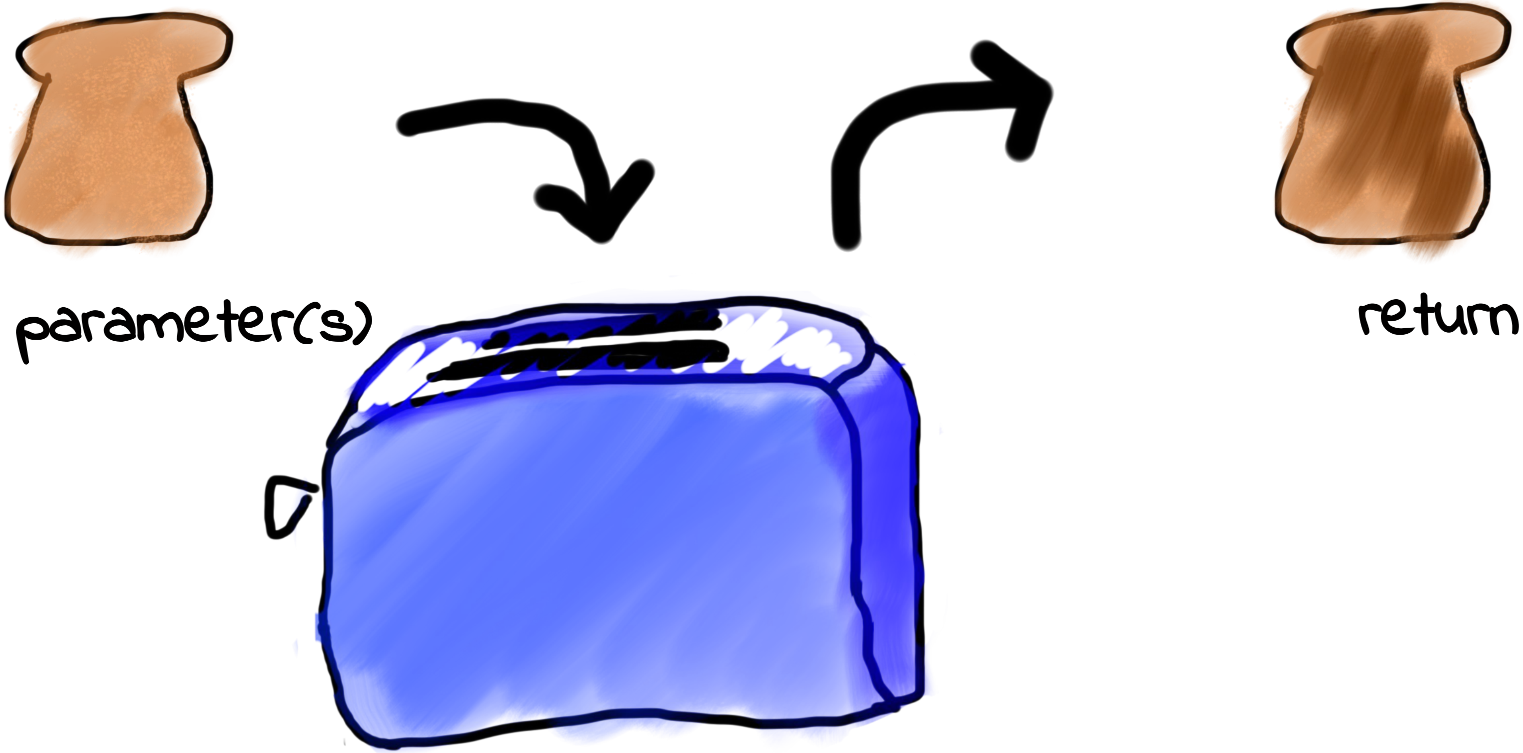


Methods are Like Toasters

Methods are Like Toasters



Methods are Like Toasters

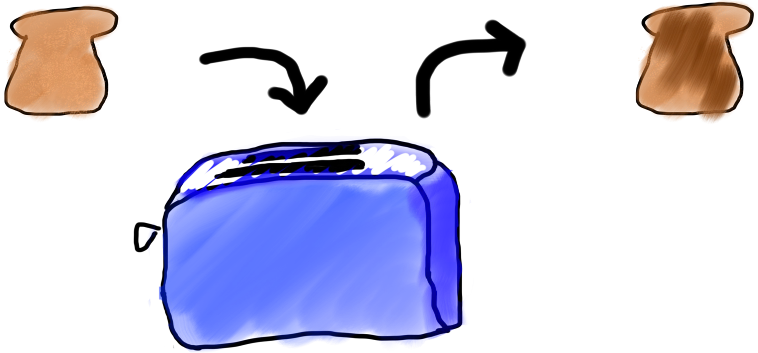


Anatomy of a method

```
public void run() {
    double mid = average(5.0, 10.2);
    println(mid);
}



private double average(double a, double b) {
    double sum = a + b;
    return sum / 2;
}
```

Anatomy of a method

```
public void run() {
    double mid = average(5.0, 10.2);
    println(mid);
}
```



Return type
(Output expected)

Parameters (Input expected)

```
private double average(double a, double b) {
    double sum = a + b;
    return sum / 2;
}
```

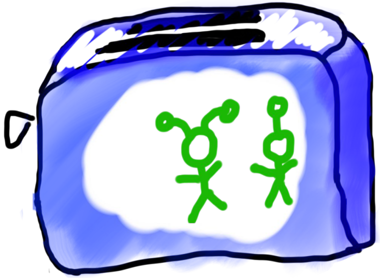
Anatomy of a method

```
public void run() {
    double mid = average(5.0, 10.2);
    println(mid);
}
```

name

```
private double average(double a, double b) {
    double sum = a + b;
    return sum / 2;
}
```

Anatomy of a method



```
public void run() {
    double mid = average(5.0, 10.2);
    println(mid);
}



private double average(double a, double b) {
    double sum = a + b;
    return sum / 2;
}
```

body

Anatomy of a method

```
public void run() {
    double mid = average(5.0, 10.2);
    println(mid);
}



private double average(double a, double b) {
    double sum = a + b;
    return sum / 2;
}
```

return

Anatomy of a method

```
public void run() {
    double mid = average(5.0, 10.2);
    println(mid);
}



private double average(double a, double b) {
    double sum = a + b;
    return sum / 2;
}
```

method "call"

Anatomy of a method

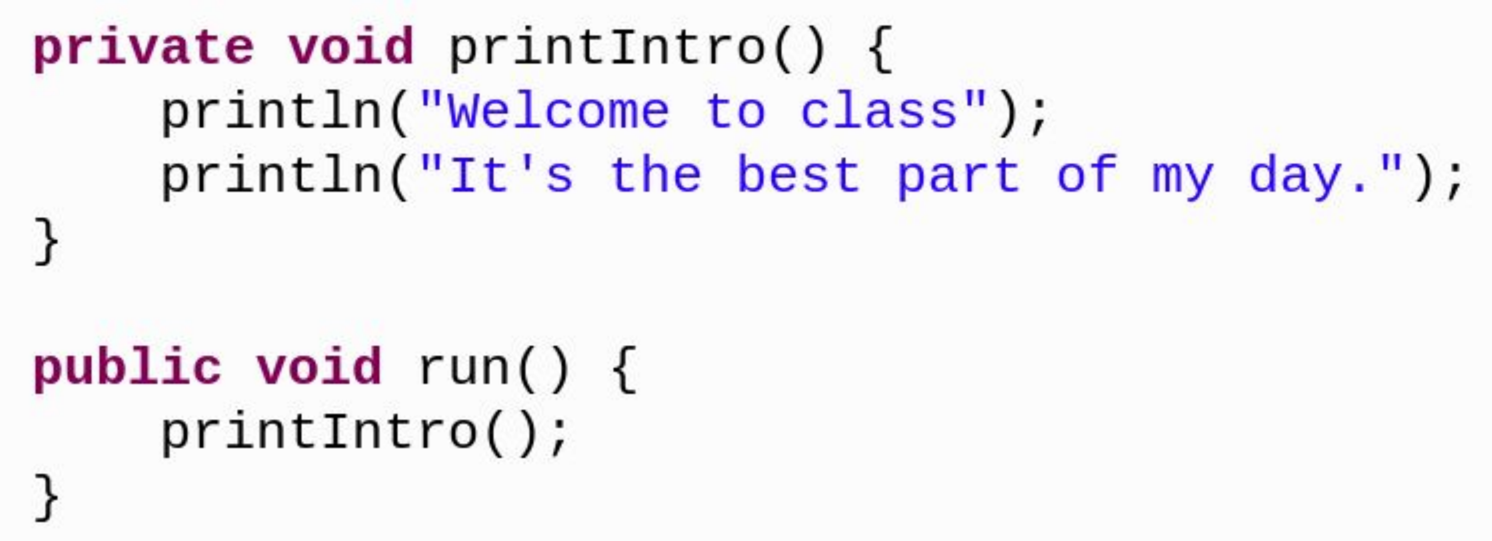
```
public void run() {
    double mid = average(5.0, 10.2);
    println(mid);
}
```

arguments

```
private double average(double a, double b) {
    double sum = a + b;
    return sum / 2;
}
```

