

Object Oriented Programming with Java

Zheng Chen



Object Oriented Programming with Java



- Java Basics
- Object Oriented Programming with Java
- Java Advanced Programming
- Java GUI Programming

Staff Info



- Name: Zheng Chen
- Office: 211-1008
- Email: chenzheng.prc@163.com

“I hear and I forget; I see and I remember; I do
and I understand.”

– Confucius



Chapter One. Java Overview



Java Overview

1. Java History
2. Great Features of Java
3. Java Development Toolkit
4. Java Program Structure
5. Hello Java World
6. Java Virtual Machine



**KEEP
CALM
AND
CODE
JAVA**

Java Overview — Java History

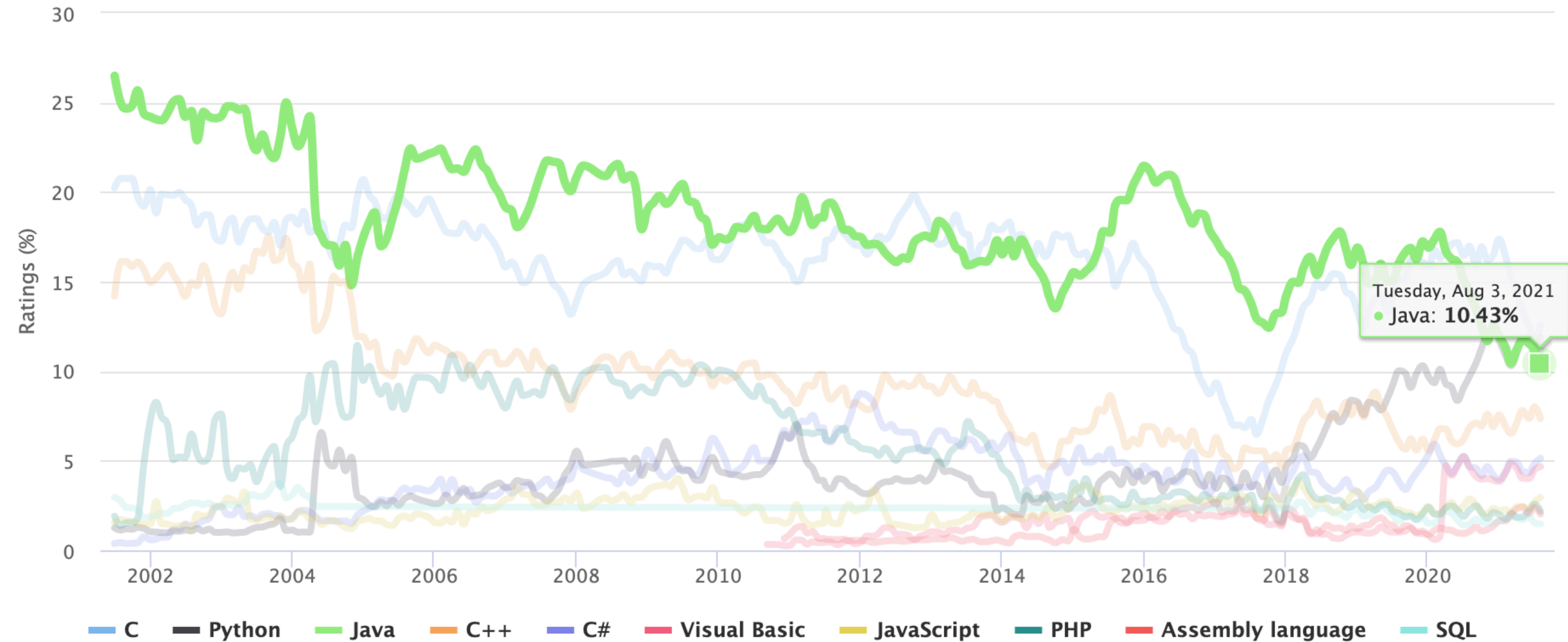
- Java was started as a project called "Oak" by James Gosling in June 1991.
- The first public implementation was Java 1.0 in 1995.
- Java 2 (released initially in 1998) had multiple configurations built for different types of platforms, which are **J2SE**, **J2EE**, and **J2ME**
- 2006, Sun released the Java HotSpot virtual machine and compiler as free software
- By now, the newest Java version is **JavaSE 16**

Java History — Java Editions

- **J2SE**(Java 2 Standard Edition) - to develop client-side standalone applications or applets.
- **J2ME**(Java 2 Micro Edition) - to develop applications for mobile devices such as cell phones.
- **J2EE**(Java 2 Enterprise Edition) - to develop server-side applications such as Java servlets and Java ServerPages.

