Introduction to Java
&
IntelliJ IDEA

COMP2026

PROBLEM SOLVING USING OBJECT ORIENTED PROGRAMMING

Overview

- Lab Arrangement
- Introduction to Java
- Using IntelliJ IDEA
- Standard Output

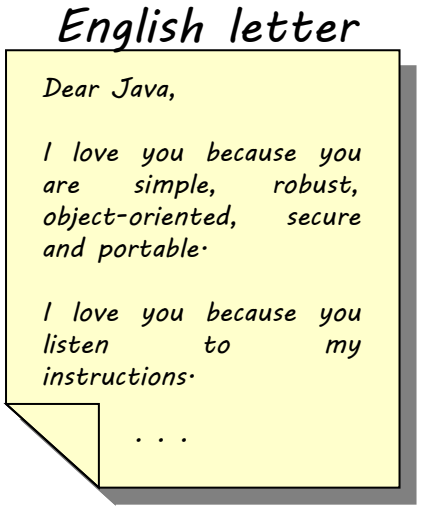
Lab Arrangement

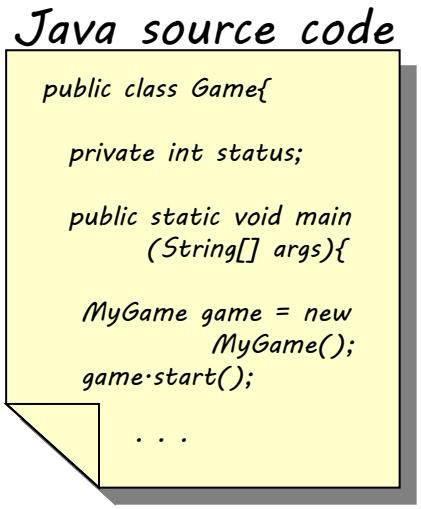
- Discovery Exercises
  - Explore and learn new things!
- Programing Exercises
  - Apply the things learnt from lectures and discovery exercises
- Hints will be given in the slides for **harder problems only**
- **Deadline of Lab Exercise: Next Monday 11:59am**

Introduction to Java

Java

- Similar to human language, programming language has its own grammar (syntax) and sentence structures (statements)

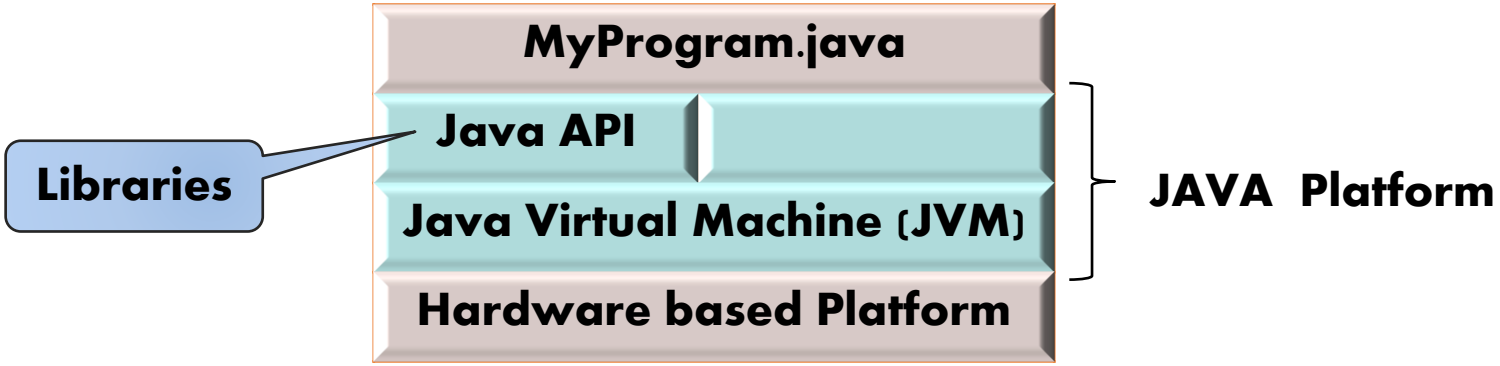




The word "code" or "source code" describes program fragments. E.g. "These four lines of codes"