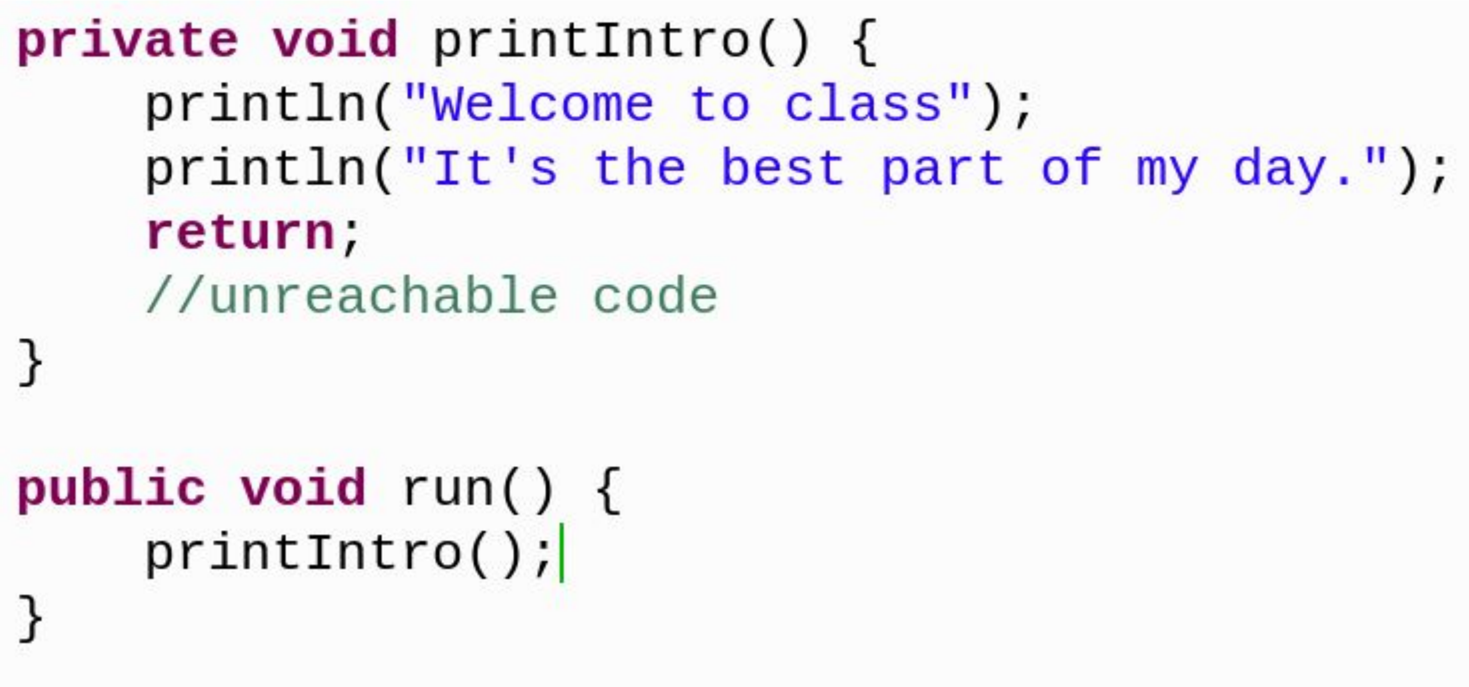
let's practise...

Example 1 - void method



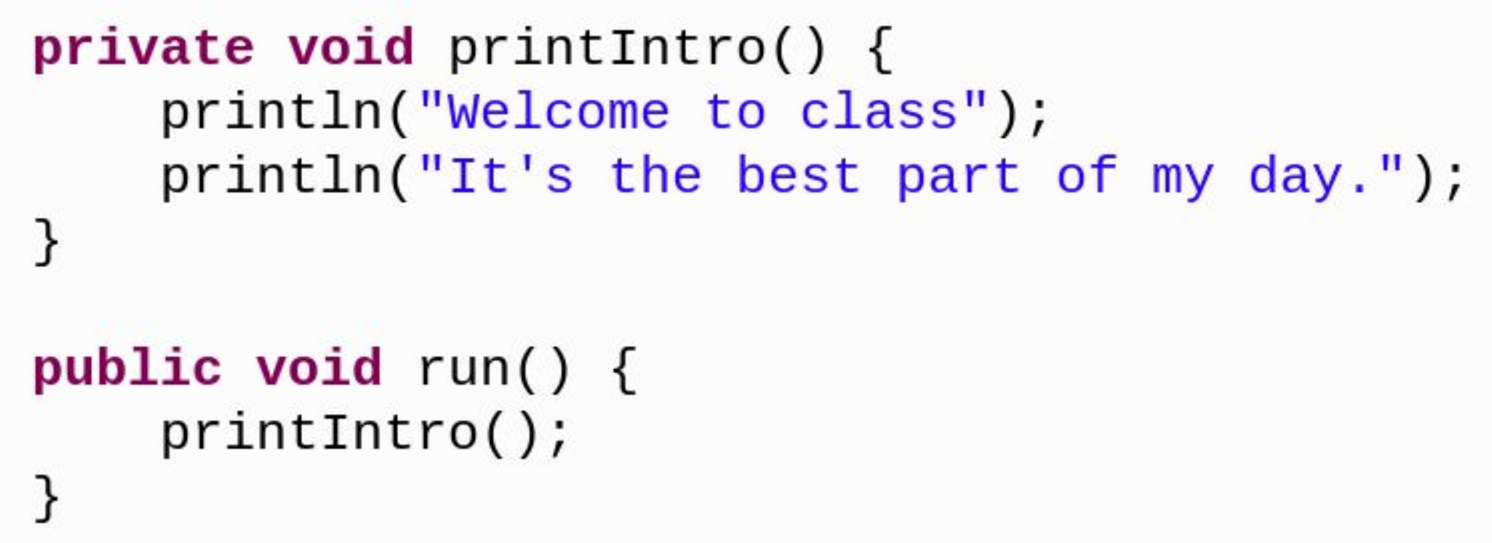
```
private void printIntro() {
    println("Welcome to class");
    println("It's the best part of my day.");
}

public void run() {
    printIntro();
}
```

Example 1 - return statement



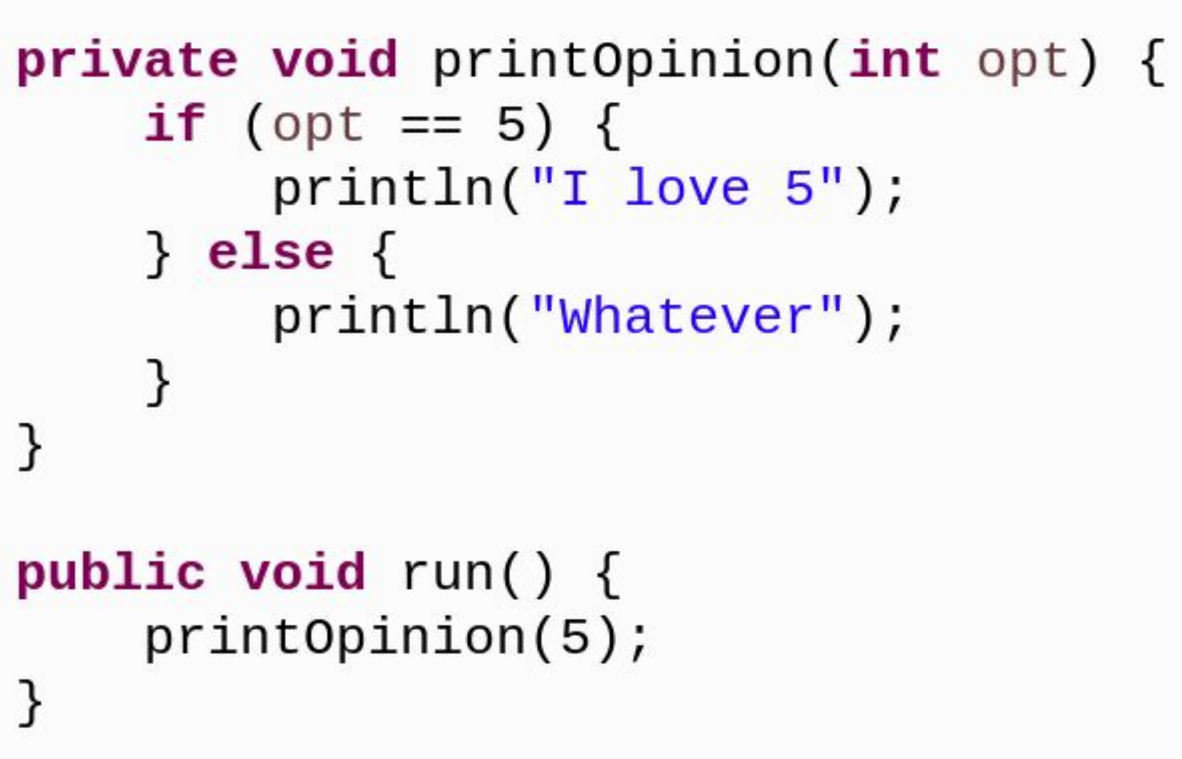
```
private void printIntro() {
    println("Welcome to class");
    println("It's the best part of my day.");
    return;
    //unreachable code
}

public void run() {
    printIntro();
}
```