TIOBE Programming Community Index

Source: www.tiobe.com



The most popular programming language most of the time.

Java in the last years

Java Overview

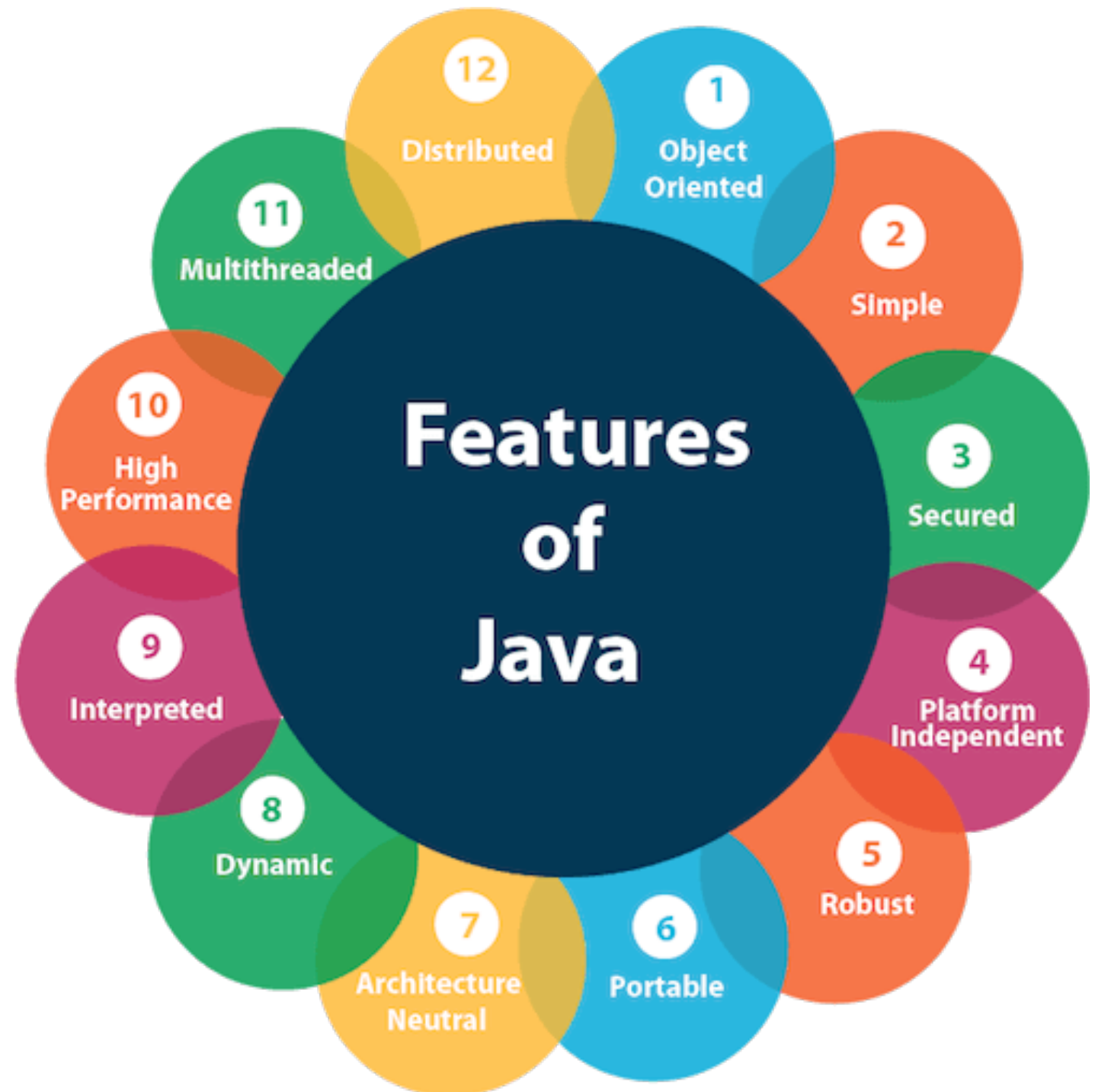
1. Java History
- 2. Great Features of Java**
3. Java Development Toolkit
4. Java Program Structure
5. Hello Java World
6. Java Virtual Machine