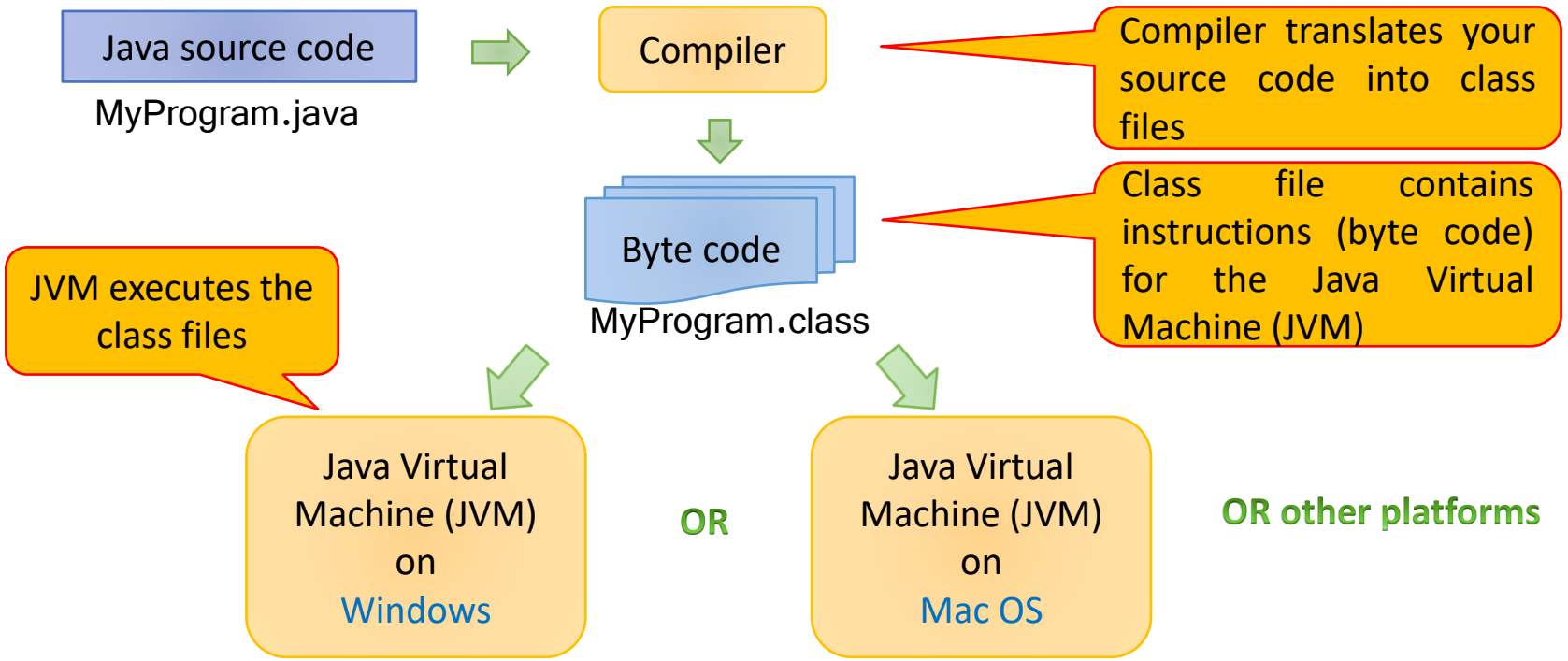
Java

- Java is not just a programming language; it is an entire platform offering a very large library and a powerful execution environment



Java Class Libraries (or Java API) is a collection of pre-existing Java code that provides solutions to common programming problems

