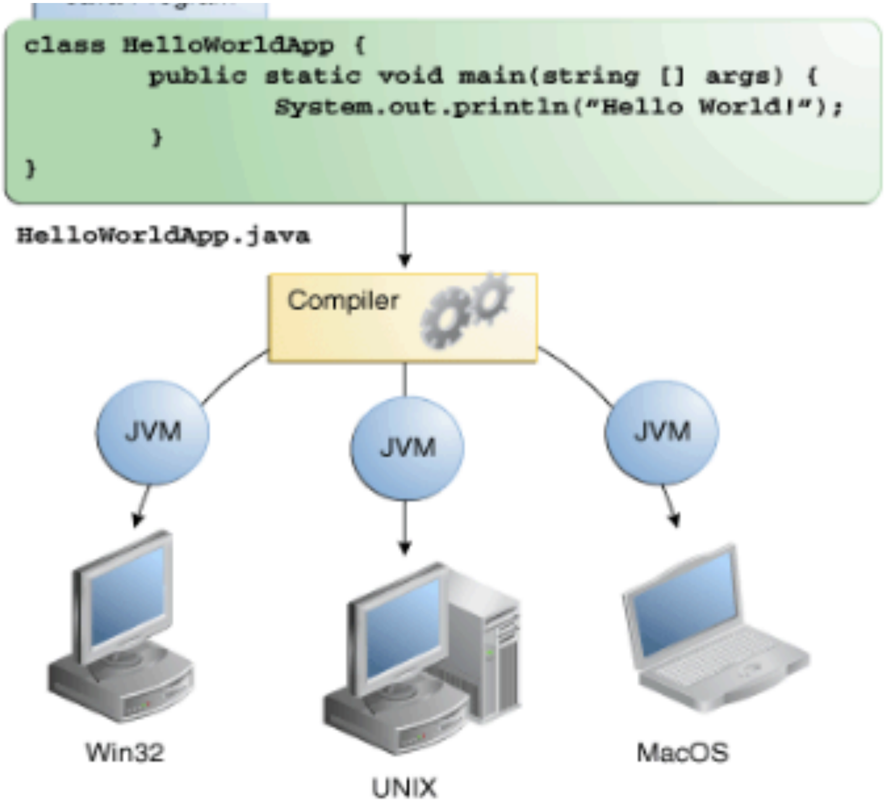
Java Confidence Course

Jan 2016

Just in case you get lost…

- Kickstart 2016 with Java
- Assumes a basic knowledge of programming with CS1010/ CS1010E
- Today’s Materials: https://github.com/WanderZ/JCC

Java



```
class HelloWorldApp {
        public static void main(string [] args) {
                System.out.println(“Hello World!”);
        }
}
```

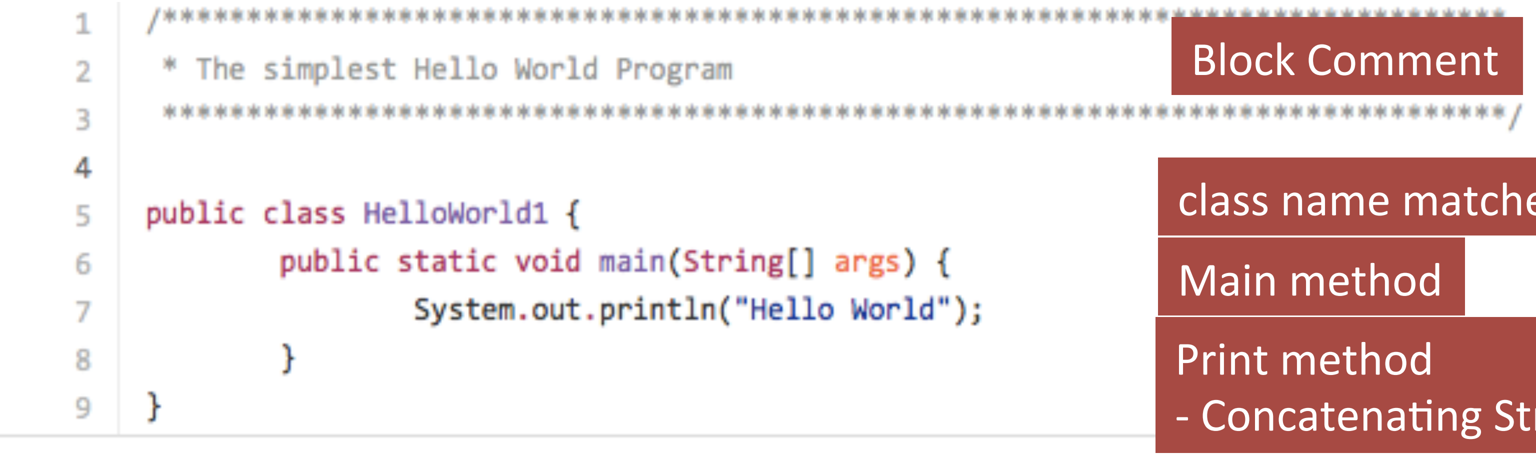
Through the Java VM, the same application is capable of running on multiple platforms.

Java Virtual Machine: Write once, run anywhere

00_HELLOWORLD

Hello World!

- HelloWorld1.java



```java
/********************************************************************************
 * The simplest Hello World Program
 ********************************************************************************/

public class HelloWorld1 {
        public static void main(String[] args) {
                System.out.println("Hello World");
        }
}
```

Block Comment

class name matches filename

Main method

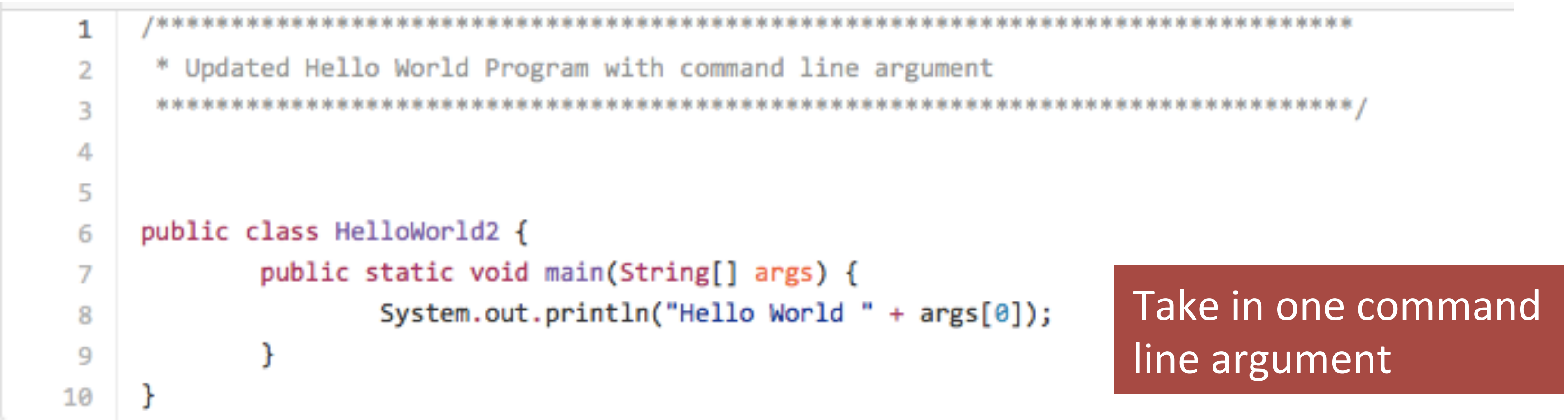
Print method
- Concatenating Strings, numbers…

javac HelloWorld1.java
java HelloWorld1

00_HelloWorld

Command Line Arguments

- HelloWorld2.java



```java
/*********************************************************************************
 * Updated Hello World Program with command line argument
 *********************************************************************************/


public class HelloWorld2 {
        public static void main(String[] args) {
                System.out.println("Hello World " + args[0]);
        }
}
```