**KEEP
CALM
AND
CODE
JAVA**

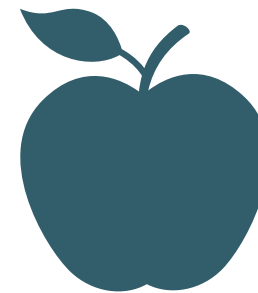
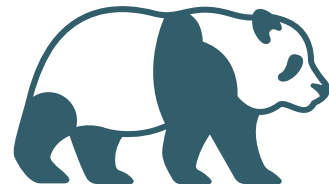
Java Overview — Java Features

- Object Oriented
- Platform Independent
- Distributed

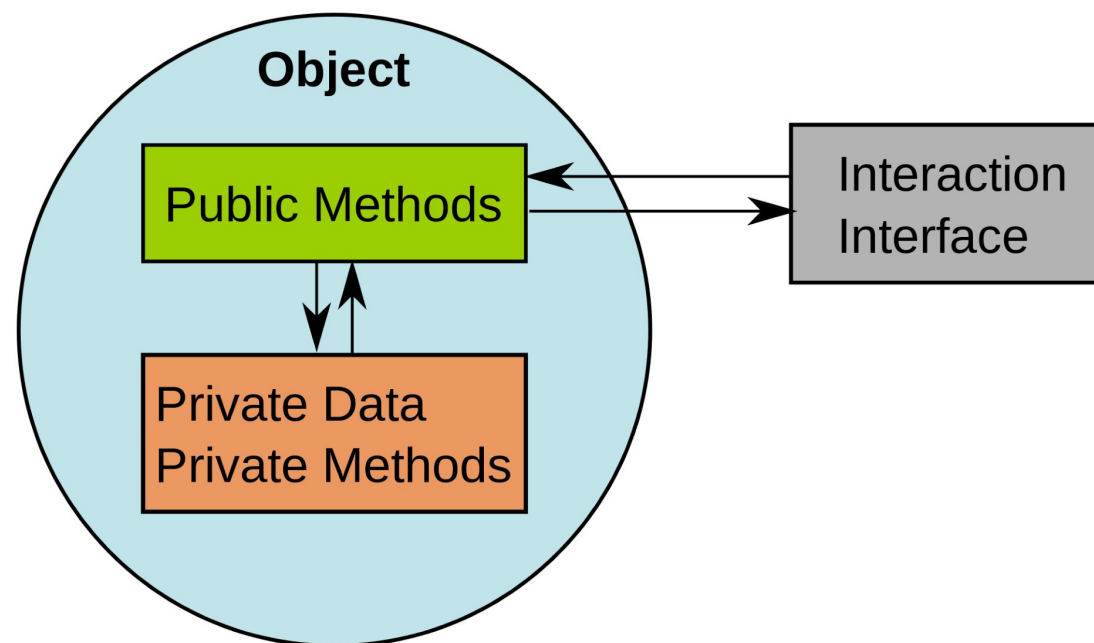


Java Features — Object Oriented

- World is made up by objects.

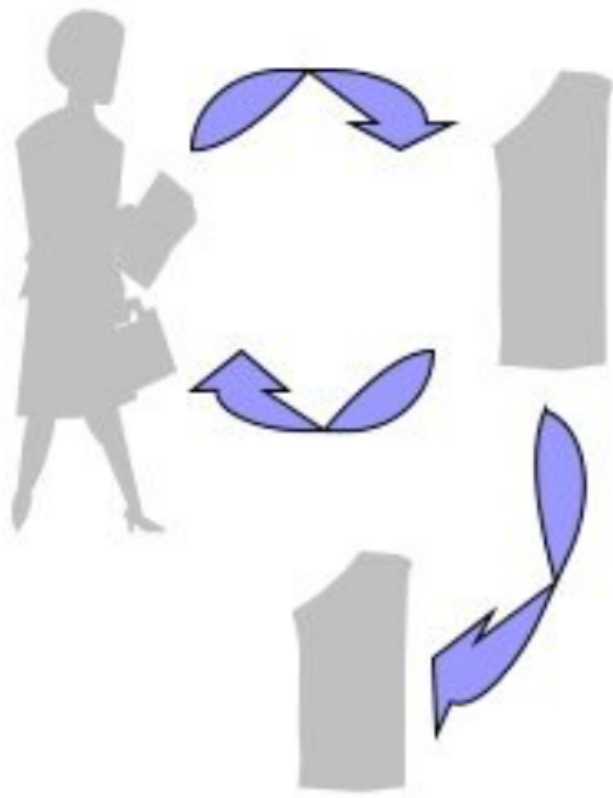


- Program is the reflection of the real world.