Java Program running process



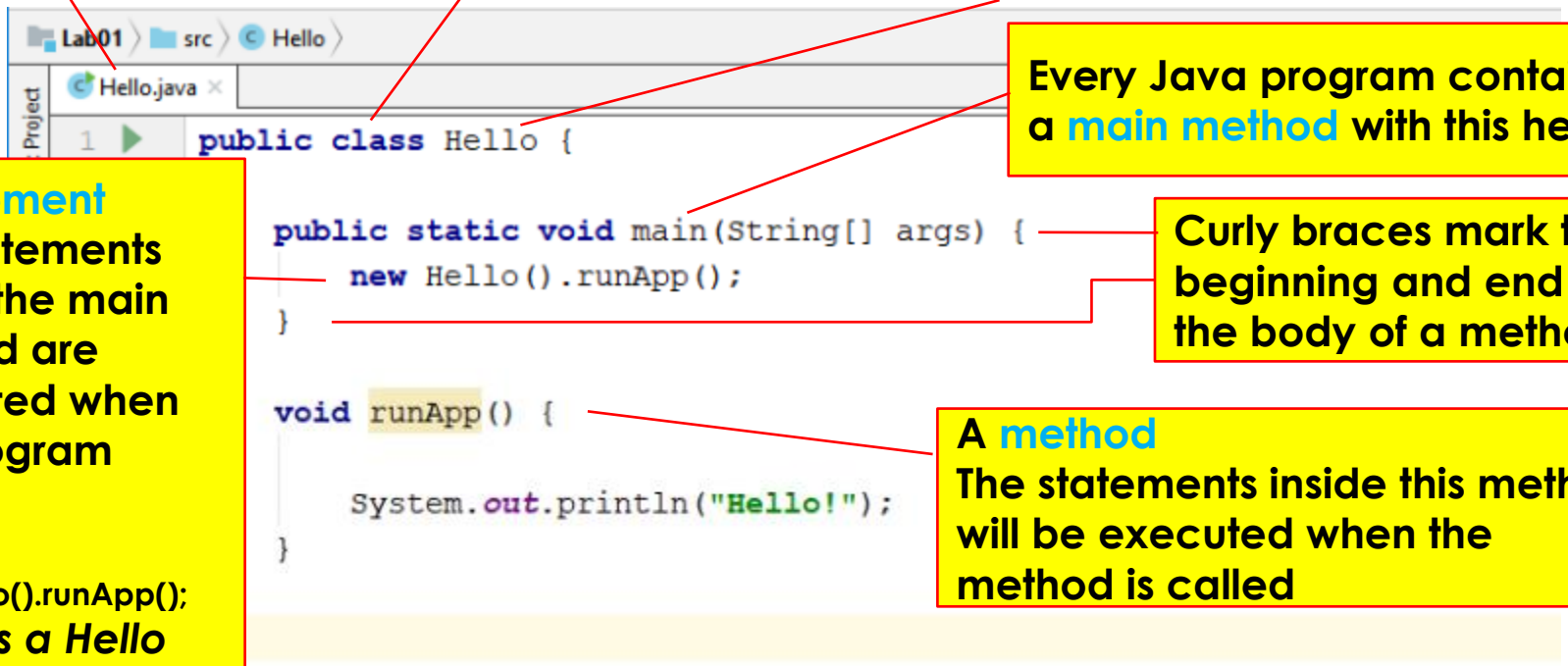
Java is platform-independent
What we need is to install Java Virtual Machine (JVM) on the targeted platform

You can understand this code

Name of the file
It must be the same as the public class name

class keyword
Every program contains at least one class

Name of this class
Java is case-sensitive



Every Java program contains a main method with this header

Curly braces mark the beginning and end of the body of a method

A statement
The statements inside the main method are executed when the program runs

new Hello().runApp();
creates a Hello object and calls the runApp() method)

A method
The statements inside this method will be executed when the method is called

Printing Statement

- In our example program, the runApp() method has a single statement:

```
System.out.println("Hello!");
```

- It prints a line of text, namely "Hello!"
- "System.out.println( )" is a method in the Java library (The programmers who wrote the Java library implement it for us)

Calling a method

- Whenever you call a method in Java, you need to specify

1. The method you want to use (in this case, System,out.println())
2. Any values the method needs to carry out its task (in this case, "Hello!"). The technical term for such a value is an argument

String Literals (Strings)

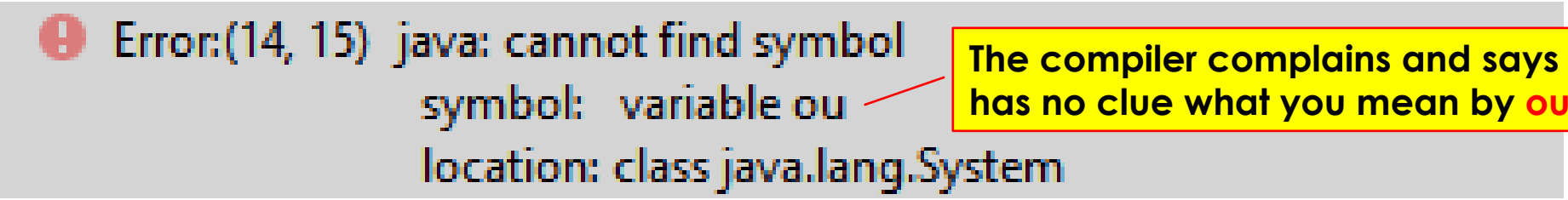
- A sequence of characters enclosed in double quotation marks
- The followings are valid string literals:
  - "Hello!"
  - "It is interesting."
  - ""
  - "123"
- The followings are NOT valid string literals:
  - 'Bad stuff here'
  - `you must use double quotation marks`

