

Lecture 11

Java – Hello World Program

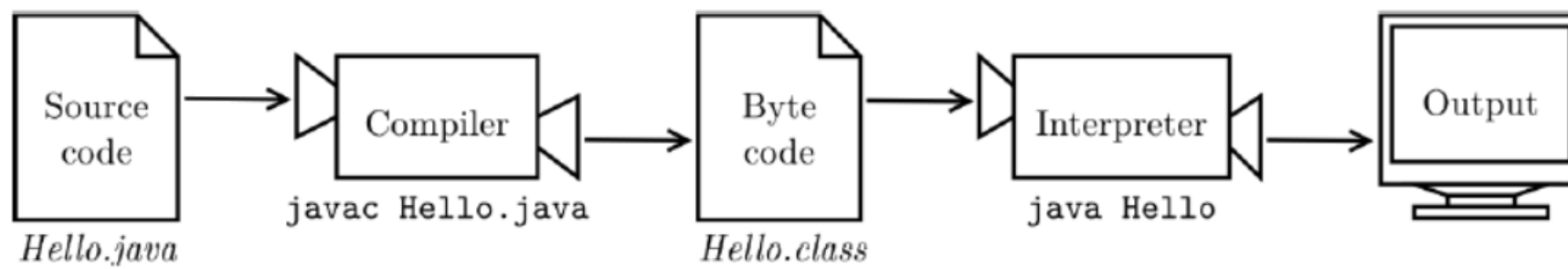
Mostafa Elhosseini

Professor at College of Computer Science and
Engineering

Taibah University

<https://youtube.com/drmelhosseini>

| Review

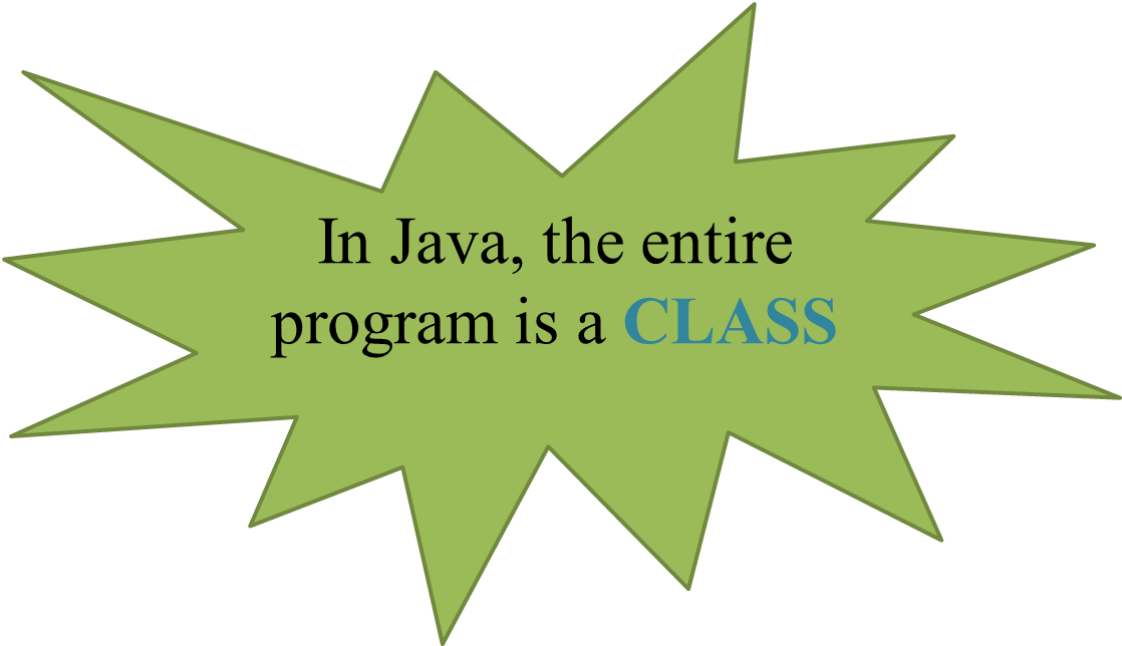


| Hello World Program

- Traditionally, the **first program** you write when learning a new programming language is called the Hello World program. All it does is output the **words Hello, World!** to the screen