Object Oriented vs Procedural

- Procedural



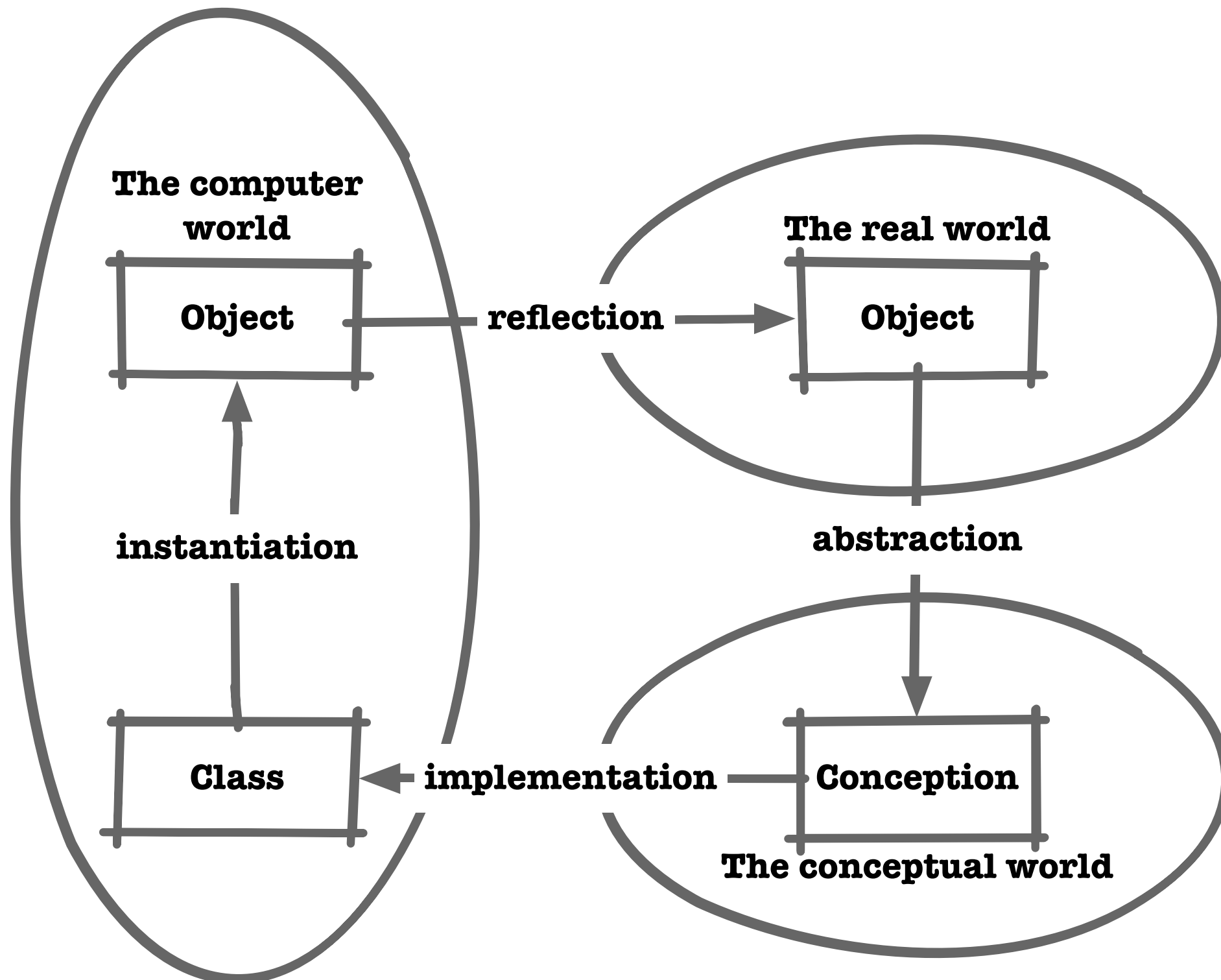
Withdraw, deposit, transfer

- Object Oriented



Customer, money, bank

Object Oriented — Explanation