Take in one command line argument

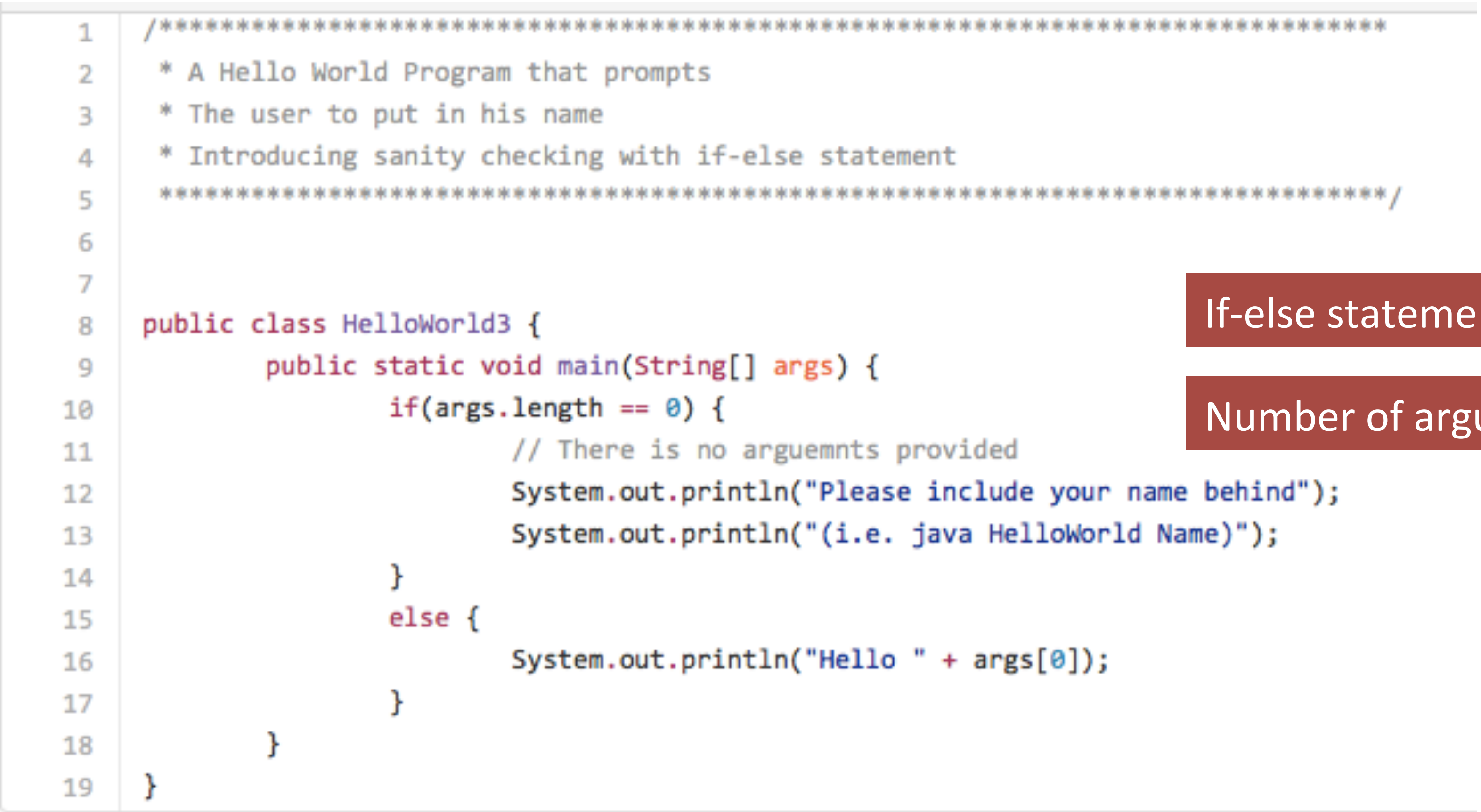
javac HelloWorld2.java
java HelloWorld2 BB8

00_HelloWorld

01_PROCEDURALPROGRAMMING

If-Else Statement

- HelloWorld3.java



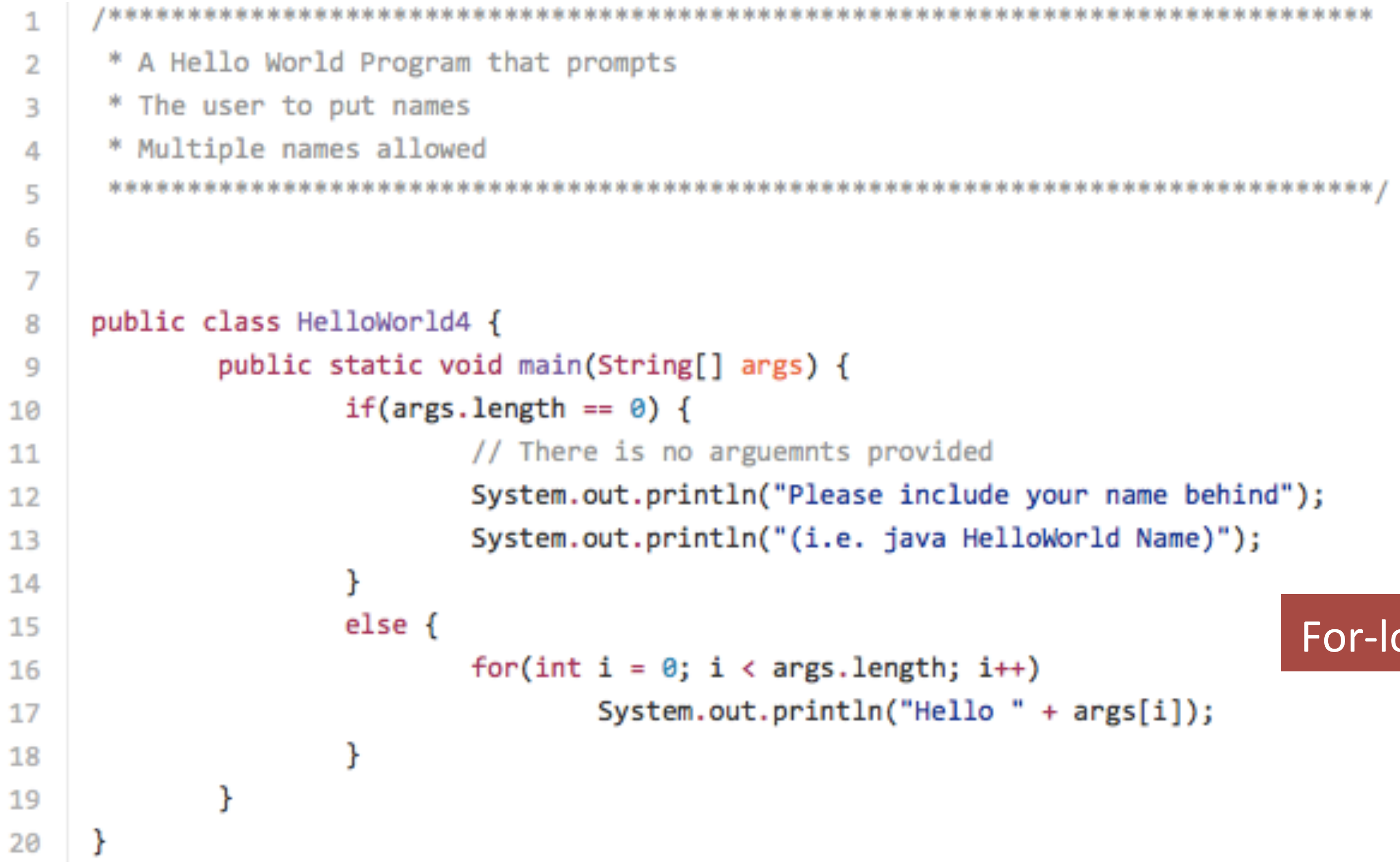
```
/********************************************************************************
 * A Hello World Program that prompts
 * The user to put in his name
 * Introducing sanity checking with if-else statement
 ********************************************************************************/


public class HelloWorld3 {
        public static void main(String[] args) {
                if(args.length == 0) {
                        // There is no arguemnts provided
                        System.out.println("Please include your name behind");
                        System.out.println("(i.e. java HelloWorld Name)");
                }
                else {
                        System.out.println("Hello " + args[0]);
                }
        }
}
```