| Checking Out Java Code – First Time

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



In Java, the entire
program is a **CLASS**

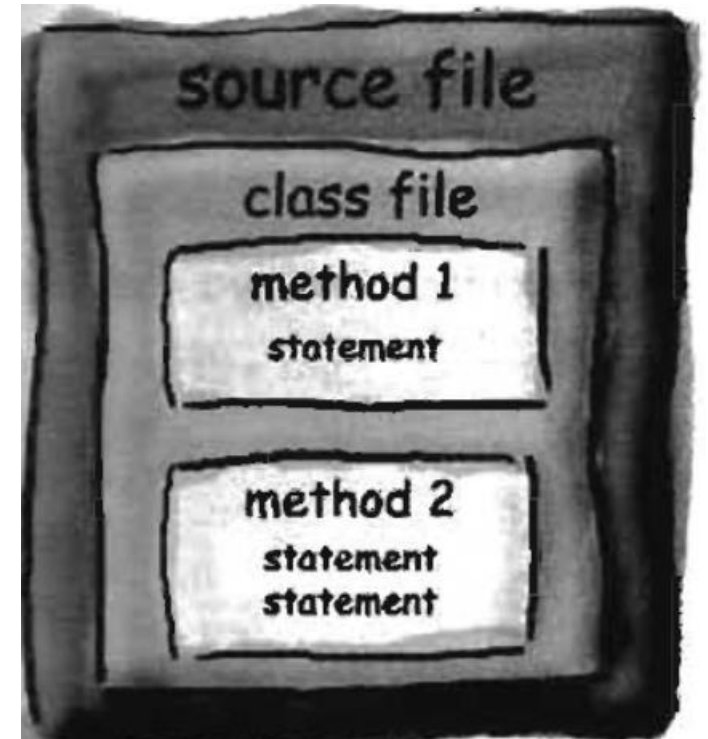
| Hello World Program

- Java programs are made up of **class** and **method** definitions, and **methods are made up of statements**.
- A **statement** is a line of code that performs a basic action.
- In the Hello World program, this line is a **print statement** that **displays a message** to the user

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

| What goes in a source file?

- A **source code file** (with the **.java** extension) **holds one** class definition.
- The **class** represents a **piece of your program**, although a very tiny application might need just a single class.
- The **class** must go within a **pair of curly braces**.



| Anatomy of a class

- When the **JVM starts running**, it **looks for the class** you give it at the **command line**. Then it **starts looking for a specially-written method** that looks exactly like:

```
public static void main(String[] args)
{
    // your code goes here
}
```

- Next, the JVM runs everything between the curly braces { } of your **main** method.
- **Every Java application** has to **have at least one class** and **at least one main** method (not one main per class; just **one main per application**).

| Understanding Simple JAVA Program

- **System.out.println** displays results on the screen; the name **println** stands for print line.
- **public static void main(String[] args)**
 - A **method** is a **named sequence of statements**
 - The **name and format of main** is **special**:
 - when the program runs, it **starts at the first statement in main and ends when it finishes the last statement**

| Understanding Simple JAVA Program

- The **System.out** object—which is predefined for you—is known as the **standard output object**.
- It allows a program to **display information in the command window** from which the program executes
- The **declaration** for the **System.out.println** method is **buried** inside the official Java API