Object Oriented — Concrete example

class name

Light

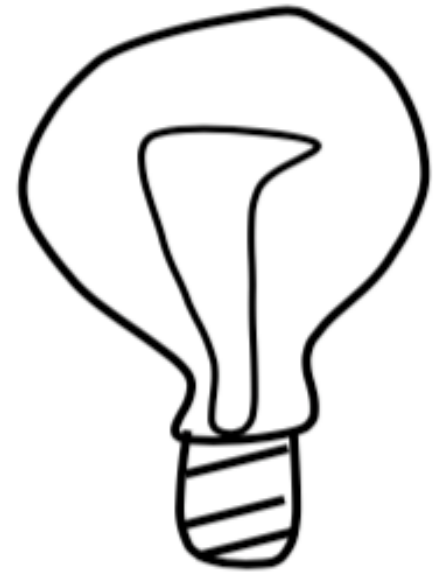
class interface

on()

off()

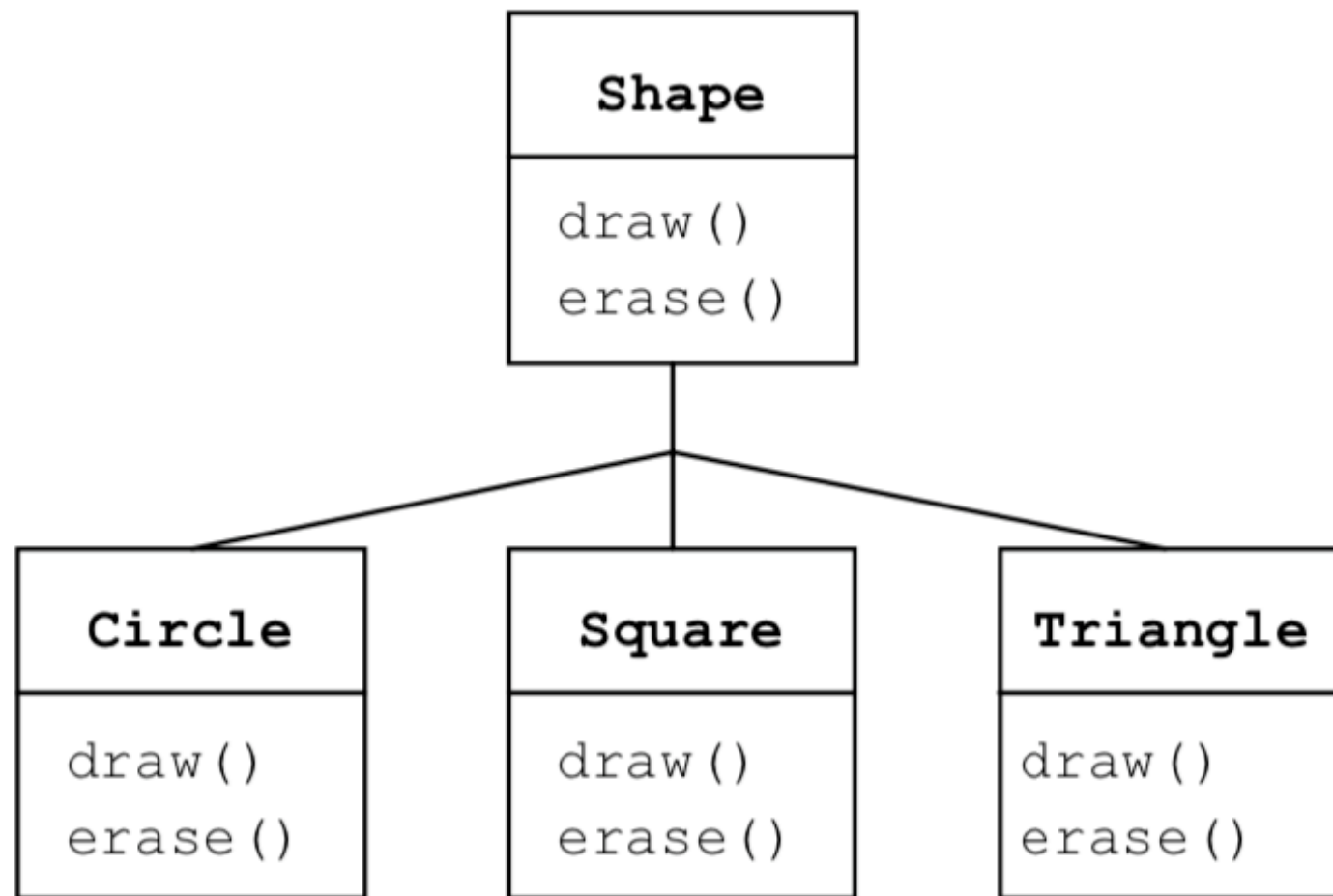
brighten()