Example 1 - void method



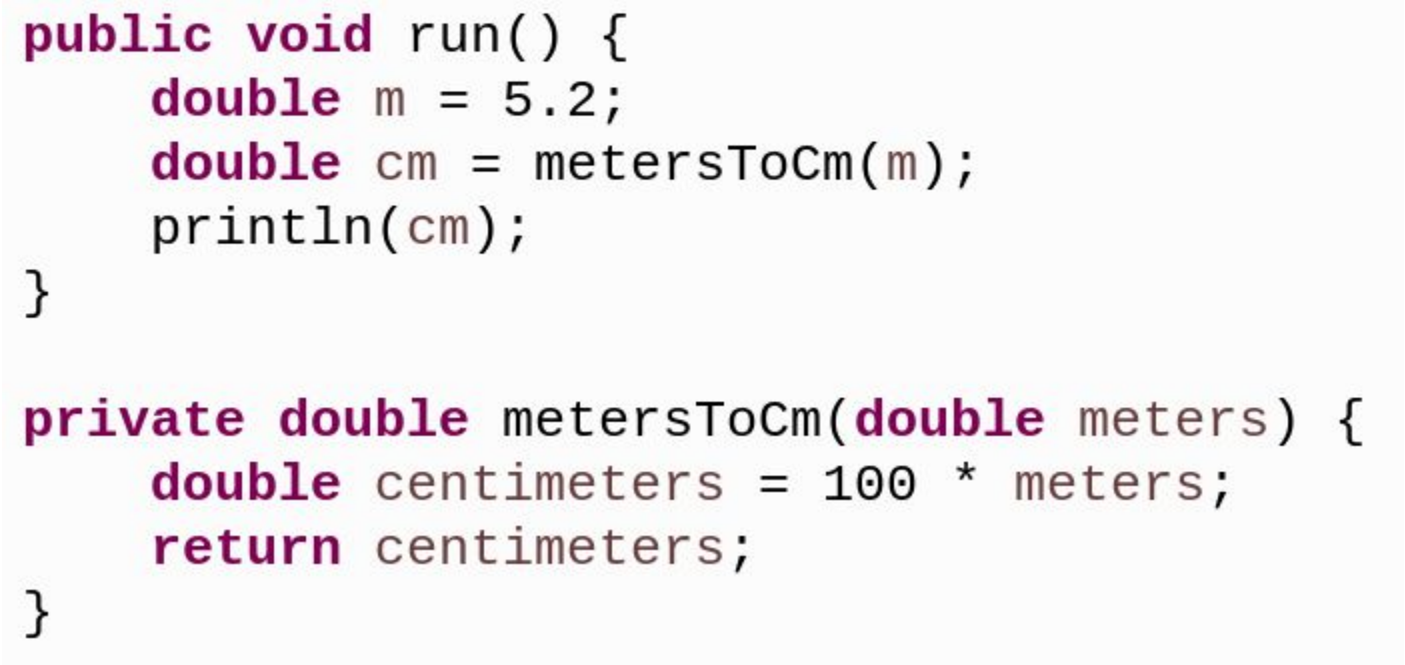
```
private void printIntro() {
    println("Welcome to class");
    println("It's the best part of my day.");
}

public void run() {
    printIntro();
}
```

Example 2 - Parameters



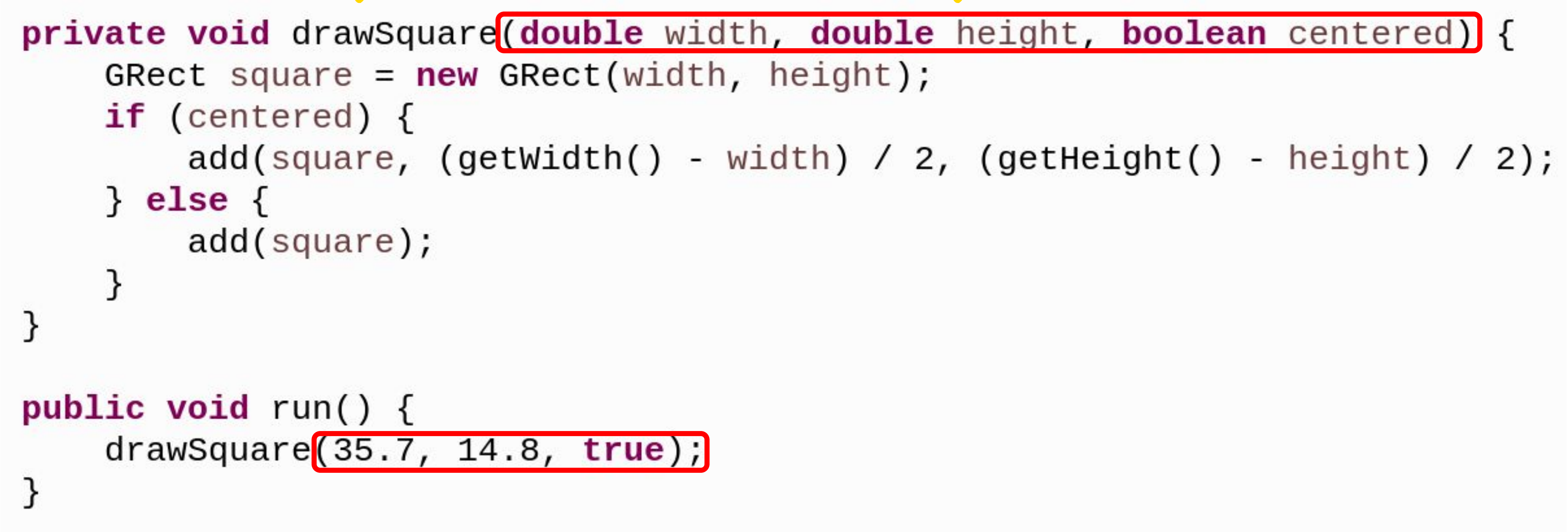
```java
private void printOpinion(int opt) {
    if (opt == 5) {
        println("I love 5");
    } else {
        println("Whatever");
    }
}

public void run() {
    printOpinion(5);
}
```

Example 3 - return statement



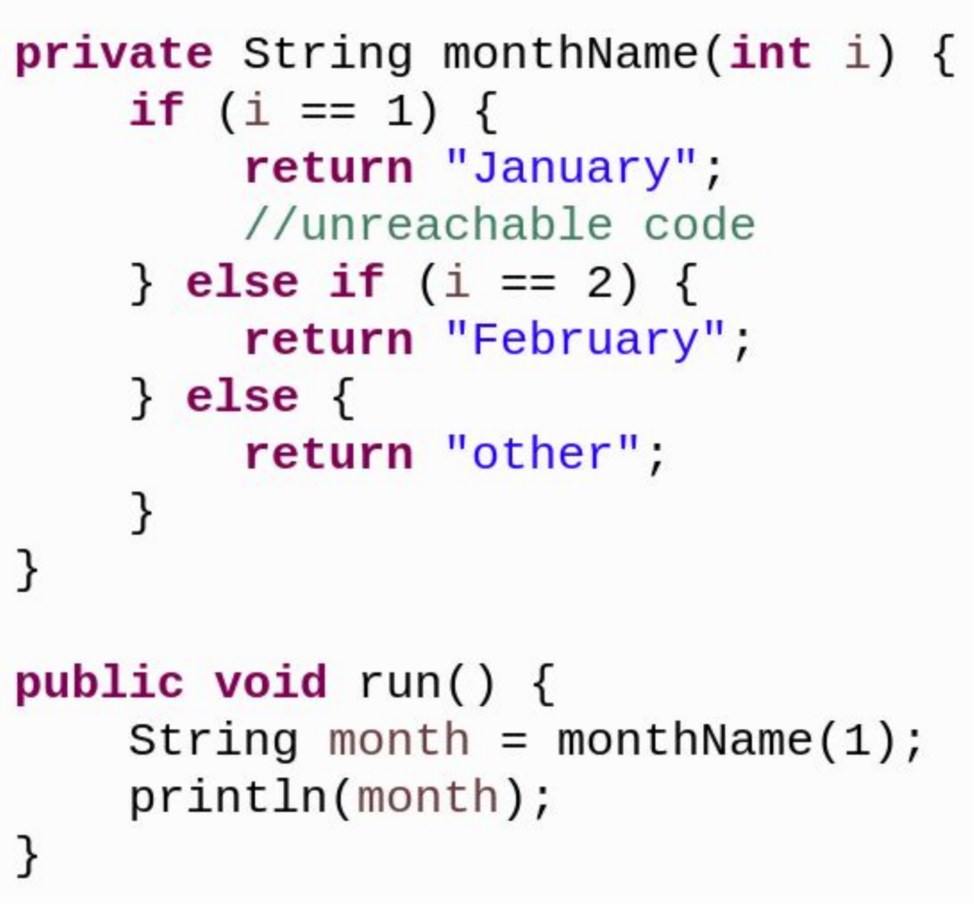
```java
public void run() {
    double m = 5.2;
    double cm = metersToCm(m);
    println(cm);
}

private double metersToCm(double meters) {
    double centimeters = 100 * meters;
    return centimeters;
}
```