If-else statement

Number of arguments

For-Loops

- HelloWorld4.java

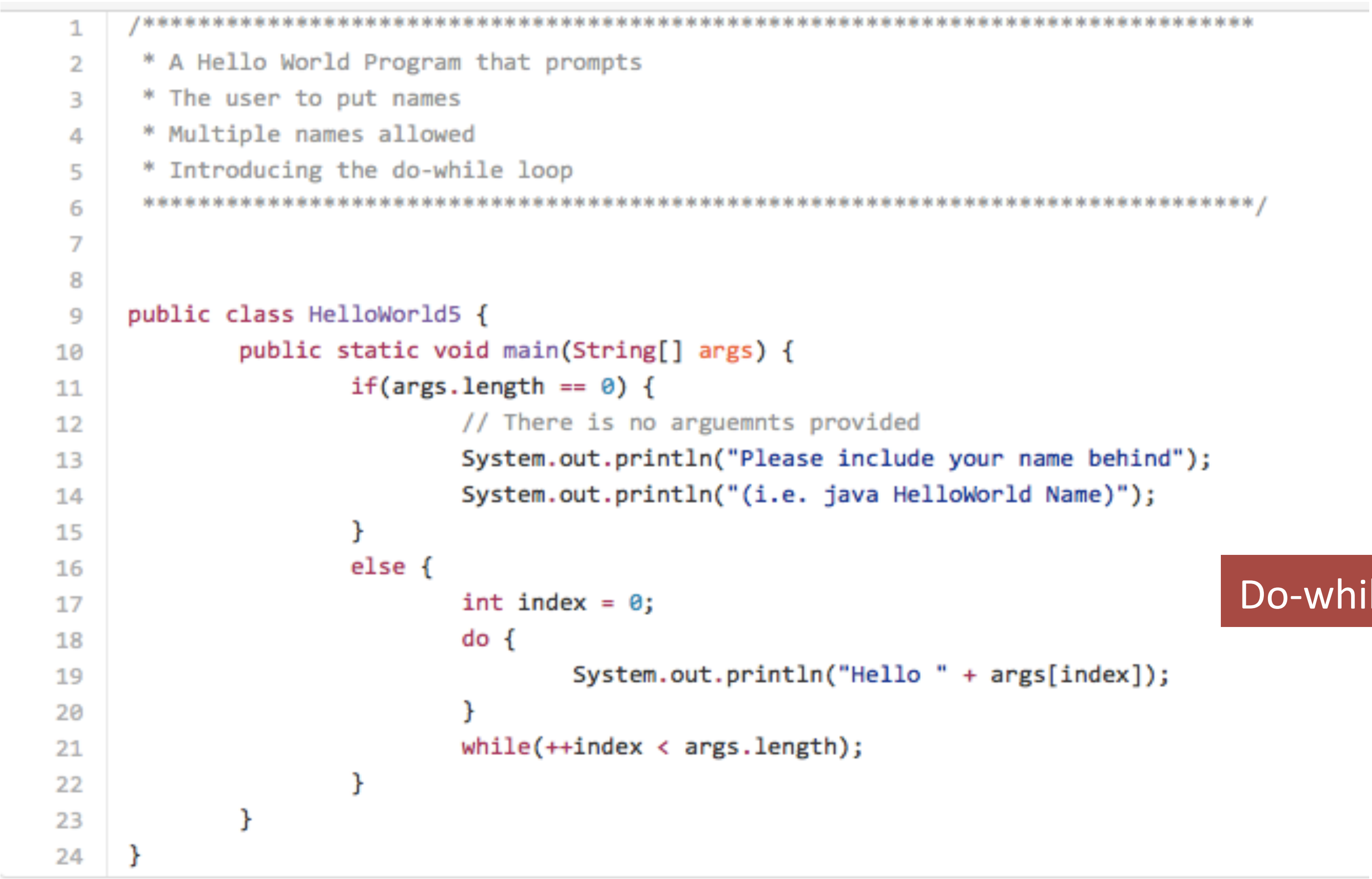


```java
/*****************************************************************************
 * A Hello World Program that prompts
 * The user to put names
 * Multiple names allowed
 *****************************************************************************/


public class HelloWorld4 {
        public static void main(String[] args) {
                if(args.length == 0) {
                        // There is no arguemnts provided
                        System.out.println("Please include your name behind");
                        System.out.println("(i.e. java HelloWorld Name)");
                }
                else {
                        for(int i = 0; i < args.length; i++)
                                System.out.println("Hello " + args[i]);
                }
        }
}
```

For-loops

Do-While loops

- HelloWorld5.java

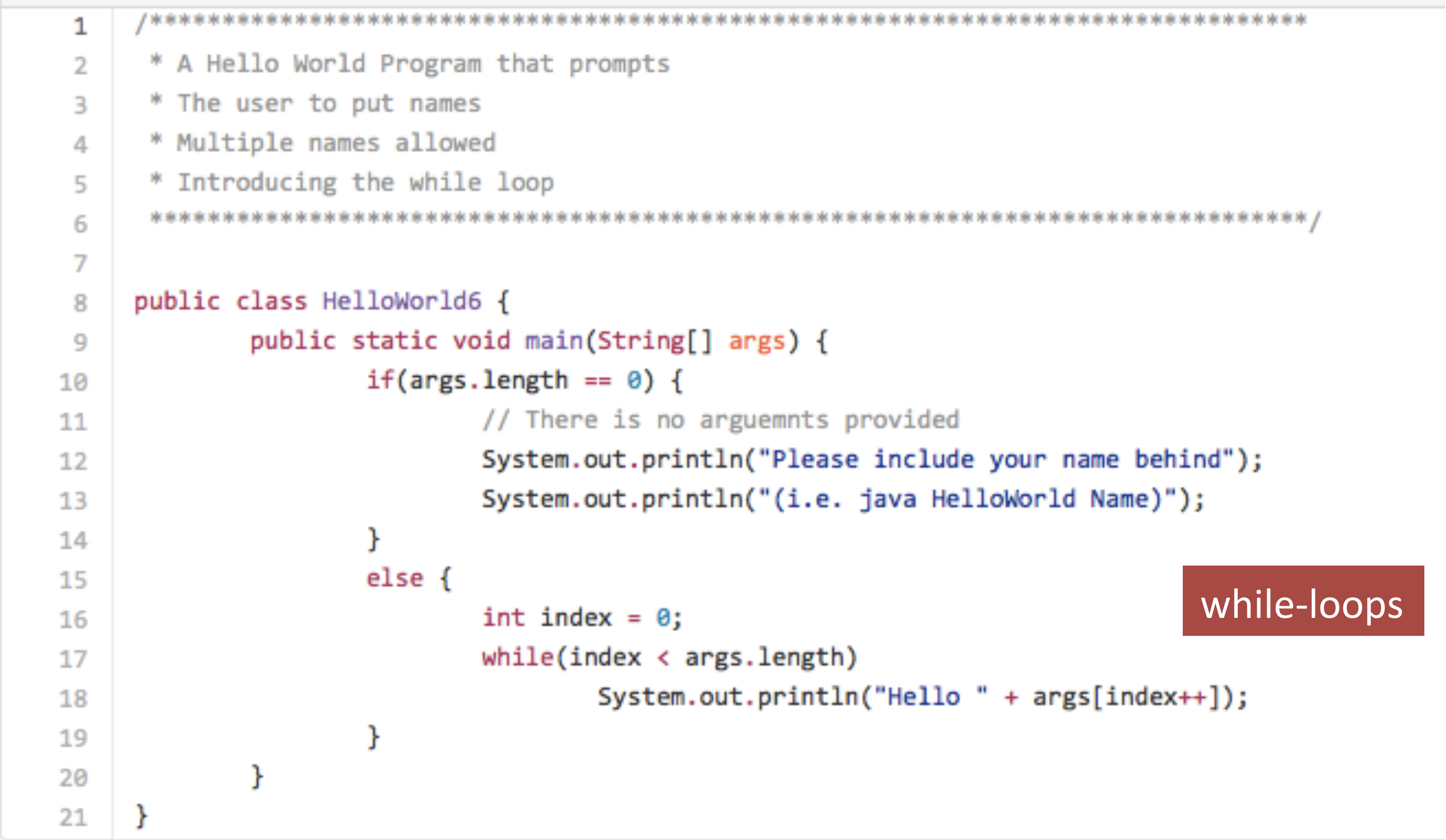


```
/********************************************************************************
 * A Hello World Program that prompts
 * The user to put names
 * Multiple names allowed
 * Introducing the do-while loop
 ********************************************************************************/


public class HelloWorld5 {
        public static void main(String[] args) {
                if(args.length == 0) {
                        // There is no arguemnts provided
                        System.out.println("Please include your name behind");
                        System.out.println("(i.e. java HelloWorld Name)");
                }
                else {
                        int index = 0;
                        do {
                                System.out.println("Hello " + args[index]);
                        }
                        while(++index < args.length);
                }
        }
}
```

Do-while-loops

While loops

- HelloWorld6.java

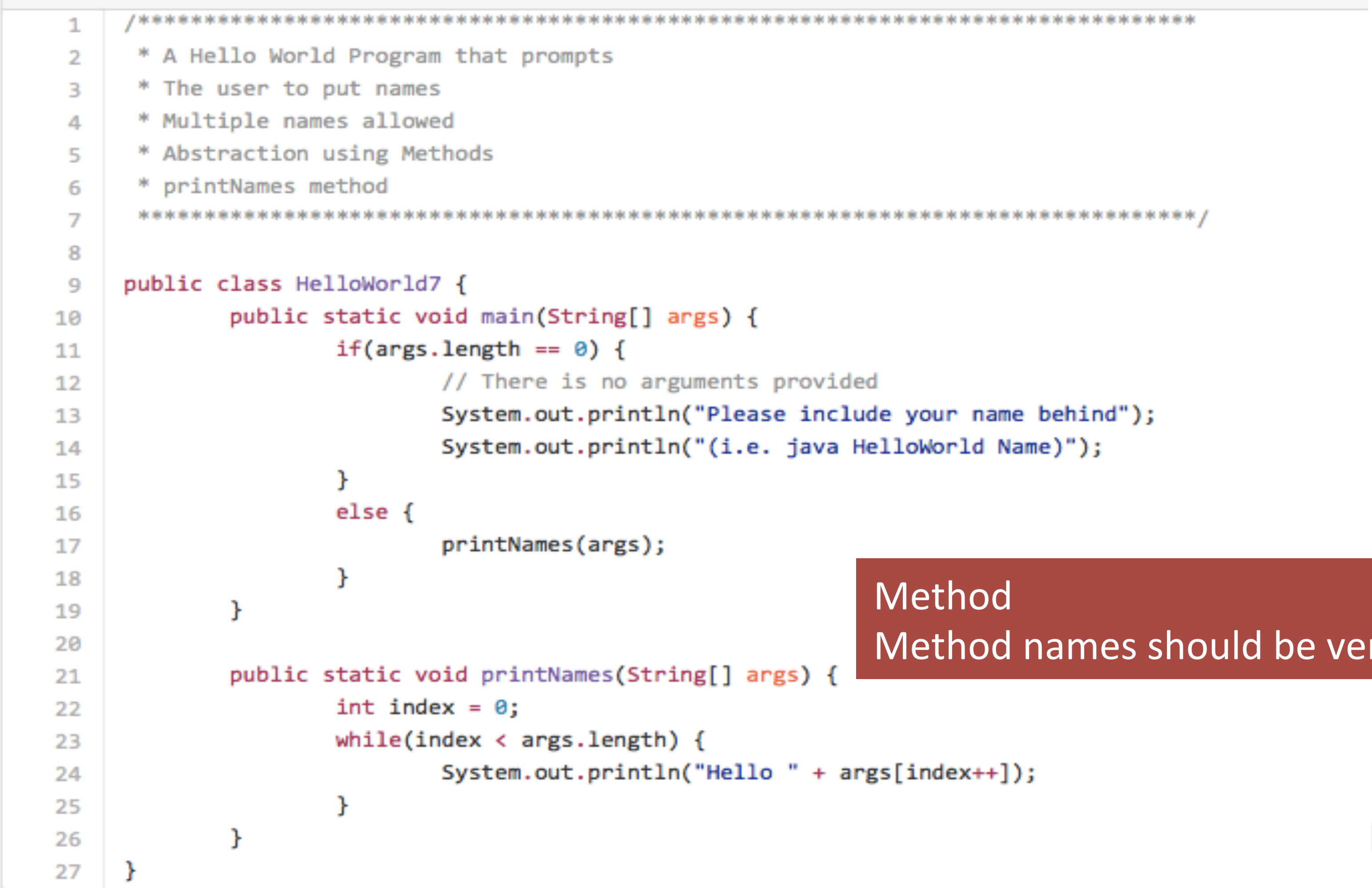


```java
/********************************************************************************
 * A Hello World Program that prompts
 * The user to put names
 * Multiple names allowed
 * Introducing the while loop
 ********************************************************************************/

public class HelloWorld6 {
        public static void main(String[] args) {
                if(args.length == 0) {
                        // There is no arguemnts provided
                        System.out.println("Please include your name behind");
                        System.out.println("(i.e. java HelloWorld Name)");
                }
                else {
                        int index = 0;
                        while(index < args.length)
                                System.out.println("Hello " + args[index++]);
                }
        }
}
```