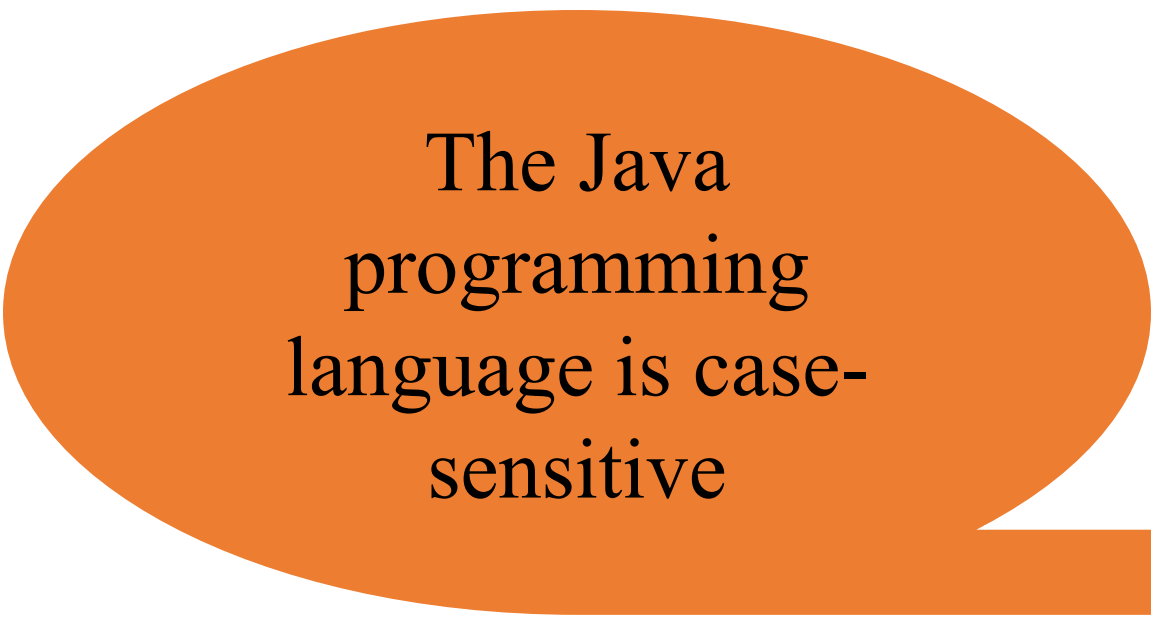
| Understanding Simple JAVA Program

- The **people at Sun Microsystems** who created Java thought of Java as coming in **two parts**. Just as English has its **grammar** and **commonly used names**, the Java programming language has its **specification** (its **grammar**) and its **Application Programming Interface (API)**
 - The **Java Language Specification**: This includes rules like, “**Always put an open parenthesis after the word for**” and “Use an **asterisk to multiply two numbers**.”
 - The **Application Programming Interface**: Java’s Application Programming Interface (**API**) contains **thousands of tools** that were added to Java after the language’s grammar was defined

| API

- These **tools** range from the commonplace to the exotic. For instance, the **tools include a routine named pow** that can raise 5 to the 10th power for you. **JFrame displays a window** on your computer's screen. Other tools **listen for the user's button clicks**, query databases, and do all kinds of useful things.
- The **API is just a set of classes and other tools** that were created by ordinary programmers who happen to participate in the official **JCP** — the **Java Community Process**

| Understanding Simple JAVA Program

An orange speech bubble with a tail pointing towards the bottom right.

The Java
programming
language is case-
sensitive

you can't replace `class` with `Class`. If you do, the whole program stops working

| Understanding Simple JAVA Program

- This program defines a **class named Test**. For now, a class is a collection of methods;
 - You can **give a class any name** you like,
 - but it is **conventional to start with a capital letter**.
 - The **name of the class** has to **match the name of the file** it is in
- Java uses **curly braces { }** to group things together
- **Blank lines, space characters and tabs** can make programs easier to read. Together, they're known as **white space**. The compiler ignores white space - Use **white space to enhance program readability**

| Understanding Simple JAVA Program

- A **comment** is a special section of text inside a program.
- It is text whose purpose is to **help people understand the program**.
- A **comment** is **part of a good program's documentation**.