Example 4 - more parameters



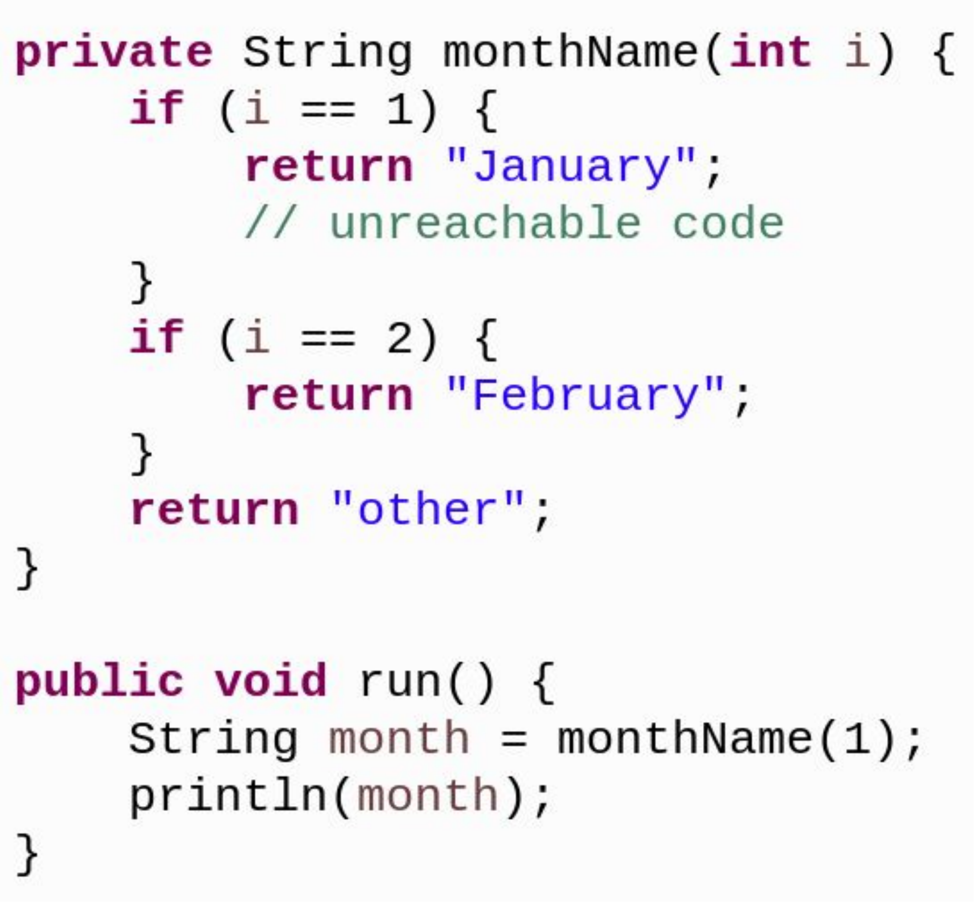
```
private void drawSquare(double width, double height, boolean centered) {
    GRect square = new GRect(width, height);
    if (centered) {
        add(square, (getWidth() - width) / 2, (getHeight() - height) / 2);
    } else {
        add(square);
    }
}

public void run() {
    drawSquare(35.7, 14.8, true);
}
```

Example 5 - Multiple Returns



```
private String monthName(int i) {
    if (i == 1) {
        return "January";
        //unreachable code
    } else if (i == 2) {
        return "February";
    } else {
        return "other";
    }
}

public void run() {
    String month = monthName(1);
    println(month);
}
```

Example 5 - Multiple Returns



```
private String monthName(int i) {
    if (i == 1) {
        return "January";
        // unreachable code
    }
    if (i == 2) {
        return "February";
    }
    return "other";
}

public void run() {
    String month = monthName(1);
    println(month);
}
```

Defining a Method

```
visibility type nameOfMethod (parameters) {
        statements
}
```

- *visibility*: usually **`private`** or **`public`**
- *type*: type returned by method (e.g., **`int`**, **`double`**, *etc.*)
  - Can be **`void`** to indicate that nothing is returned
- *parameters*: information passed into method

Powerful,
you now are.

Yes, hmmm.

```
private void run() {
   double r = readPositive(“Enter radius: “);
   double area = getArea(r);
   println(area);
}

private double readPositive(String prompt) {
   double value = readDouble(prompt);
   while(value < 0) {
      println(“Invalid”);
      value = readDouble(prompt);
   }
   return value;
}


private double getArea(double radius) {
   return PI * radius * radius;
}
```

```
private void run() {
    double r = readPositive("Enter radius: ");
    double area = getArea(r);
    println(area);
}
```

```
private void run() {
    double r = readPositive(“Enter radius: “);
    double area = getArea(r);
    println(area);
}
```

```
private void run() {
   double r = readPositive(“Enter radius: “);
   double area = getArea(r);
   println(area);
}
```

```
private void run() {
```

```
private double readPositive(String prompt) {
   double value = readDouble(prompt);
   while(value < 0) {
      println("Invalid");
      value = readDouble(prompt);
   }
   return value;
}
```

```
private void run() {
```

```
private double readPositive(String prompt) {
    double value = readDouble(prompt);
    while(value < 0) {
        println("Invalid");
        value = readDouble(prompt);
    }
    return value;
}
```

prompt "Enter radius: "

```
private void run() {
```

```
private double readPositive(String prompt) {
    double value = readDouble(prompt);
    while(value < 0) {
        println("Invalid");
        value = readDouble(prompt);
    }
    return value;
}
```

prompt "Enter radius: "

value -3

```
private void run() {
```

```
private double readPositive(String prompt) {
    double value = readDouble(prompt);
    while(value < 0) {
        println("Invalid");
        value = readDouble(prompt);
    }
    return value;
}
```

prompt "Enter radius: "

value -3

```
private void run() {
```

```
private double readPositive(String prompt) {
    double value = readDouble(prompt);
    while(value < 0) {
        println("Invalid");
        value = readDouble(prompt);
    }
    return value;
}
```

prompt "Enter radius: "

value -3

```
private void run() {
```

```
private double readPositive(String prompt) {
    double value = readDouble(prompt);
    while(value < 0) {
        println("Invalid");
        value = readDouble(prompt);
    }
    return value;
}
```

prompt "Enter radius: "

value -3

```
private void run() {
```

```
private double readPositive(String prompt) {
   double value = readDouble(prompt);
   while(value < 0) {
      println("Invalid");
      value = readDouble(prompt);
   }
   return value;
}
```

prompt "Enter radius: "

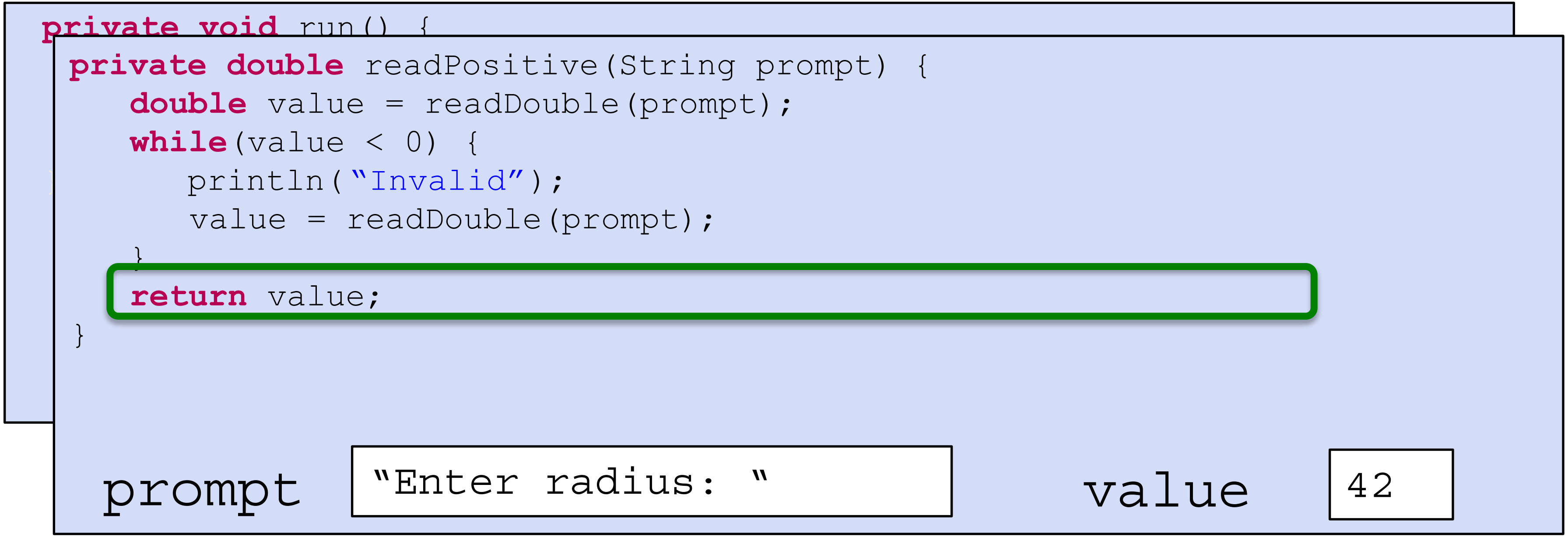
value 42

```
private void run() {
```

```
private double readPositive(String prompt) {
    double value = readDouble(prompt);
    while(value < 0) {
        println(“Invalid”);
        value = readDouble(prompt);
    }
    return value;
}
```

prompt “Enter radius: “

value 42



```
private void run() {
```

```
private double readPositive(String prompt) {
   double value = readDouble(prompt);
   while(value < 0) {
      println(“Invalid”);
      value = readDouble(prompt);
   }
   return value;
}
```

prompt “Enter radius: “

value 42

```
private void run() {
    double r = readPositive(“Enter radius: “);
    double area = getArea(r);
    println(area);
}
```