while-loops

Abstraction using Methods

- HelloWorld7.java

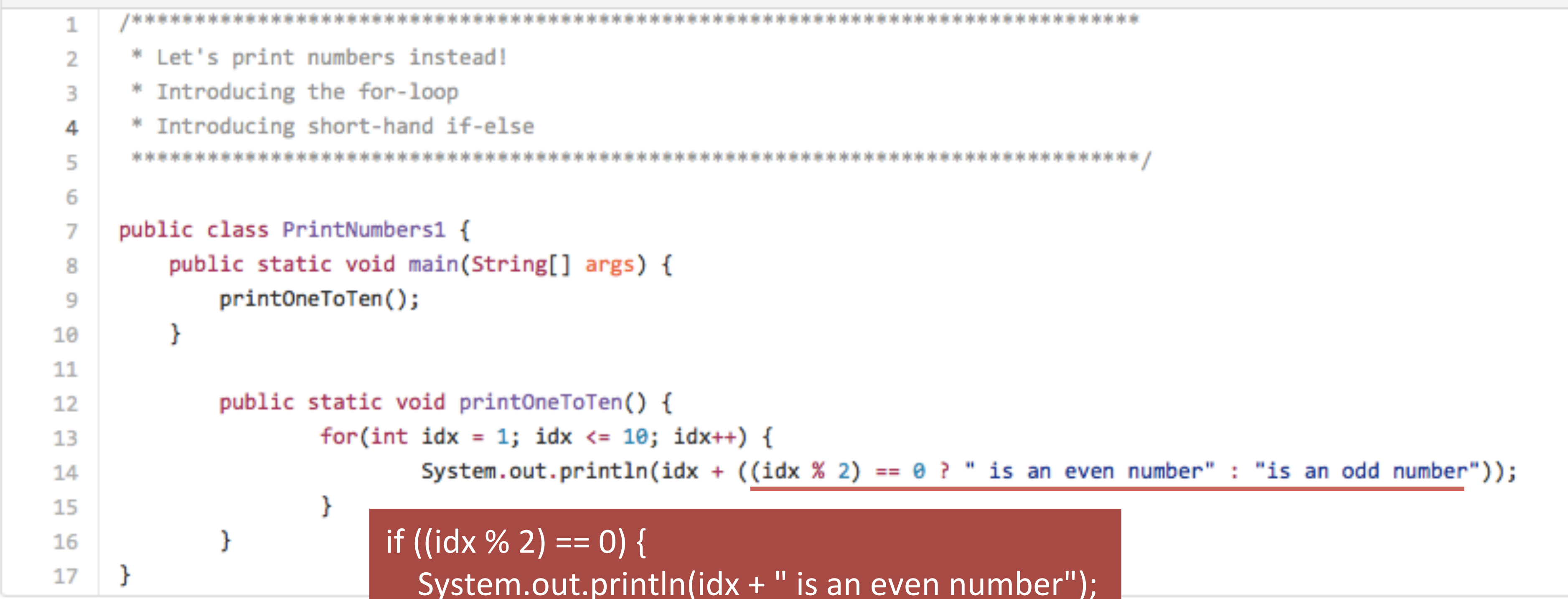


```java
/********************************************************************************
 * A Hello World Program that prompts
 * The user to put names
 * Multiple names allowed
 * Abstraction using Methods
 * printNames method
 ********************************************************************************/

public class HelloWorld7 {
        public static void main(String[] args) {
                if(args.length == 0) {
                        // There is no arguments provided
                        System.out.println("Please include your name behind");
                        System.out.println("(i.e. java HelloWorld Name)");
                }
                else {
                        printNames(args);
                }
        }

        public static void printNames(String[] args) {
                int index = 0;
                while(index < args.length) {
                        System.out.println("Hello " + args[index++]);
                }
        }
}
```

Method
Method names should be verbs, mixed-case

Ternary Operator

- PrintNumbers1.java

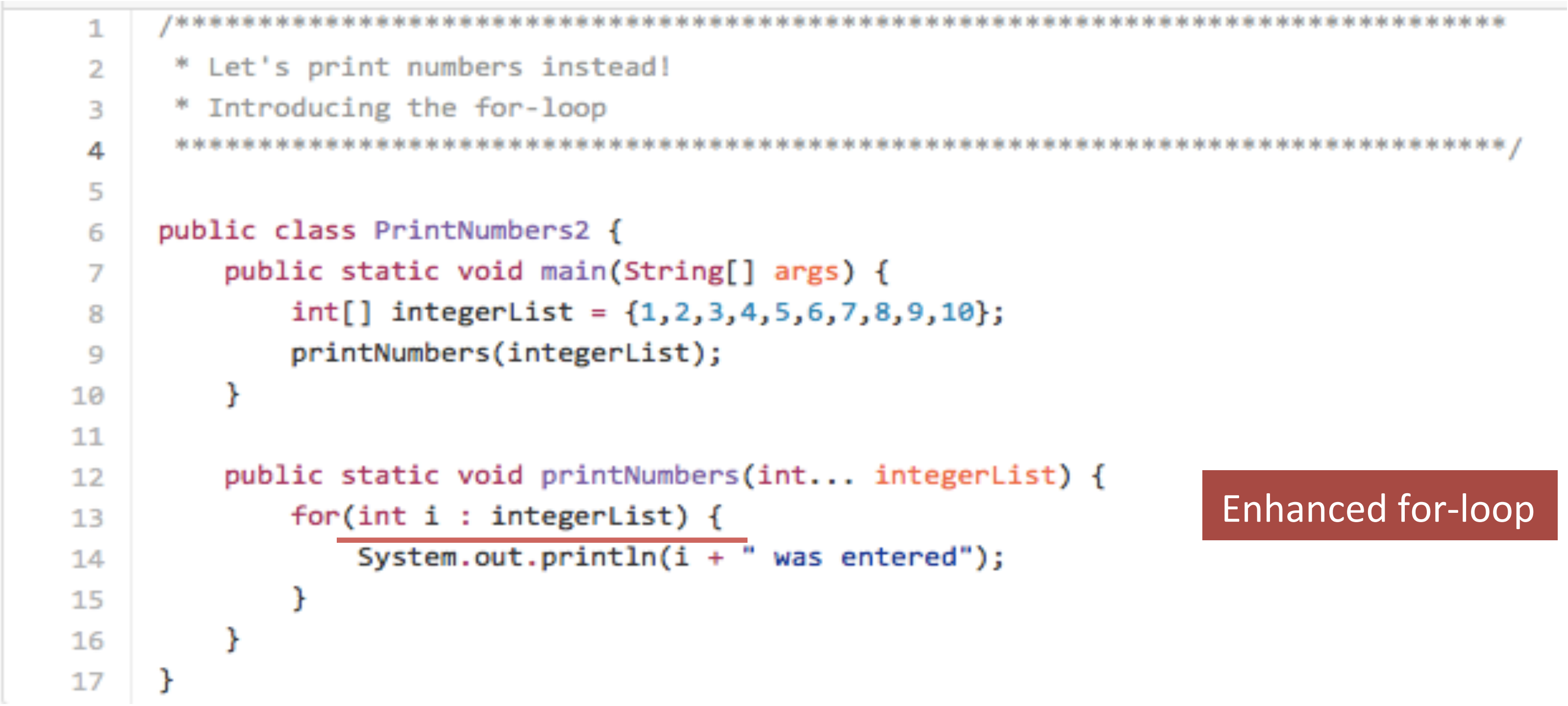


```
/*******************************************************************************
 * Let's print numbers instead!
 * Introducing the for-loop
 * Introducing short-hand if-else
 *******************************************************************************/

public class PrintNumbers1 {
    public static void main(String[] args) {
        printOneToTen();
    }

        public static void printOneToTen() {
                for(int idx = 1; idx <= 10; idx++) {
                        System.out.println(idx + ((idx % 2) == 0 ? " is an even number" : "is an odd number"));
                }
        }
}
```

```
if ((idx % 2) == 0) {
    System.out.println(idx + " is an even number");
}
else {
    System.out.println(idx + " is an odd number");
}
```