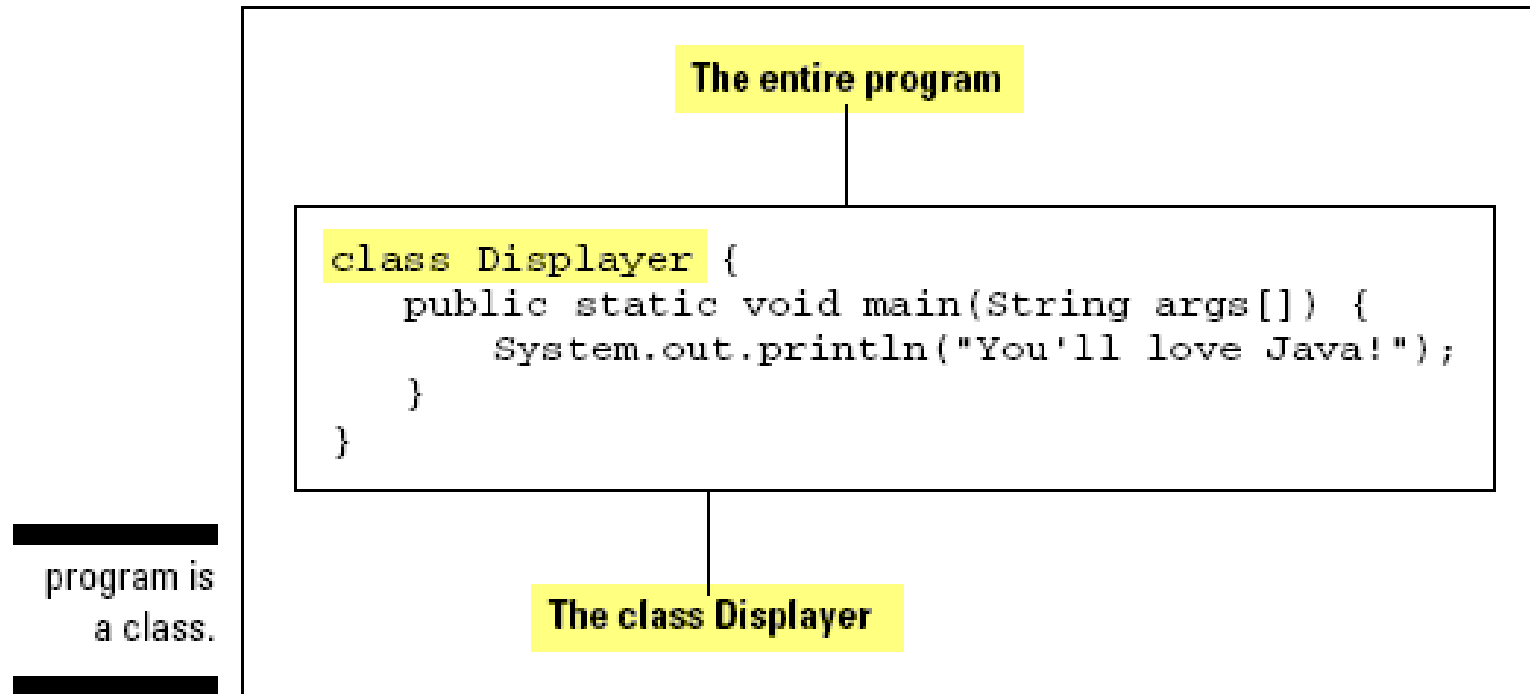
Three Kinds of Comments

```
/*  
 * Listing 3-6 in "Java For Dummies"  
 *  
 * Copyright 2006 Wiley Publishing, Inc. All rights reserved.  
 */  
  
/**  
 * The Displayer class displays text on the computer screen.  
 *  
 * @author Barry Burd  
 * @version 1.6 02/21/06  
 * @see java.lang.System  
 */  
class Displayer {  
  
    /**  
     * Execution of the program starts at this main method.  
     *  
     * @param args (See Chapter 11.)  
     */  
    public static void main(String args[]) {  
        System.out.println("I love Java!"); //Changed to "I"  
    }  
}
```

| Understanding Simple JAVA Program

- The Java programming language has three different kinds of comments:
 - **End of line comments** begins with `//`, indicating that it's an end-of-line comment
 - **Traditional comments**: spread over several lines: These begin with the delimiter `/*` and end with `*/`
 - **Javadoc comments**: delimited by `/**` and `*/`. The **javadoc utility program** (part of the **JDK**) reads Javadoc comments and uses them to prepare program documentation in **HTML5 web-page format**