42

```
private void run() {
   double r = readPositive(“Enter radius: “);
   double area = getArea(r);
   println(area);
}
```

42

r 42

```
private void run() {
    double r = readPositive("Enter radius: ");
    double area = getArea(r);
    println(area);
}
```

r 42

```
private void run() {
    double r = readPositive("Enter radius: ");
    double area = getArea(r);
    println(area);
}
```

r 42

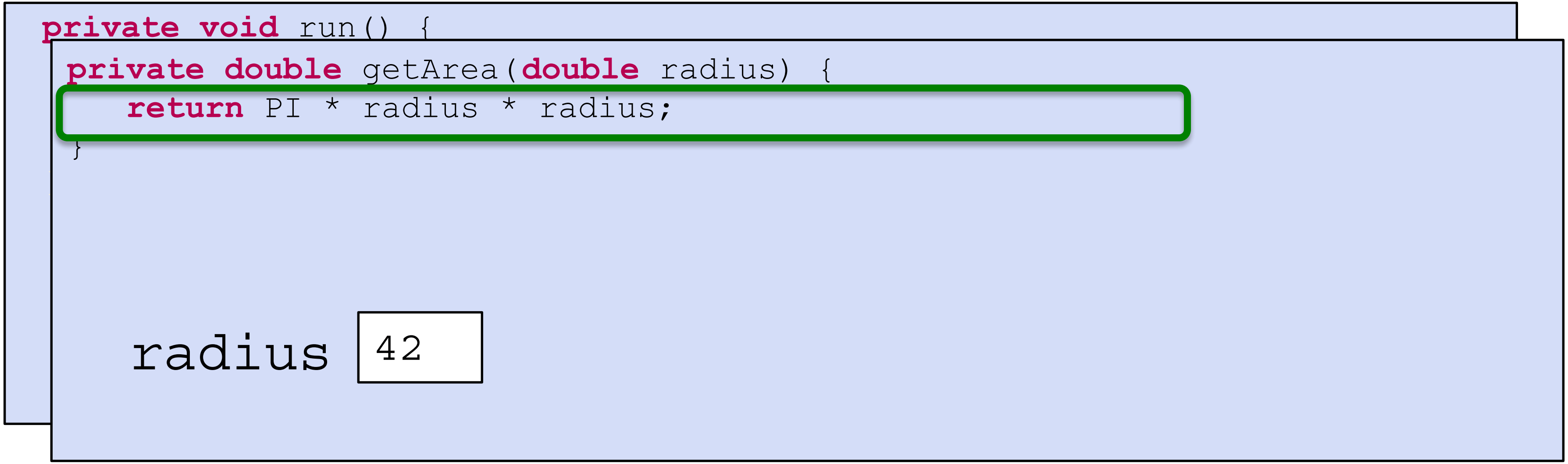
```
private void run() {
```

```
private double getArea(double radius) {
   return PI * radius * radius;
}
```

```
private void run() {
```

```
private double getArea(double radius) {
    return PI * radius * radius;
}
```

radius 42



```
private void run() {
```

```
private double getArea(double radius) {
    return PI * radius * radius;
}
```

radius 42

```
private void run() {
```

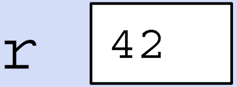
```
private double getArea(double radius) {
    return PI * radius * radius;
}
```

5538.96

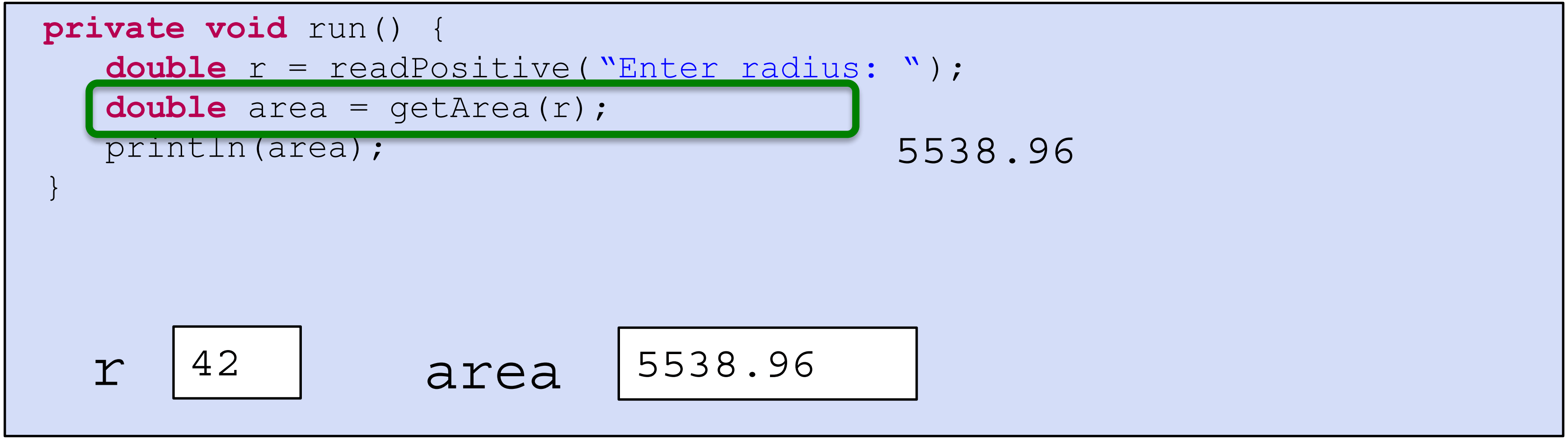
radius 42

```
private void run() {
   double r = readPositive("Enter radius: ");
   double area = getArea(r);
   println(area);
}
```

5538.96



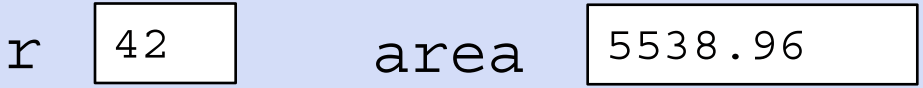
r 42



```
private void run() {
    double r = readPositive("Enter radius: ");
    double area = getArea(r);
    println(area);
}
```

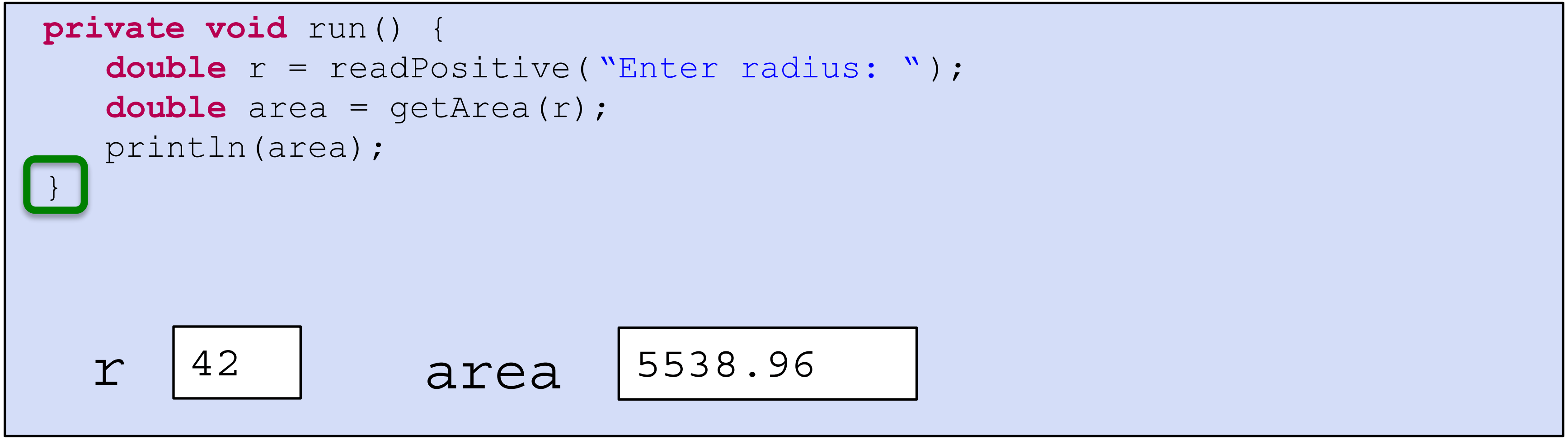
```
5538.96
```

```
private void run() {
   double r = readPositive("Enter radius: ");
   double area = getArea(r);
   println(area);
}
```