Enhanced For-Loop

- PrintNumbers2.java



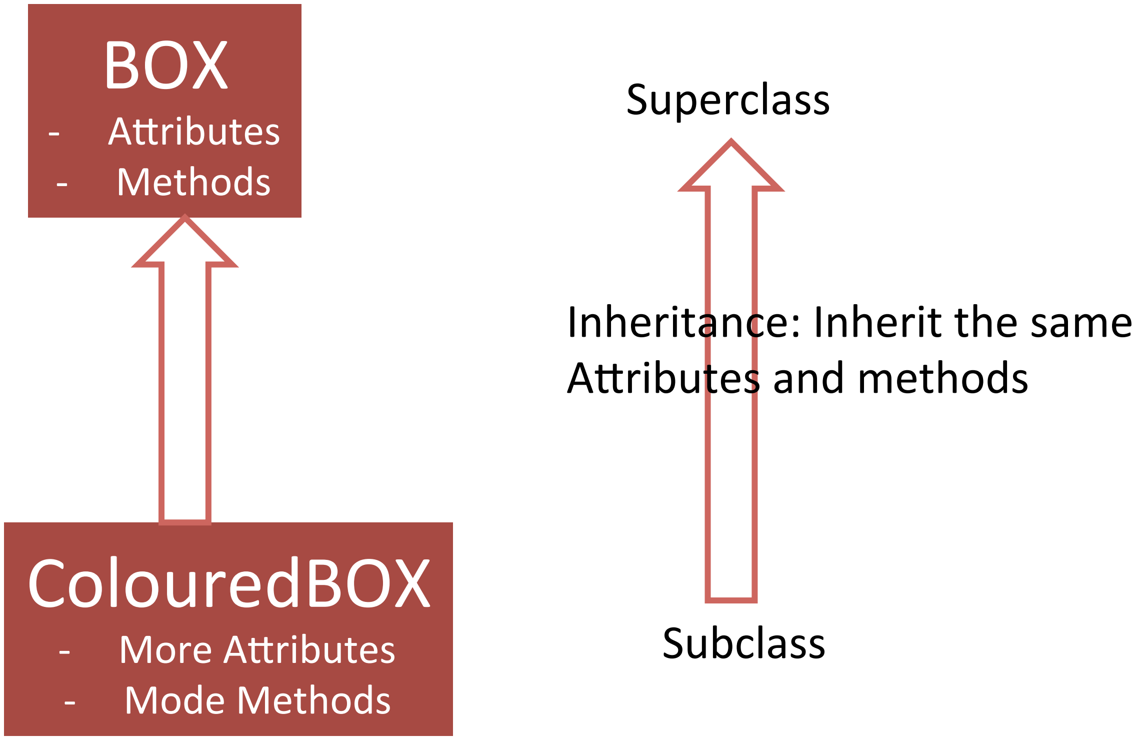
```java
/********************************************************************************
 * Let's print numbers instead!
 * Introducing the for-loop
 ********************************************************************************/

public class PrintNumbers2 {
    public static void main(String[] args) {
        int[] integerList = {1,2,3,4,5,6,7,8,9,10};
        printNumbers(integerList);
    }

    public static void printNumbers(int... integerList) {
        for(int i : integerList) {
            System.out.println(i + " was entered");
        }
    }
}
```

Enhanced for-loop

03_OBJECTORIENTEDPROGRAMMING

OOP

Objects like Box have Attributes
And Methods



BOX

- Attributes
- Methods

ColouredBOX

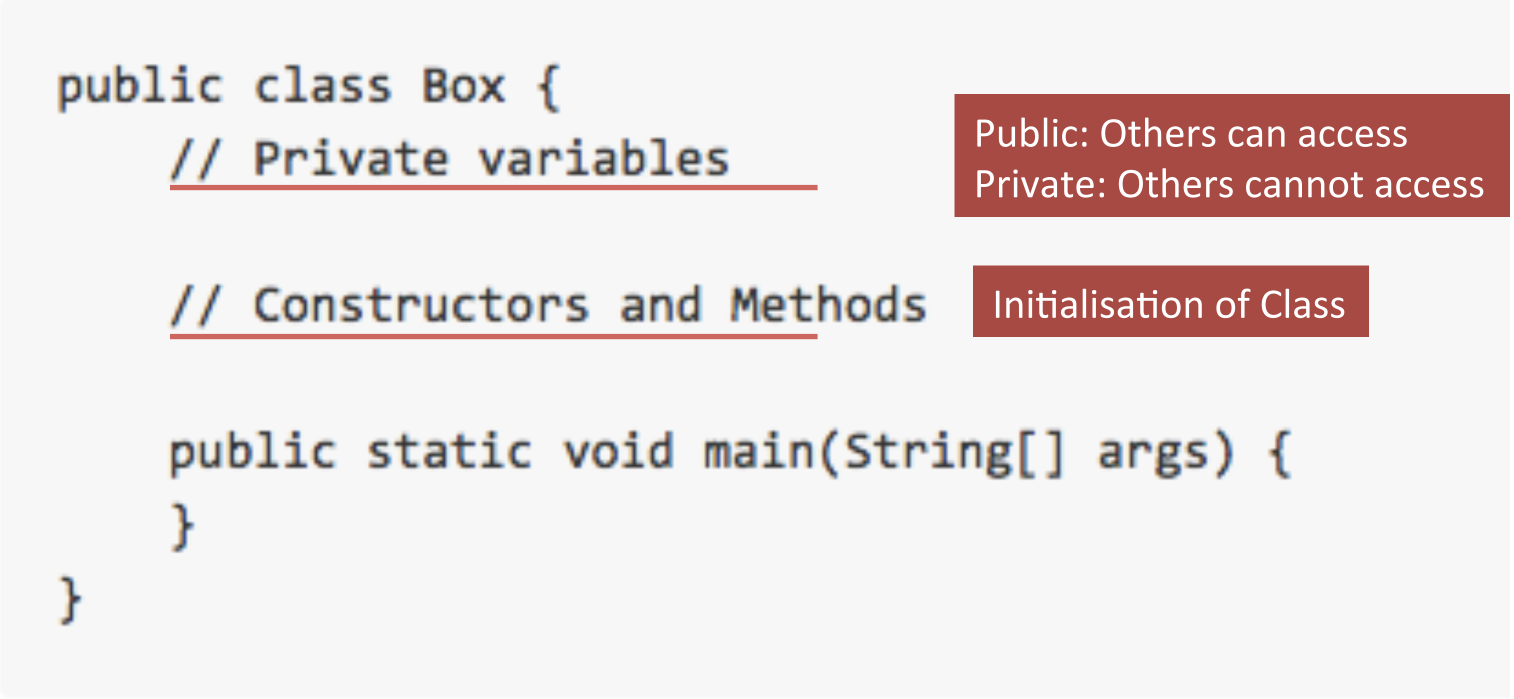
- More Attributes
- Mode Methods

Superclass

Inheritance: Inherit the same
Attributes and methods

Subclass

General Outline

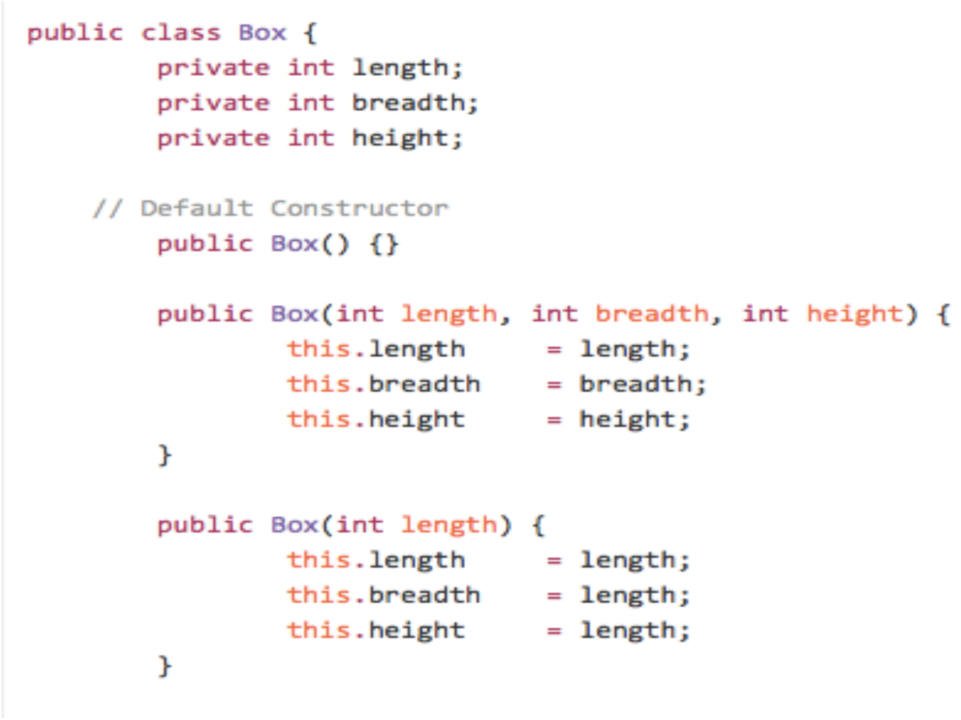


```
public class Box {
    // Private variables

    // Constructors and Methods

    public static void main(String[] args) {
    }
}
```