dim()



```
Light lt = new Light();  
lt.on();
```

Object Oriented — Inheritance and Polymorphism



```
void doStuff (Shape s) {  
    s.erase ();  
    //...  
    s.draw ();  
}
```

```
Circle c = new Circle();  
Triangle t = new Triangle();  
Square s = new Square();  
doStuff(c);  
doStuff(t);  
doStuff(s);
```

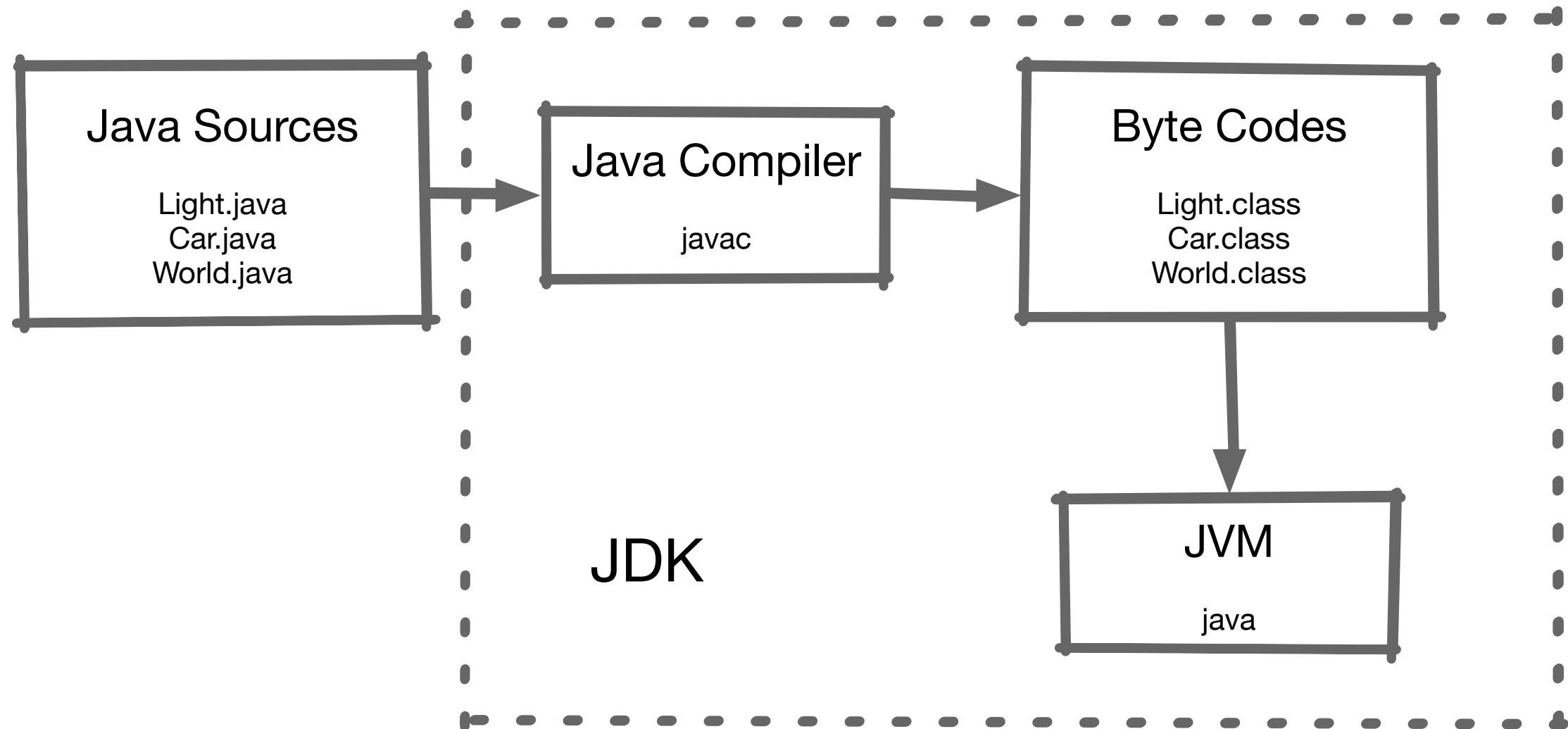

Java Overview

1. Java History
2. Great Features of Java
- 3. Java Development Toolkit**
4. Java Program Structure
5. Hello Java World
6. Java Virtual Machine