r 42

area 5538.96

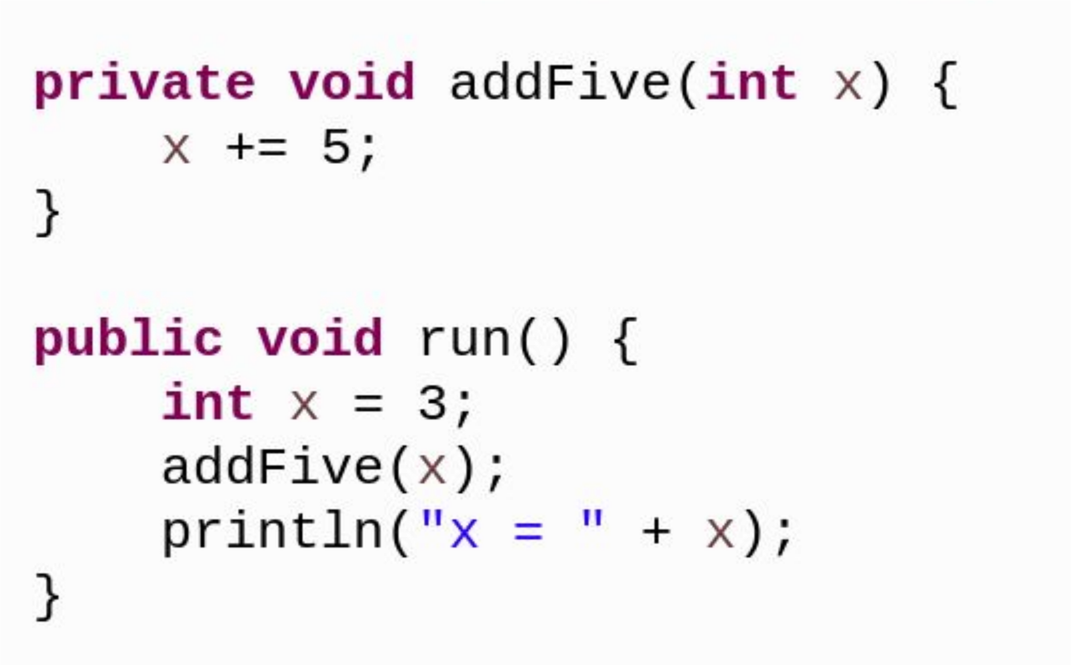


```
private void run() {
   double r = readPositive("Enter radius: ");
   double area = getArea(r);
   println(area);
}
```

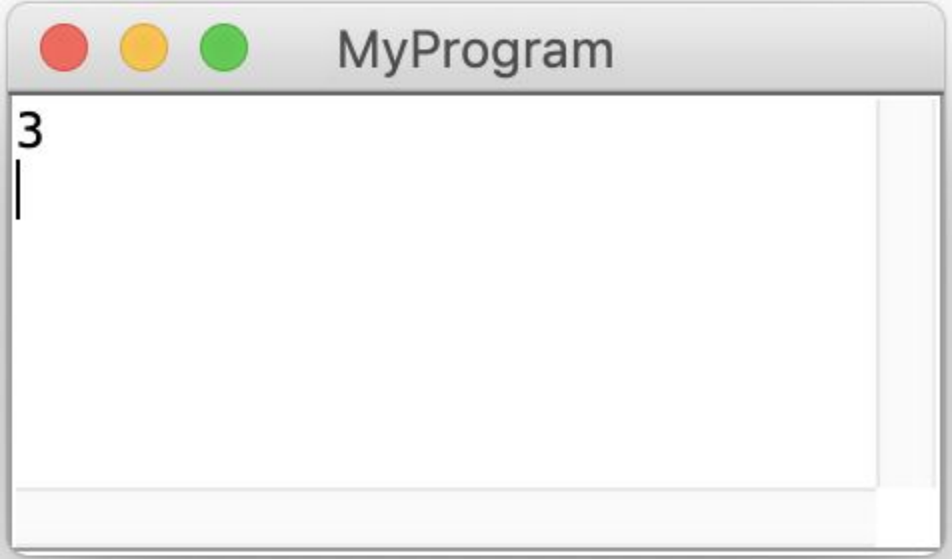
r 42

area 5538.96

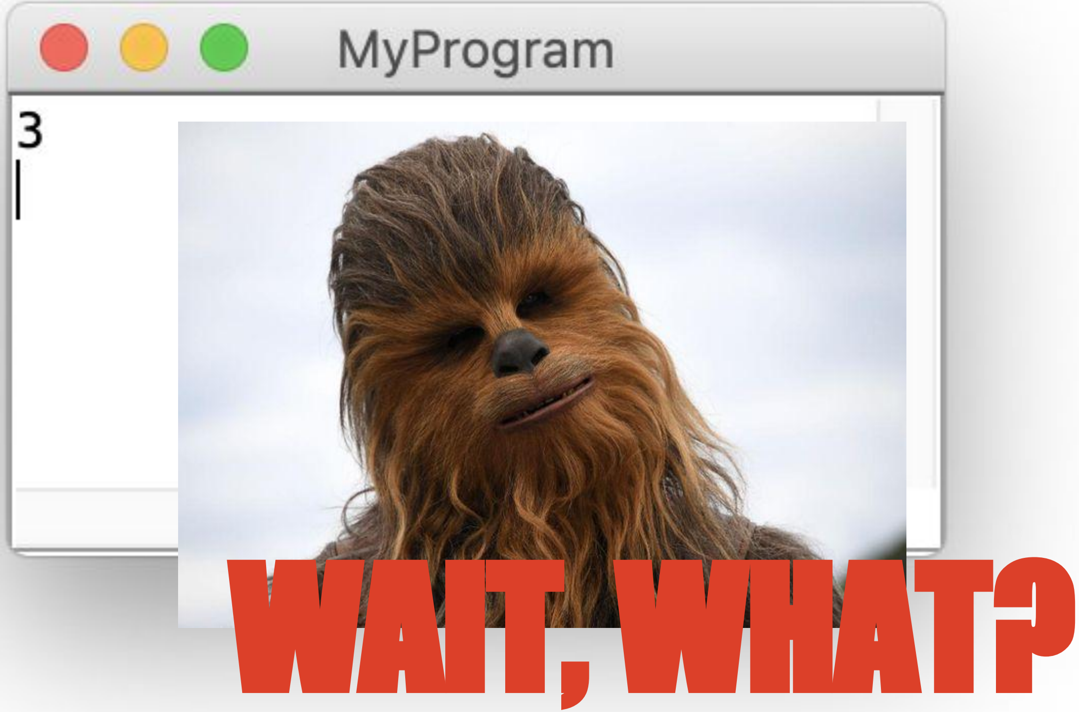
More practise...



```
private void addFive(int x) {
    x += 5;
}

public void run() {
    int x = 3;
    addFive(x);
    println("x = " + x);
}
```



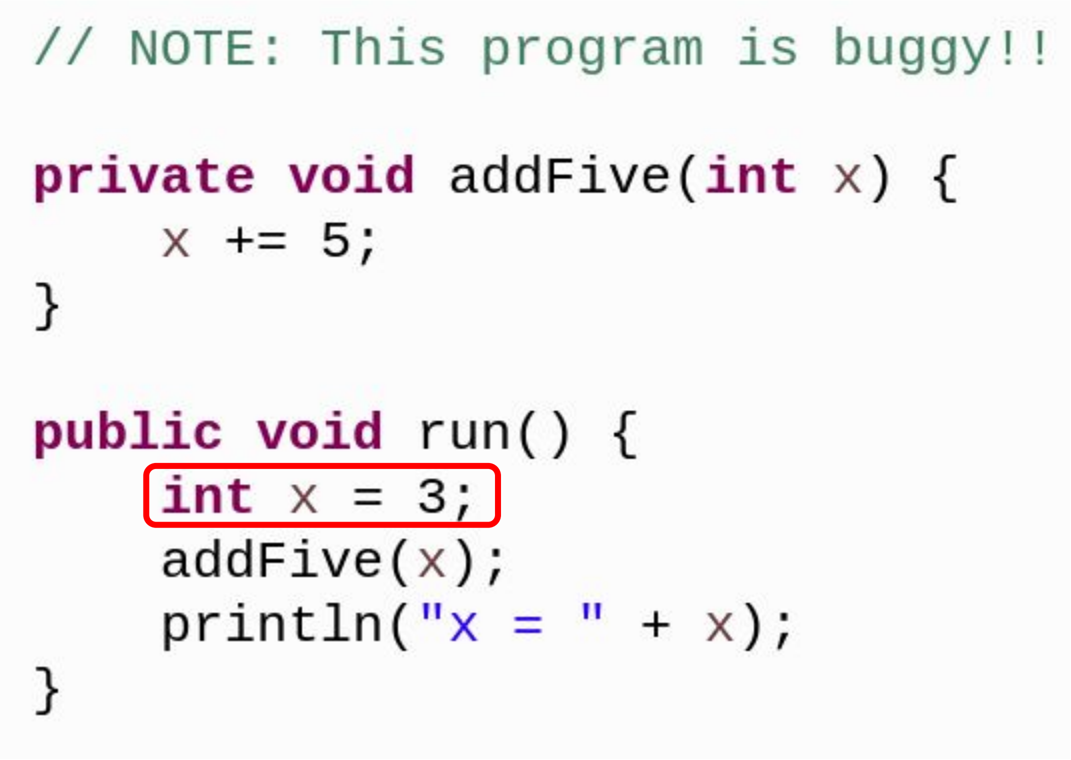
```
3
```



```
3
```

WAIT, WHAT?