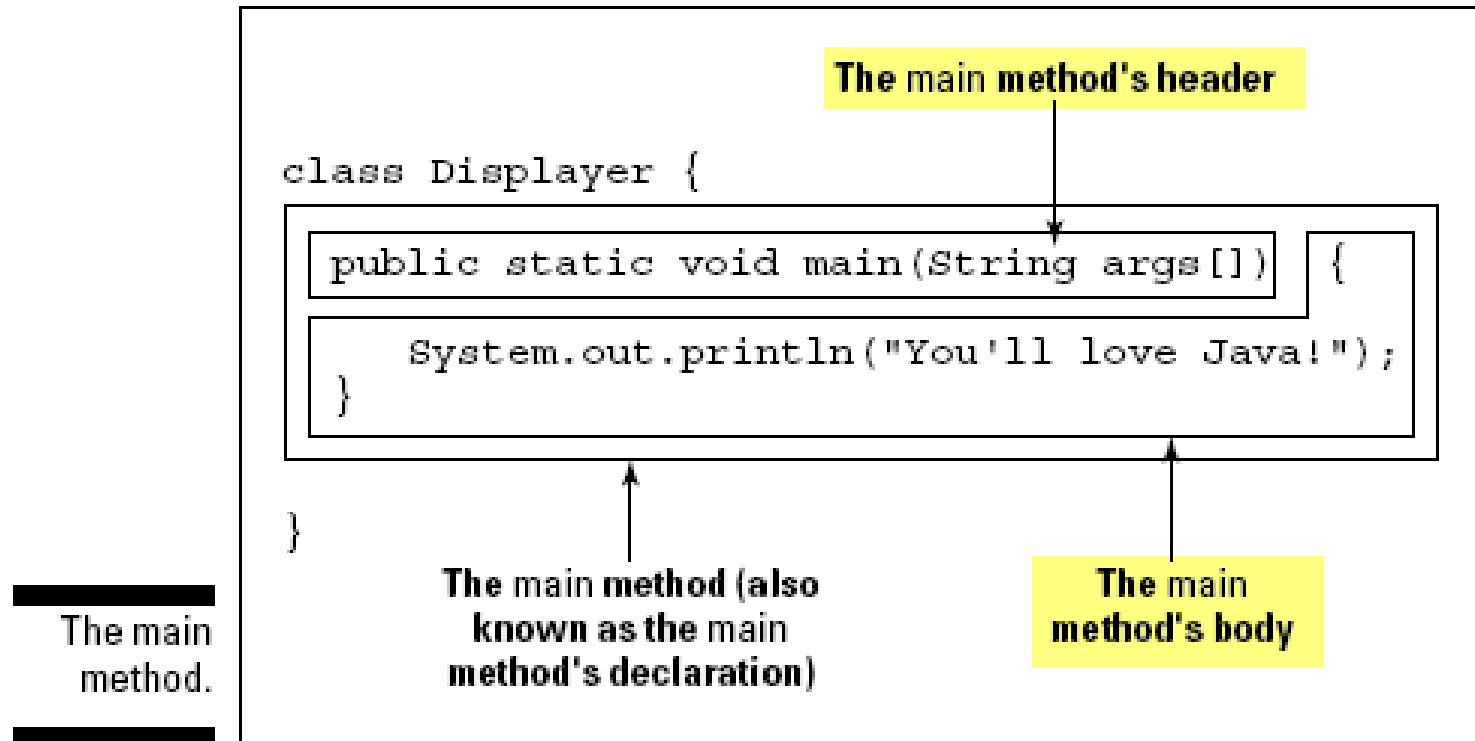
| Understanding Simple JAVA Program



| Understanding Simple JAVA Program

- In Java, a **method** is a **list of things to do**.
- Every method **has a name**, and you tell the computer to do the things in the list by using the method's name in your program
- A **method's declaration** tells the computer what happens if you call the method into action.
- A **method call** (a separate piece of code) tells the computer to actually call the method into action.
- A method's declaration and the method's call **tend to be in different parts** of the Java program