Public: Others can access
Private: Others cannot access

Initialisation of Class

Constructors



```
public class Box {
        private int length;
        private int breadth;
        private int height;

    // Default Constructor
        public Box() {}

        public Box(int length, int breadth, int height) {
                this.length     = length;
                this.breadth    = breadth;
                this.height     = height;
        }

        public Box(int length) {
                this.length     = length;
                this.breadth    = length;
                this.height     = length;
        }
```

Private Variables

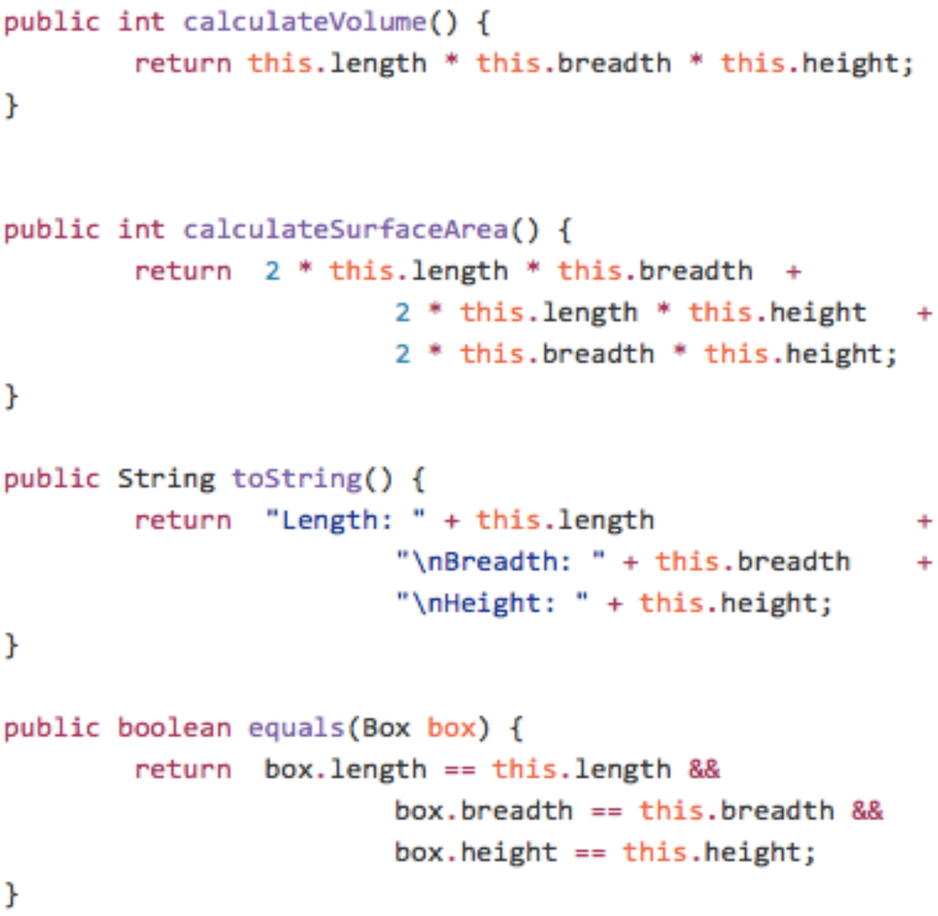
Default Constructor

Overloaded Constructor

This keyword to refer to attributes
In this particular object

02_OOP Box.java

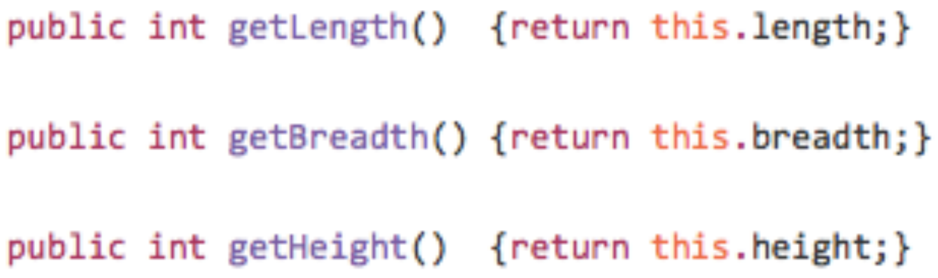
Some Public Methods



```
public int calculateVolume() {
        return this.length * this.breadth * this.height;
}


public int calculateSurfaceArea() {
        return  2 * this.length * this.breadth  +
                2 * this.length * this.height   +
                2 * this.breadth * this.height;
}

public String toString() {
        return  "Length: " + this.length                +
                "\nBreadth: " + this.breadth    +
                "\nHeight: " + this.height;
}

public boolean equals(Box box) {
        return  box.length == this.length &&
                box.breadth == this.breadth &&
                box.height == this.height;
}
```

02_OOP Box.java

Assessors/ get-set methods

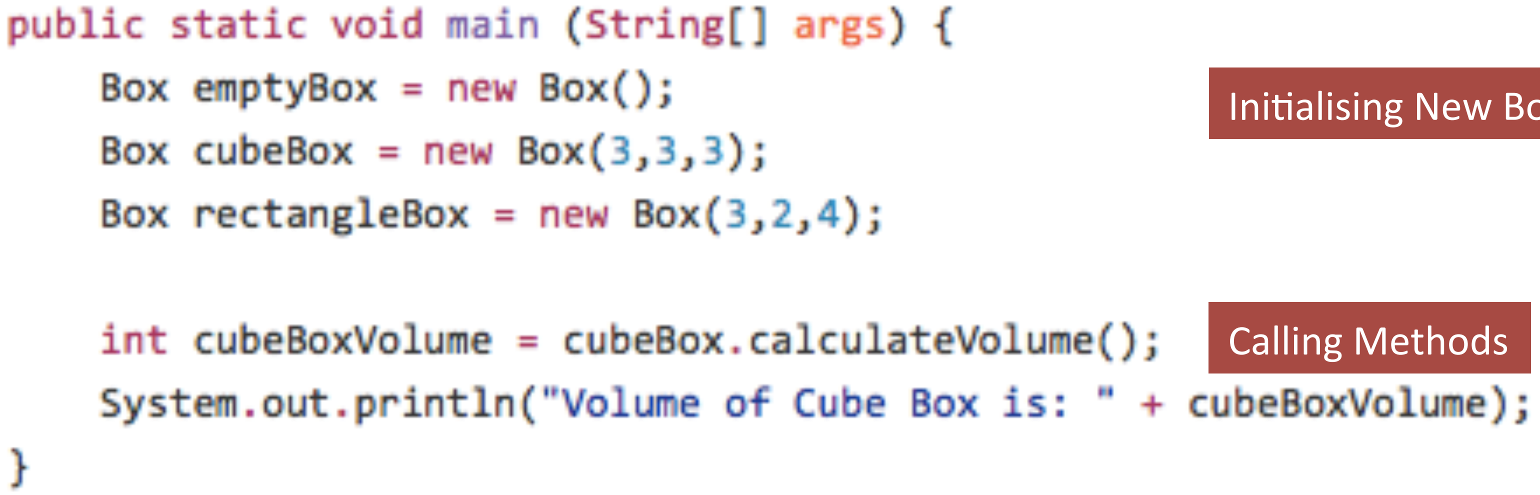


```
public int getLength()  {return this.length;}

public int getBreadth() {return this.breadth;}

public int getHeight()  {return this.height;}
```

Get/Set private variables

02_OOP Box.java

Main Method



```
public static void main (String[] args) {
    Box emptyBox = new Box();
    Box cubeBox = new Box(3,3,3);
    Box rectangleBox = new Box(3,2,4);

    int cubeBoxVolume = cubeBox.calculateVolume();
    System.out.println("Volume of Cube Box is: " + cubeBoxVolume);
}
```