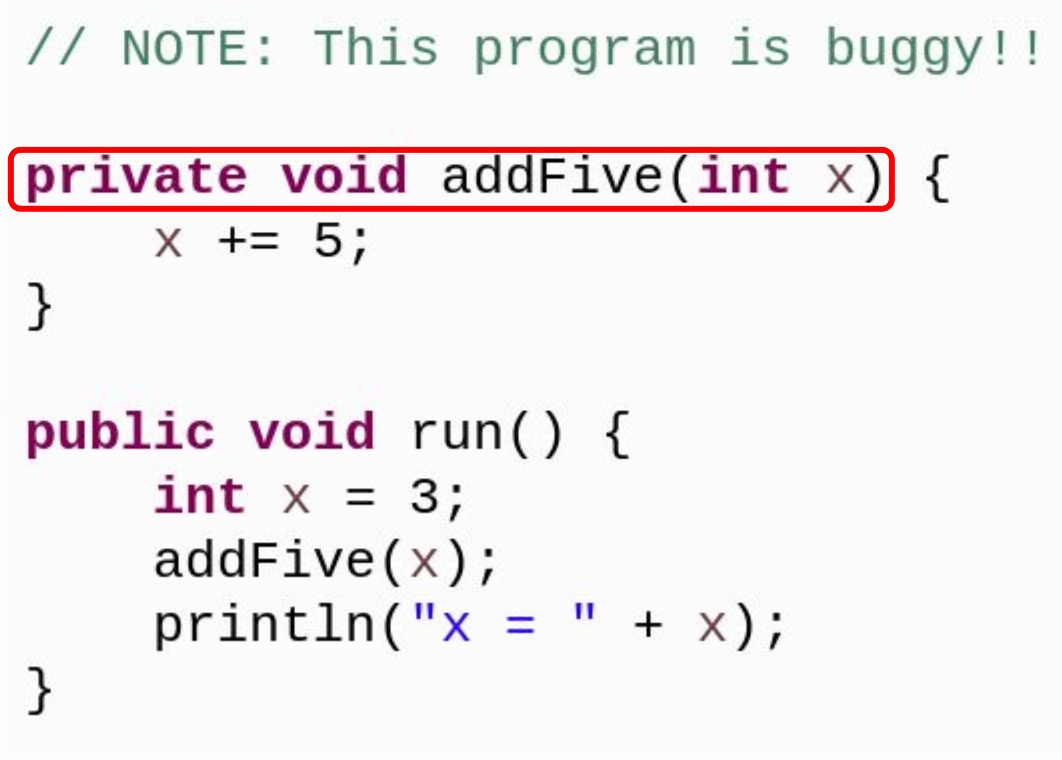
There is a Bug!



```
// NOTE: This program is buggy!!

private void addFive(int x) {
    x += 5;
}

public void run() {
    int x = 3;
    addFive(x);
    println("x = " + x);
}
```

x: 3

There is a Bug!

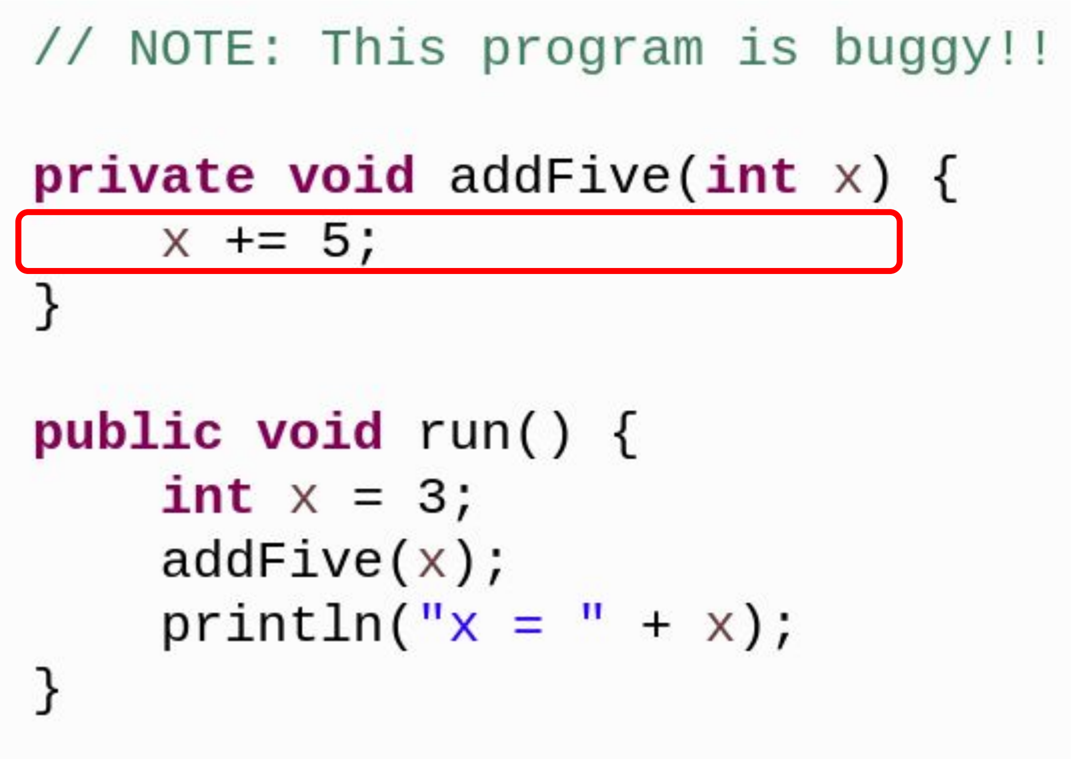


```
// NOTE: This program is buggy!!

private void addFive(int x) {
    x += 5;
}

public void run() {
    int x = 3;
    addFive(x);
    println("x = " + x);
}
```

int x = 3

x: 3

There is a Bug!

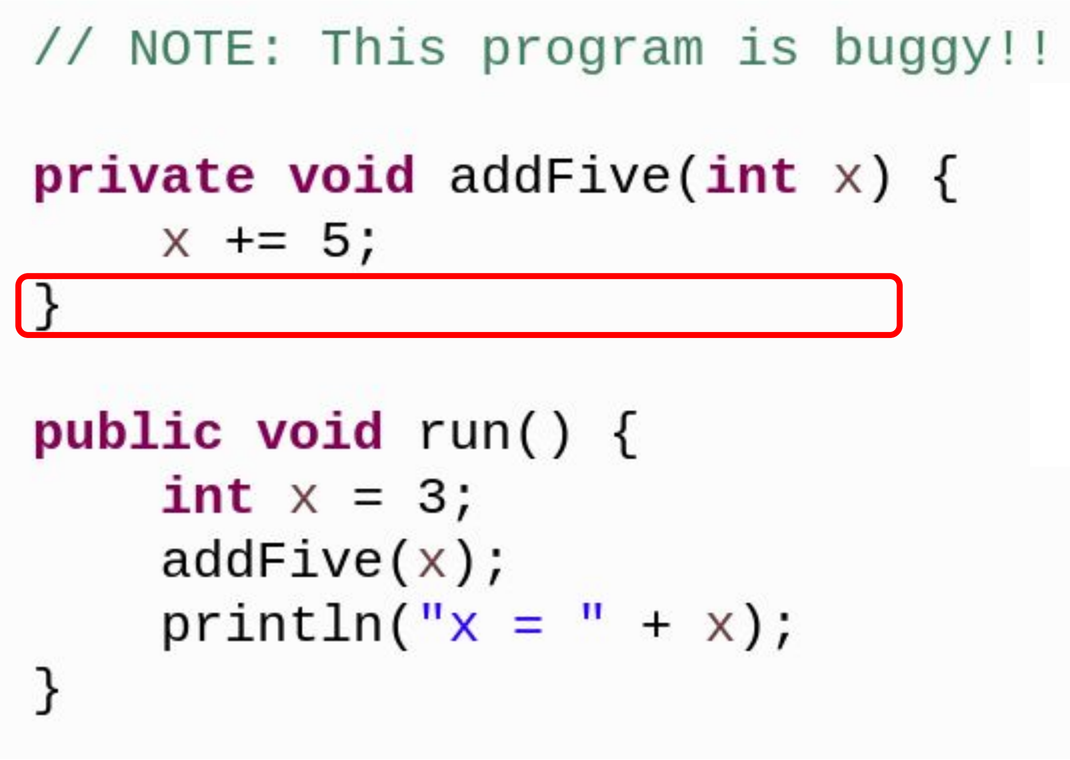


```
// NOTE: This program is buggy!!

private void addFive(int x) {
    x += 5;
}

public void run() {
    int x = 3;
    addFive(x);
    println("x = " + x);
}
```