| Understanding Simple JAVA Program



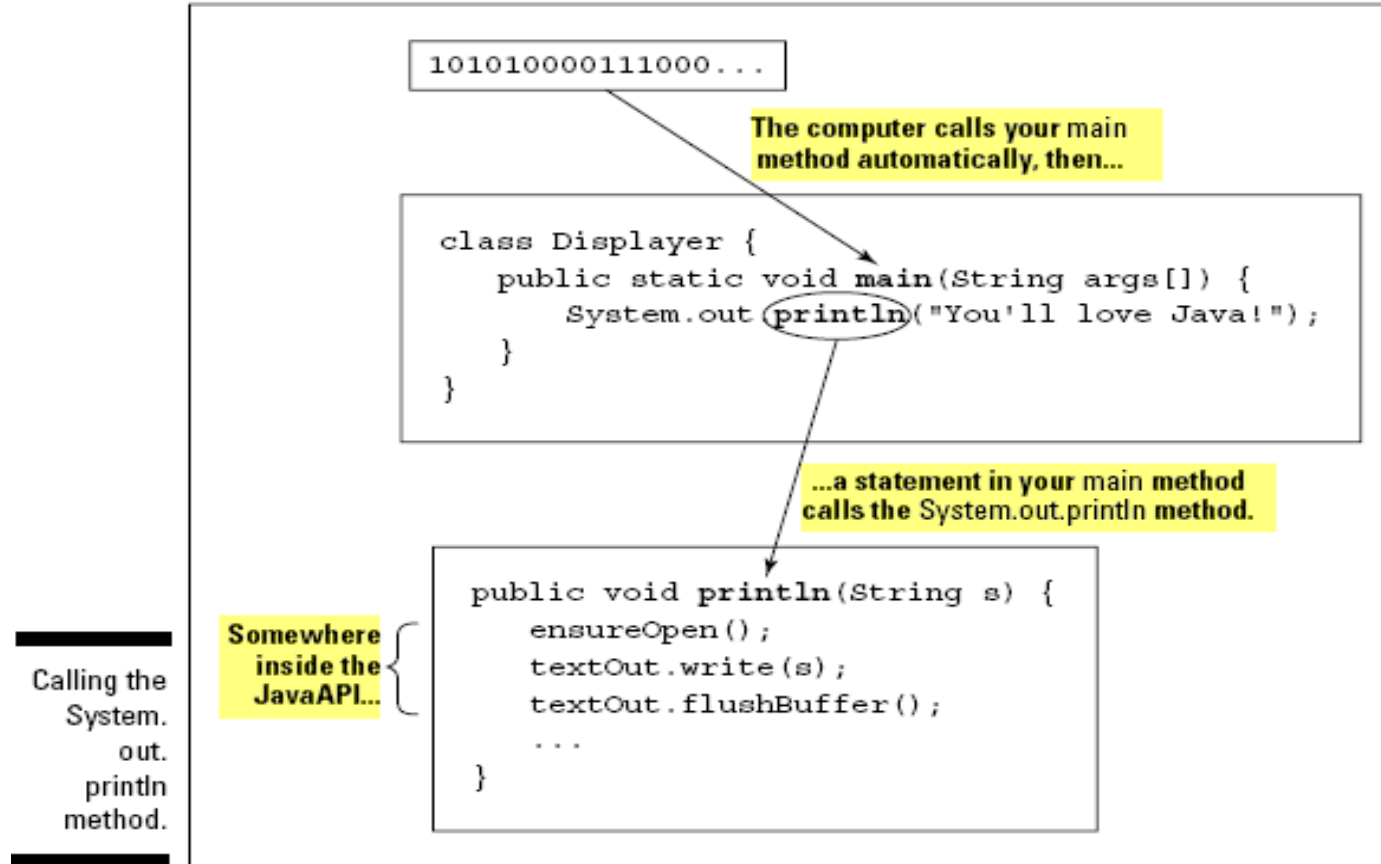
| Understanding Simple JAVA Program

```
class Displayer {  
    public static void main(String args[]) {  
        System.out.println("You'll love Java!");  
    }  
}
```