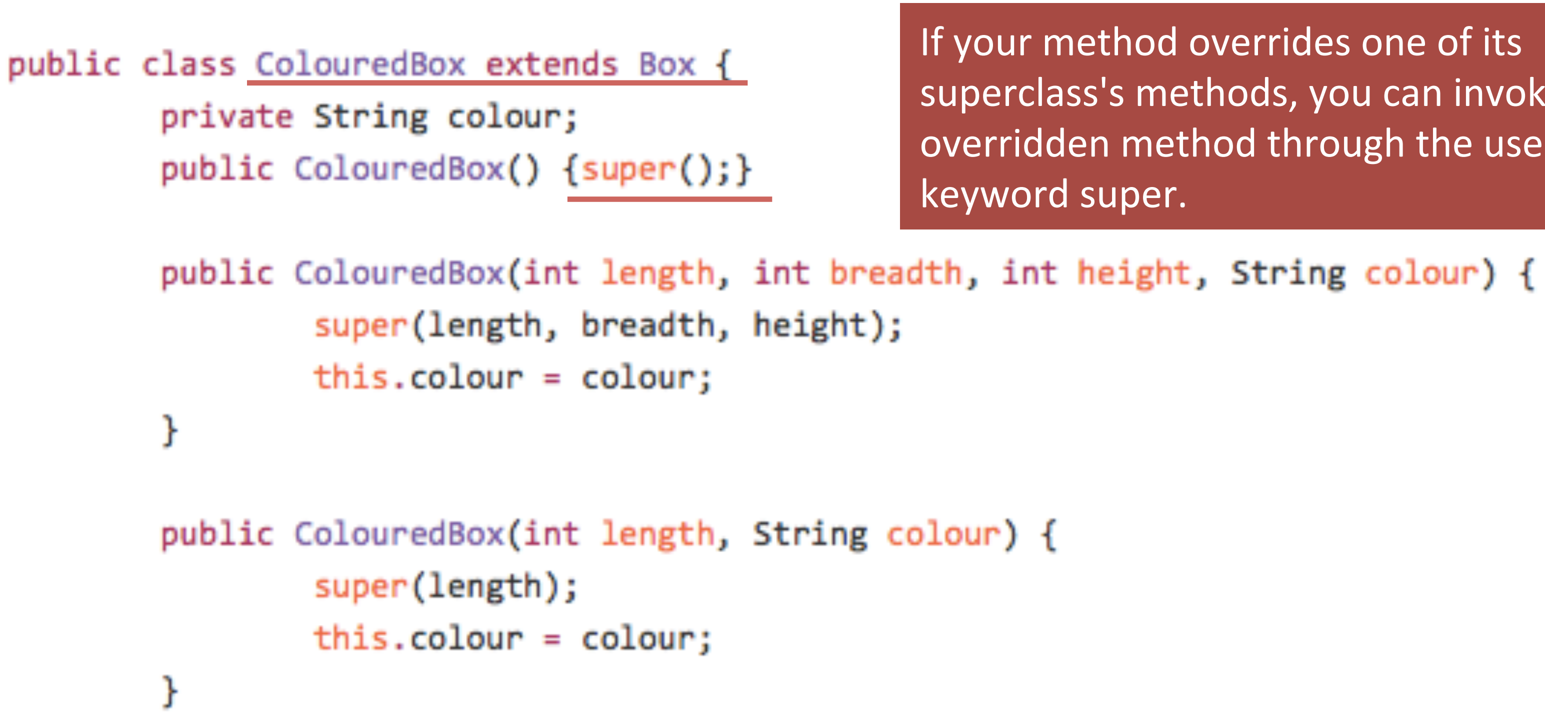
Initialising New Boxes

Calling Methods

Try calling other methods!

02_OOP Box.java

Inheritance

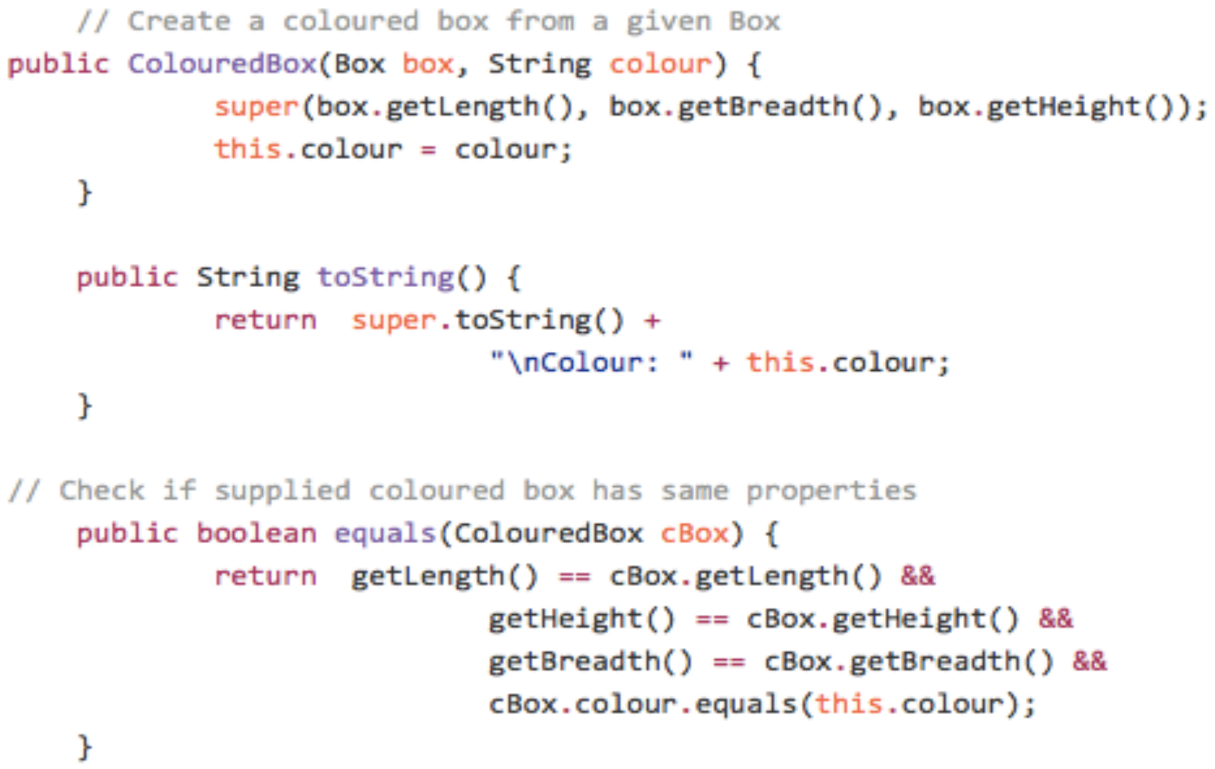


```
public class ColouredBox extends Box {
        private String colour;
        public ColouredBox() {super();}

        public ColouredBox(int length, int breadth, int height, String colour) {
                super(length, breadth, height);
                this.colour = colour;
        }

        public ColouredBox(int length, String colour) {
                super(length);
                this.colour = colour;
        }
```

If your method overrides one of its superclass's methods, you can invoke the overridden method through the use of the keyword super.

02_OOP ColouredBox.java

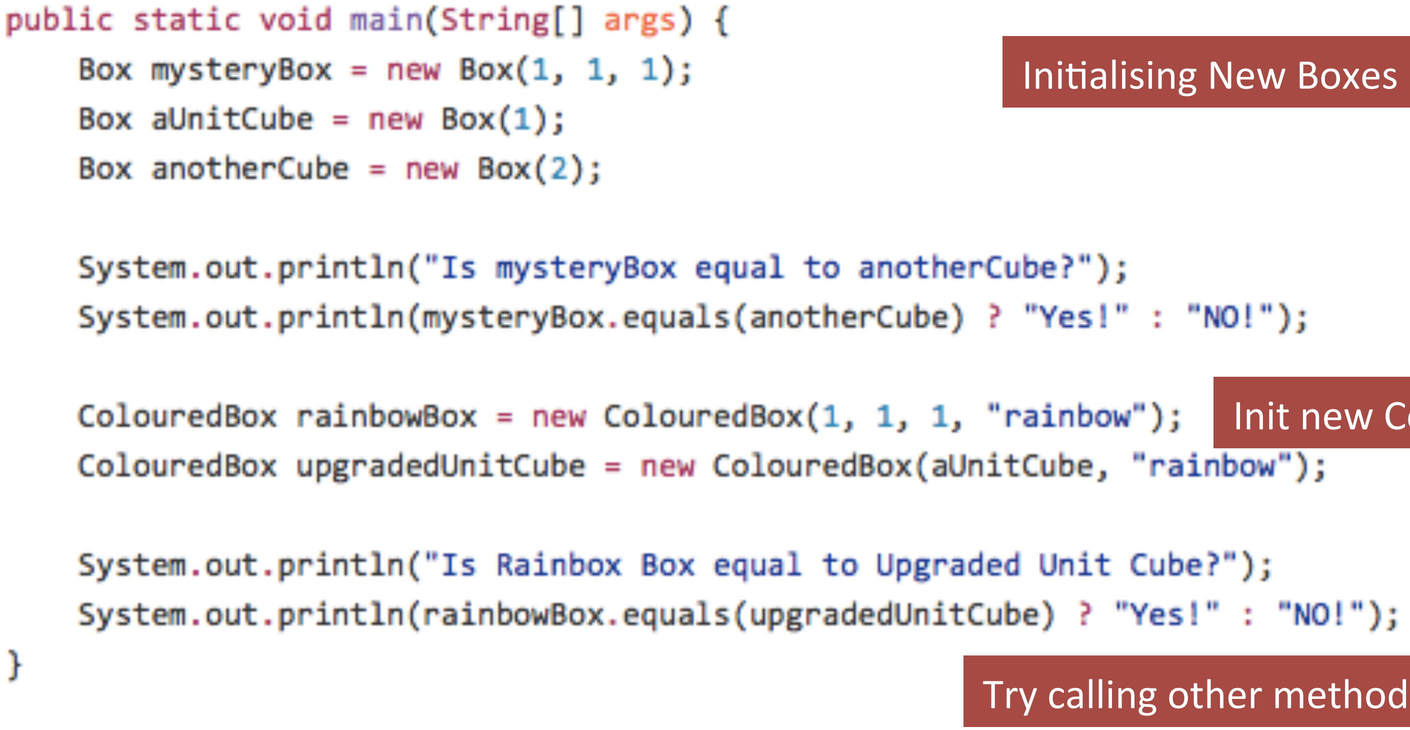
Inheritance & Methods



```
    // Create a coloured box from a given Box
public ColouredBox(Box box, String colour) {
            super(box.getLength(), box.getBreadth(), box.getHeight());
            this.colour = colour;
    }

    public String toString() {
            return  super.toString() +
                            "\nColour: " + this.colour;
    }

// Check if supplied coloured box has same properties
    public boolean equals(ColouredBox cBox) {
            return  getLength() == cBox.getLength() &&
                            getHeight() == cBox.getHeight() &&
                            getBreadth() == cBox.getBreadth() &&
                            cBox.colour.equals(this.colour);
    }
```