Experiment 1

- What happens if you make a typing mistake such as

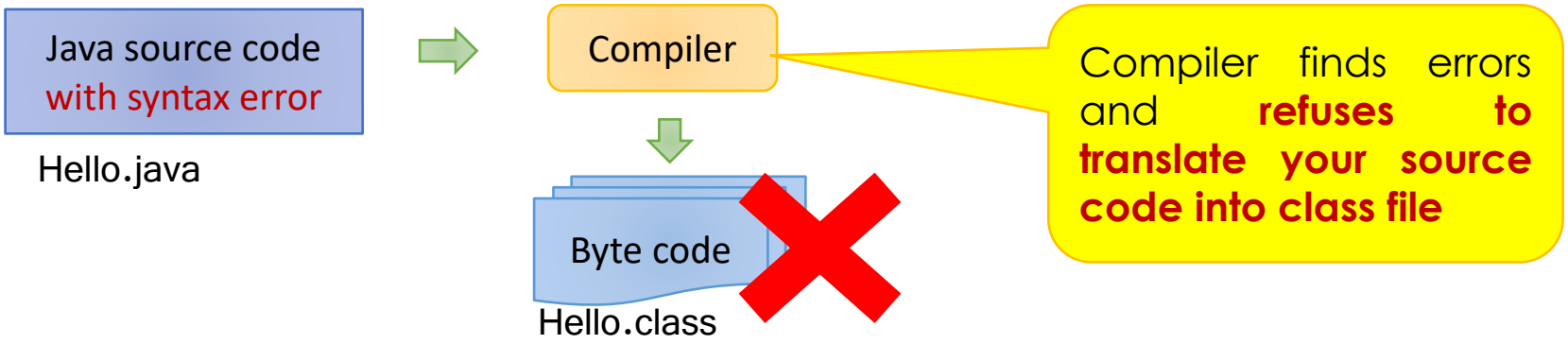
  System.ou.println("Hello!");

- After you click the run button, error messages will be shown at the console



Syntax Error

- This is what we called syntax error



- You must fix the error and compile again

A syntax error is a violation of the programming language rules that is detected by the compiler

Let's do it together!

Experiment 2

- What happens if you make a typing mistake such as

  System.out.println("Hxllo!");

- After you click the run button, the program will compile and run, but the output will be wrong

Logic Error

- This is what we called logic error
- The program is syntactically correct and does something, but it doesn't do what it is supposed to do
- It is caused by logical flaws

- It is your responsibility to find and fix the logic error

A logic error causes a program to take an action that the programmer did not intend

More about Printing Statement

- You can also print numerical values
- For example, the statement

```
System.out.println(3+4);
```

evaluates the expression 3+4 and displays the number 7

Modify the *Hello.java* program to print numerical values

You may use these operators:

+	**plus**
–	**minus**
*	**times**
/	**divide**
%	**modulo *(finding remainder)***

Standard Output

Standard Output Example

StandardOutputExample.java

```
1  public class StandardOutputExample {
2      public static void main(String[] args) {
3          new StandardOutputExample().runApp();
4      }
5
6
7      void runApp() {
8          System.out.print("This ");
9          System.out.print("is ");
10         System.out.println("line 1.");
11         System.out.println("This is line 2.");
12     }
13 }
```