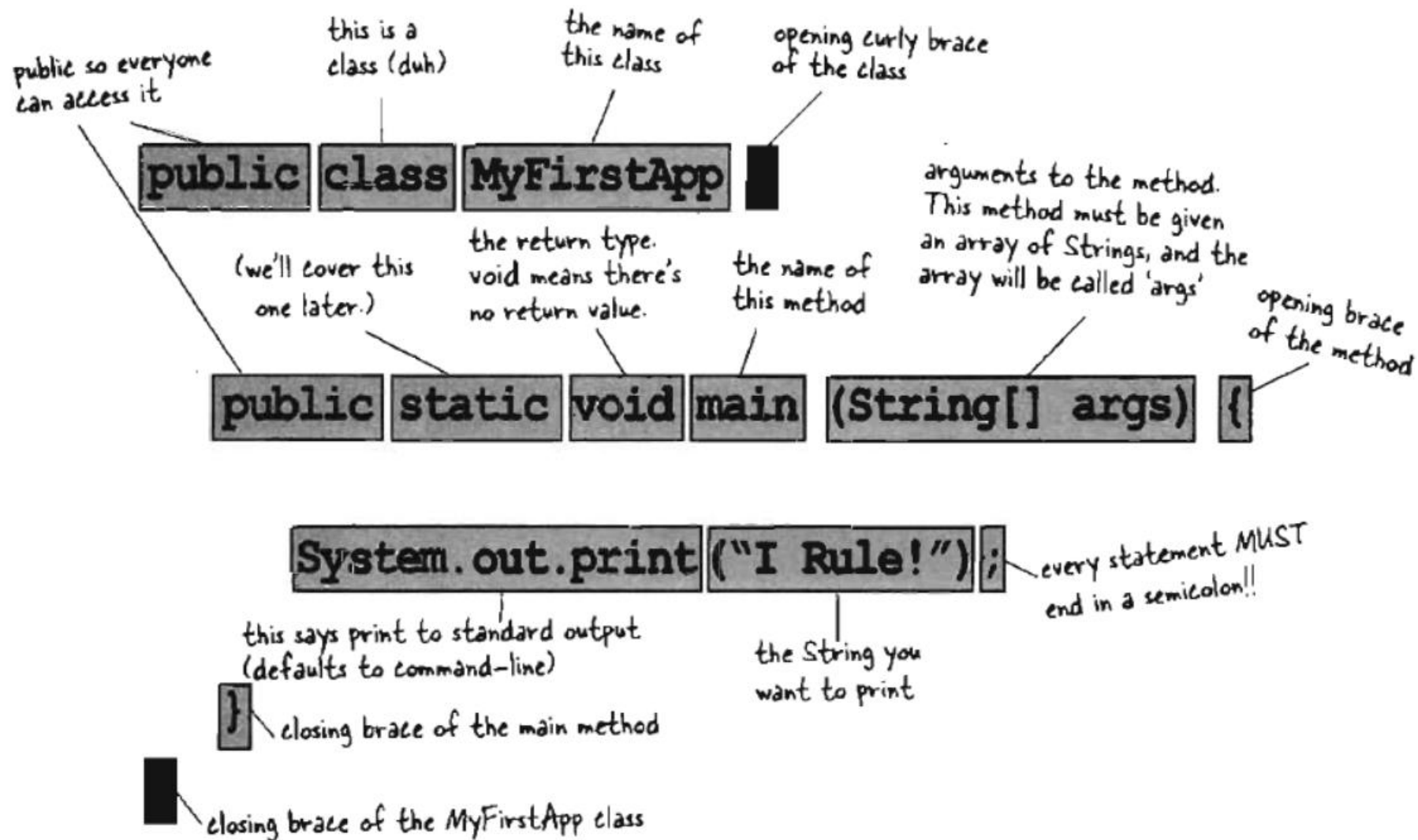
A Java
statement.

A statement (a call to the
System.out.println method)

| Understanding Simple JAVA Program



| Understanding Simple JAVA Program



| References

- Java: How to Program, 9th Edition (Deitel) 9th Edition by Paul Deitel , Harvey M. Deitel
- Head First Java, 2nd Edition 2nd Edition by Kathy Sierra, Bert Bates
- Think Java: How to Think Like a Computer Scientist 1st Edition by Allen B. Downey, Chris Mayfield