02_OOP ColouredBox.java

Main Method



```java
public static void main(String[] args) {
    Box mysteryBox = new Box(1, 1, 1);
    Box aUnitCube = new Box(1);
    Box anotherCube = new Box(2);

    System.out.println("Is mysteryBox equal to anotherCube?");
    System.out.println(mysteryBox.equals(anotherCube) ? "Yes!" : "NO!");

    ColouredBox rainbowBox = new ColouredBox(1, 1, 1, "rainbow");
    ColouredBox upgradedUnitCube = new ColouredBox(aUnitCube, "rainbow");

    System.out.println("Is Rainbox Box equal to Upgraded Unit Cube?");
    System.out.println(rainbowBox.equals(upgradedUnitCube) ? "Yes!" : "NO!");
}
```

Initialising New Boxes

Init new ColouredBox

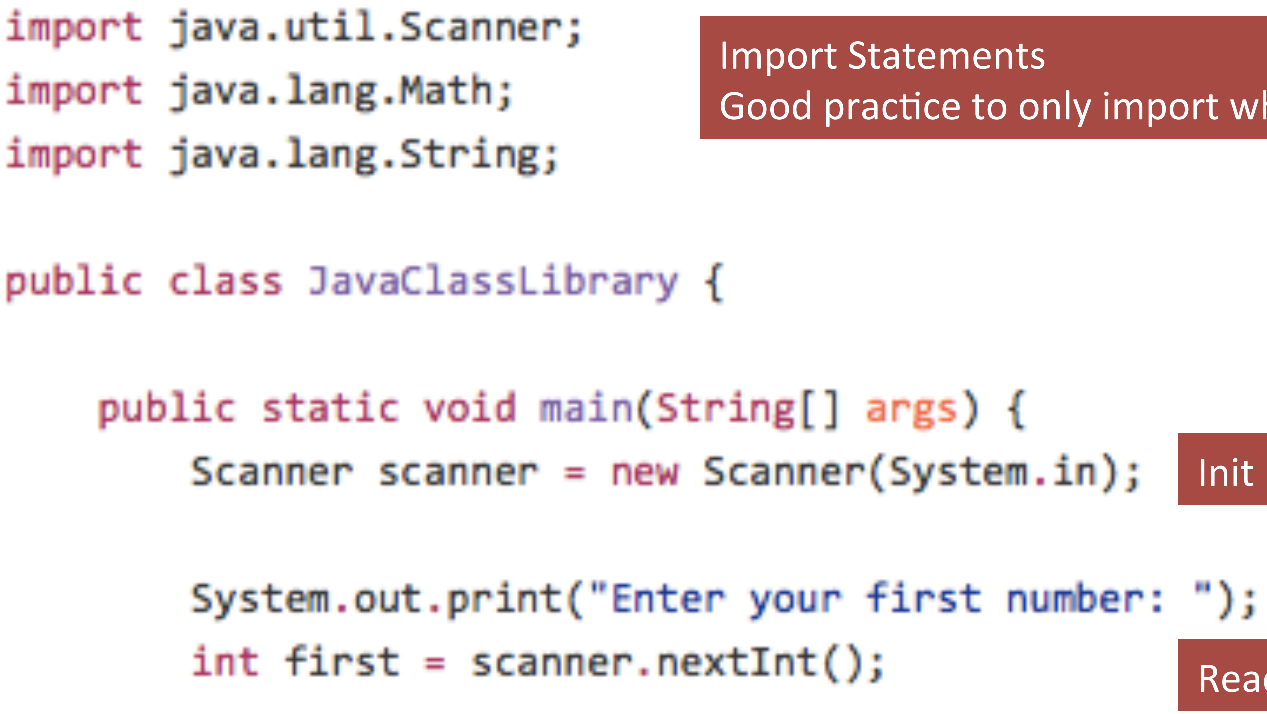
Try calling other methods!

02_OOP ColouredBox.java

03_JAVACLASSLIBRARY

Scanner

Scanner for I/O



```
import java.util.Scanner;
import java.lang.Math;
import java.lang.String;

public class JavaClassLibrary {

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter your first number: ");
        int first = scanner.nextInt();
```

Import Statements
Good practice to only import what you need

Init new Scanner

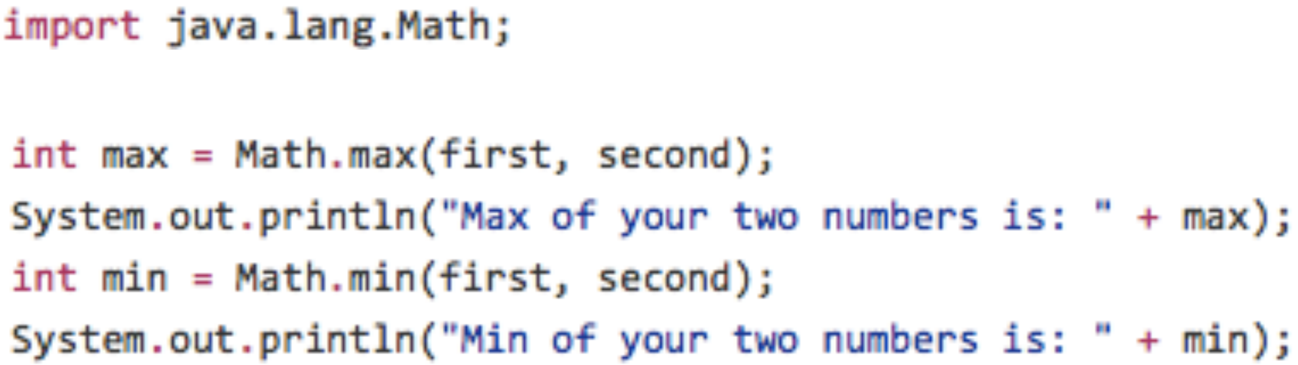
Read next integer

Explore other API – nextDouble()…

03_JavaClassLibrary

Math

Math for… Math

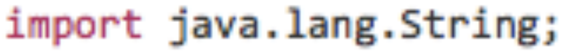


```
import java.lang.Math;

int max = Math.max(first, second);
System.out.println("Max of your two numbers is: " + max);
int min = Math.min(first, second);
System.out.println("Min of your two numbers is: " + min);
```

Explore other API – pow()…

String

For manipulating Strings

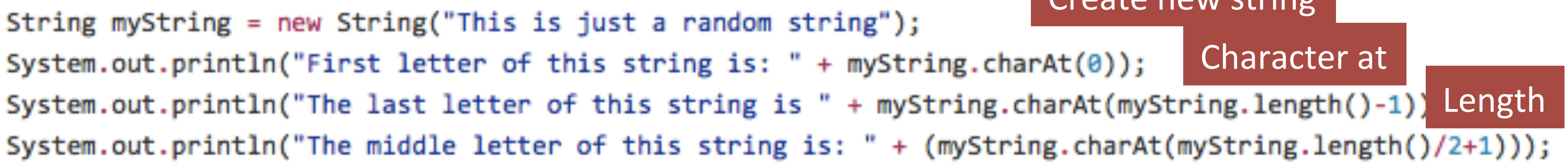


```java
import java.lang.String;
```

Create new string

Character at

Length



```java
String myString = new String("This is just a random string");
System.out.println("First letter of this string is: " + myString.charAt(0));
System.out.println("The last letter of this string is " + myString.charAt(myString.length()-1))
System.out.println("The middle letter of this string is: " + (myString.charAt(myString.length()/2+1)));
```

Explore other API

When In Doubt

- Java Docs

https://docs.oracle.com/javase/7/docs/api/

- Google is still your best friend ☺

The End!

All the Best for CS1020!