print()
prints the string

println()
prints the string and moves the cursor to a newline

Run: StandardOutputExample

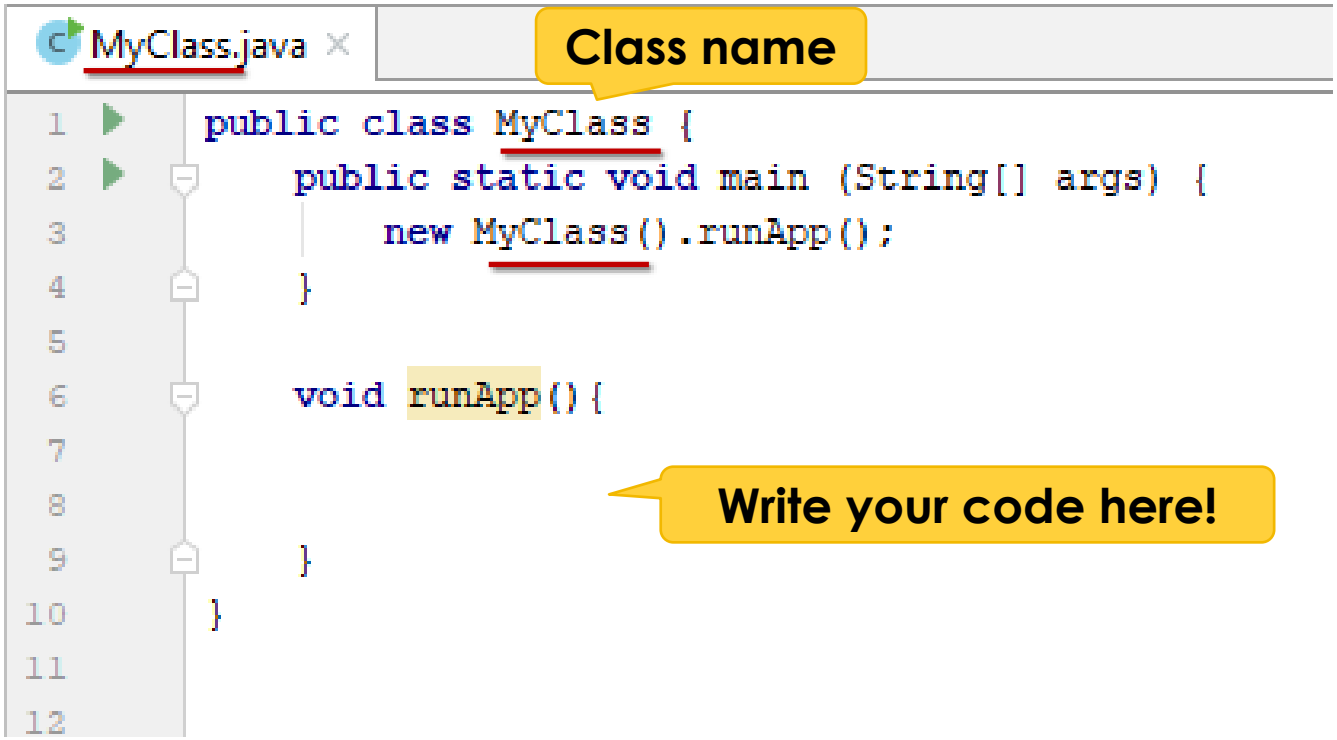
```
"C:\Program Files\Java\jdk1.8.0_181\bin\java.exe" ...
This is line 1.
This is line 2.

Process finished with exit code 0
```

4: Run 6: TODO Terminal

How to start?

- Suppose you are told to write a Java program called MyClass



Lab Exercise Submission

- Submit the following to Moodle
  - **XXXXXXXX_lab01.docx**

*Replace "***XXXXXXXX***" with your **student ID**

Deadline: Next Monday 11:59am

References

- Dean, J., & Dean, R. (2008). Introduction to programming with Java: A problem solving approach. Boston: McGraw-Hill.
- Forouzan, B. A., & Gilberg, R. F. (2007). Computer science: A structured programming approach using C (3rd ed.). Boston, MA: Thomson Course Technology.
- Gaddis, T. (2016). Starting out with Java (6th ed.). Pearson.
- Liang, Y. D. (2013). *Introduction to Java programming: Comprehensive version*. (8th ed.). Pearson.
- Schildt, H. (2006). *Java a beginner's guide*. New York: McGraw Hill.
- Wu, C. T. (2010). *An introduction to object-oriented programming with Java*. Boston: McGraw Hill Higher Education
- Xavier, C. (2011). Java programming: A practical approach. New Delhi: Tata McGraw Hill.
- Zakhour, S., Kannan, S., & Gallardo, R. (2013). The Java tutorial: A short course on the basics (5th ed.).
- yet another insignificant Programming Notes. (n.d.). Retrieved from https://www3.ntu.edu.sg/home/ehchua/programming