x: 8

x: 3

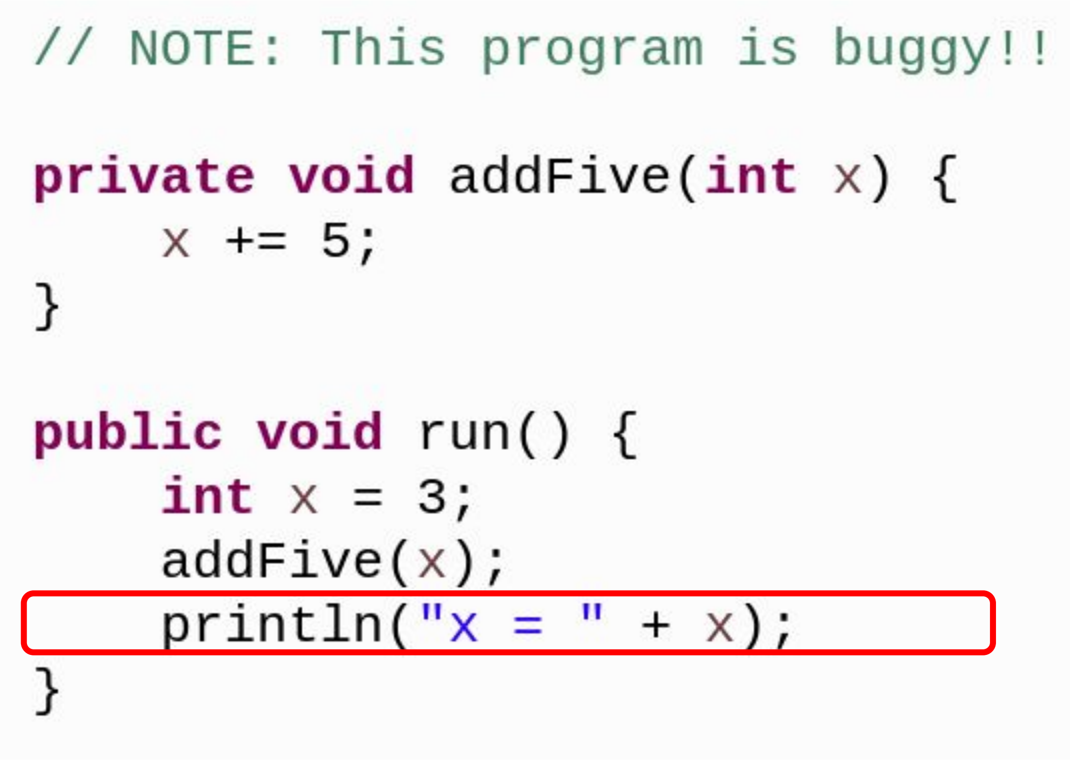
There is a Bug!



```
// NOTE: This program is buggy!!

private void addFive(int x) {
    x += 5;
}

public void run() {
    int x = 3;
    addFive(x);
    println("x = " + x);
}
```

x: 3

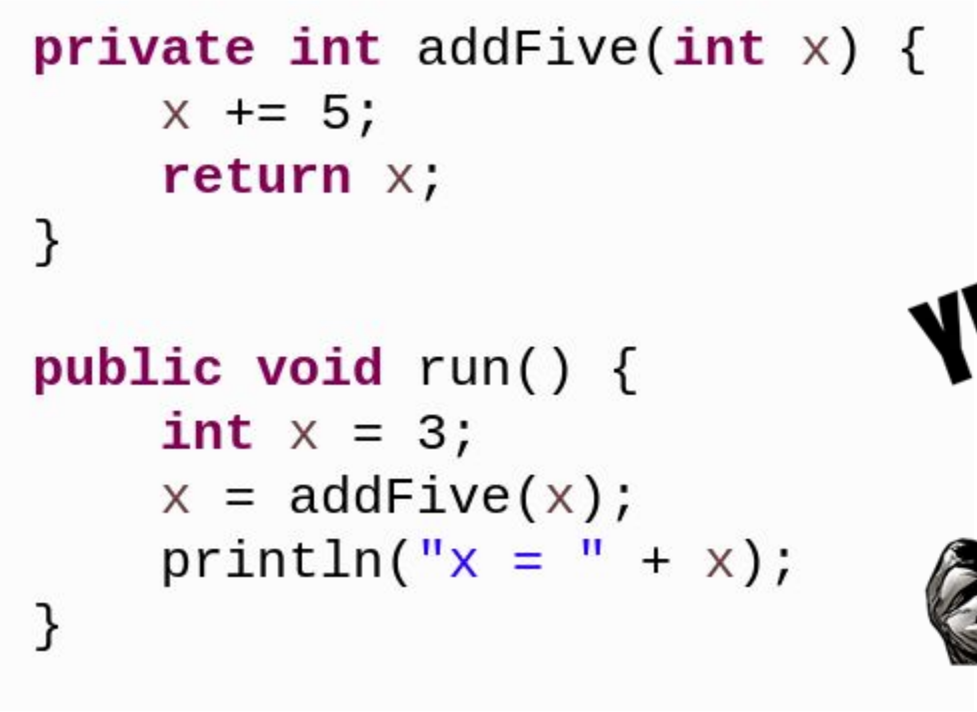
There is a Bug!



```
// NOTE: This program is buggy!!

private void addFive(int x) {
    x += 5;
}

public void run() {
    int x = 3;
    addFive(x);
    println("x = " + x);
}
```

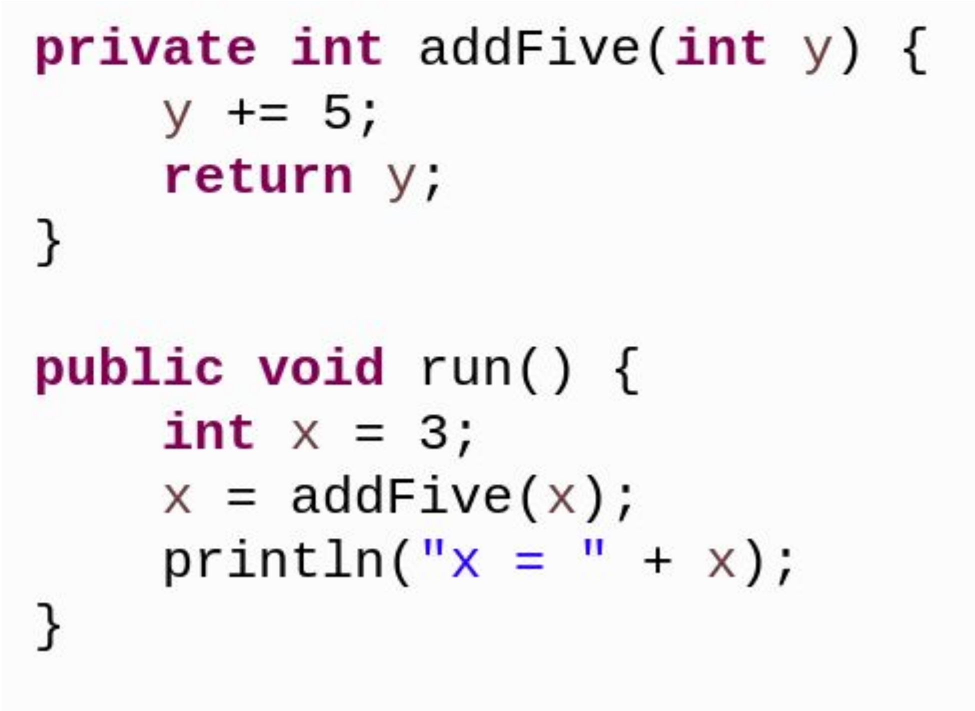
x: 3

Correct program



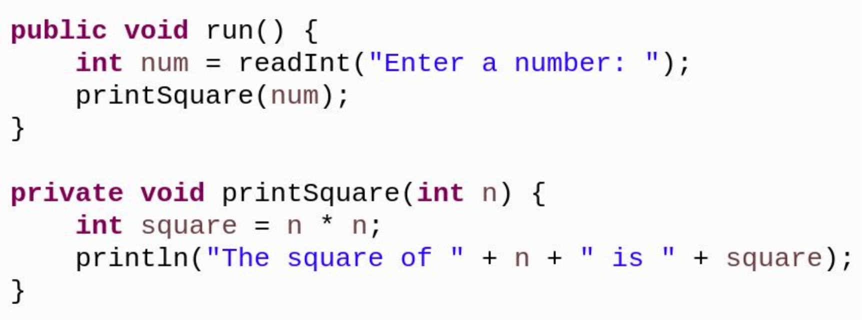
```
private int addFive(int x) {
    x += 5;
    return x;
}

public void run() {
    int x = 3;
    x = addFive(x);
    println("x = " + x);
}
```

works also with a different name

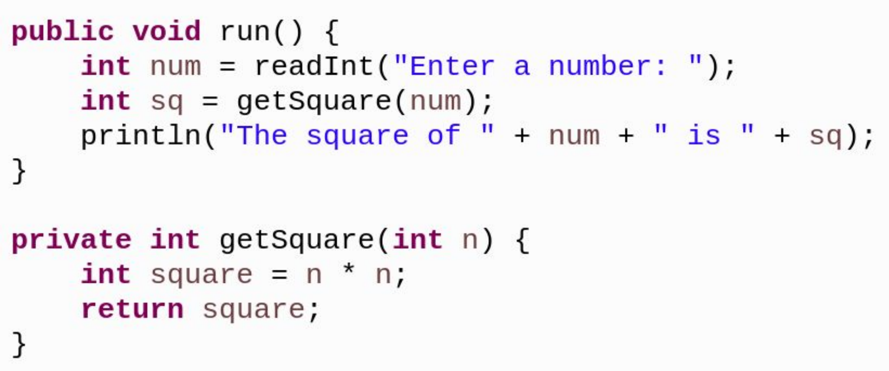


```
private int addFive(int y) {
    y += 5;
    return y;
}

public void run() {
    int x = 3;
    x = addFive(x);
    println("x = " + x);
}
```

Print vs. Return



```
public void run() {
    int num = readInt("Enter a number: ");
    printSquare(num);
}

private void printSquare(int n) {
    int square = n * n;
    println("The square of " + n + " is " + square);
}
```



```
public void run() {
    int num = readInt("Enter a number: ");
    int sq = getSquare(num);
    println("The square of " + num + " is " + sq);
}

private int getSquare(int n) {
    int square = n * n;
    return square;
}
```

Both programs do the same thing

THAT'S IT

MAY THE FORCE BE WITH YOU