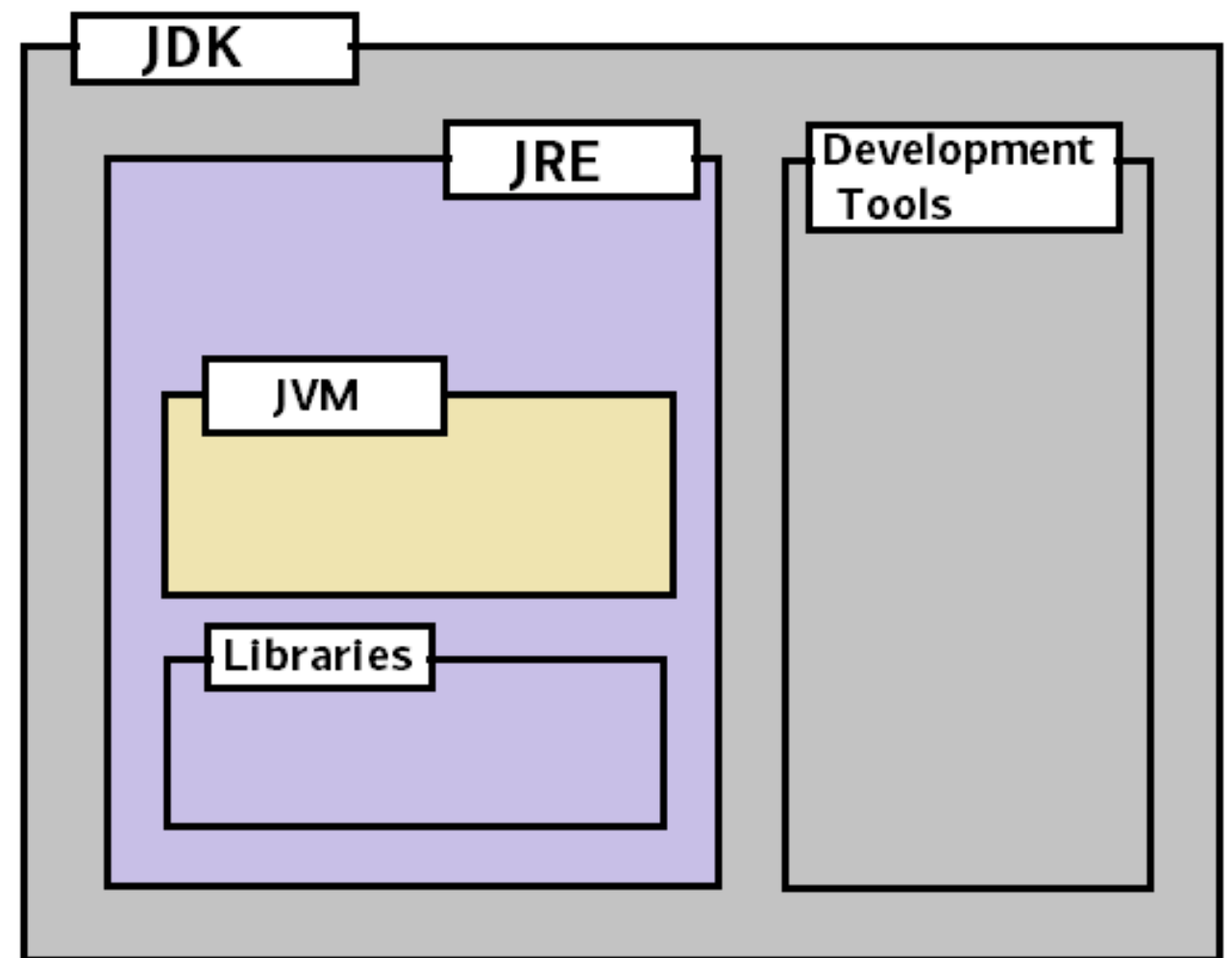
KEEP
CALM
AND
CODE
JAVA

Java Overview — Java Development Toolkit

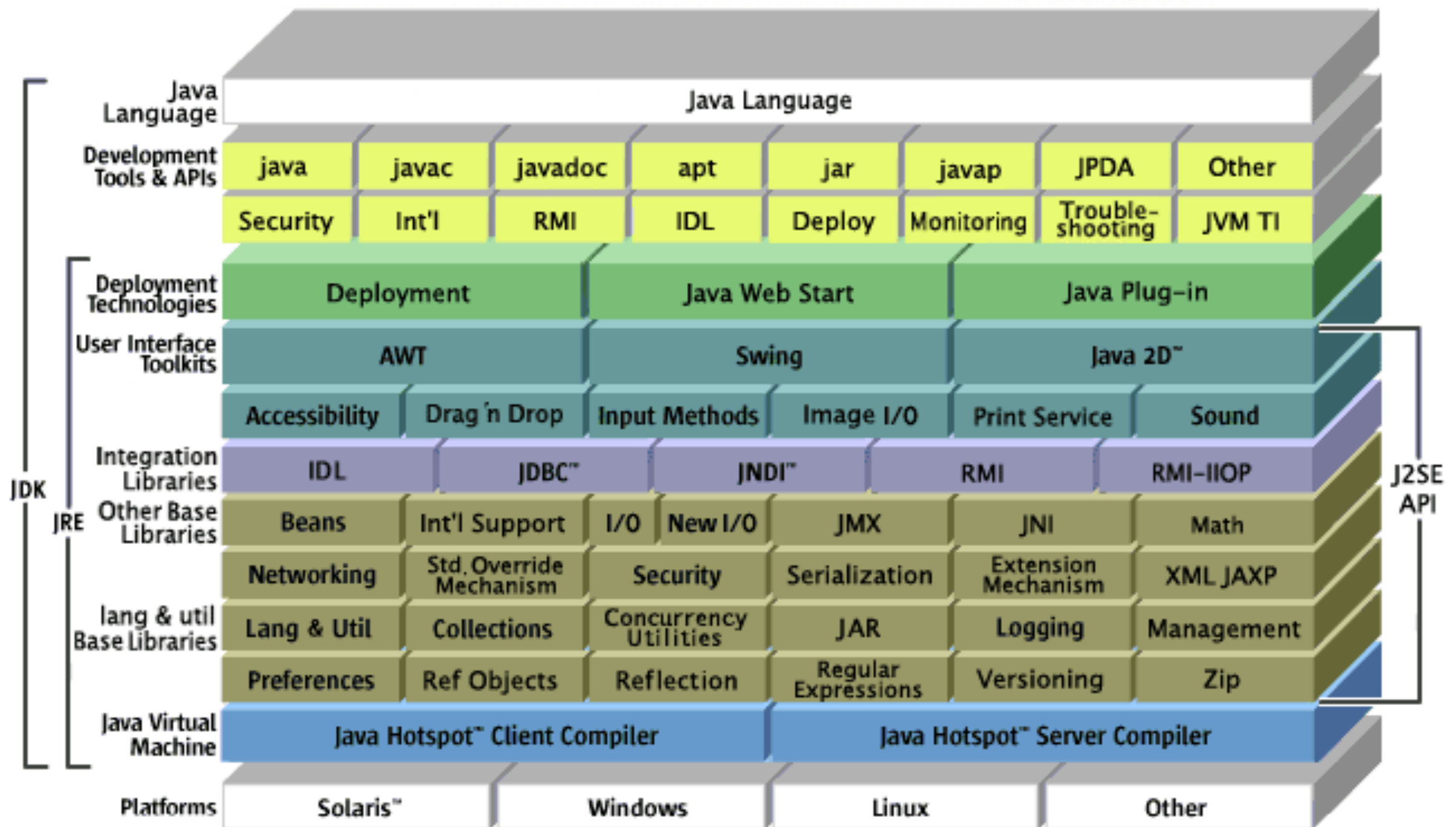


Java Overview — Java Development Toolkit

- JDK
 - Java Development Toolkit
- JRE
 - Java Runtime Environment
- JVM
 - Java Virtual machine

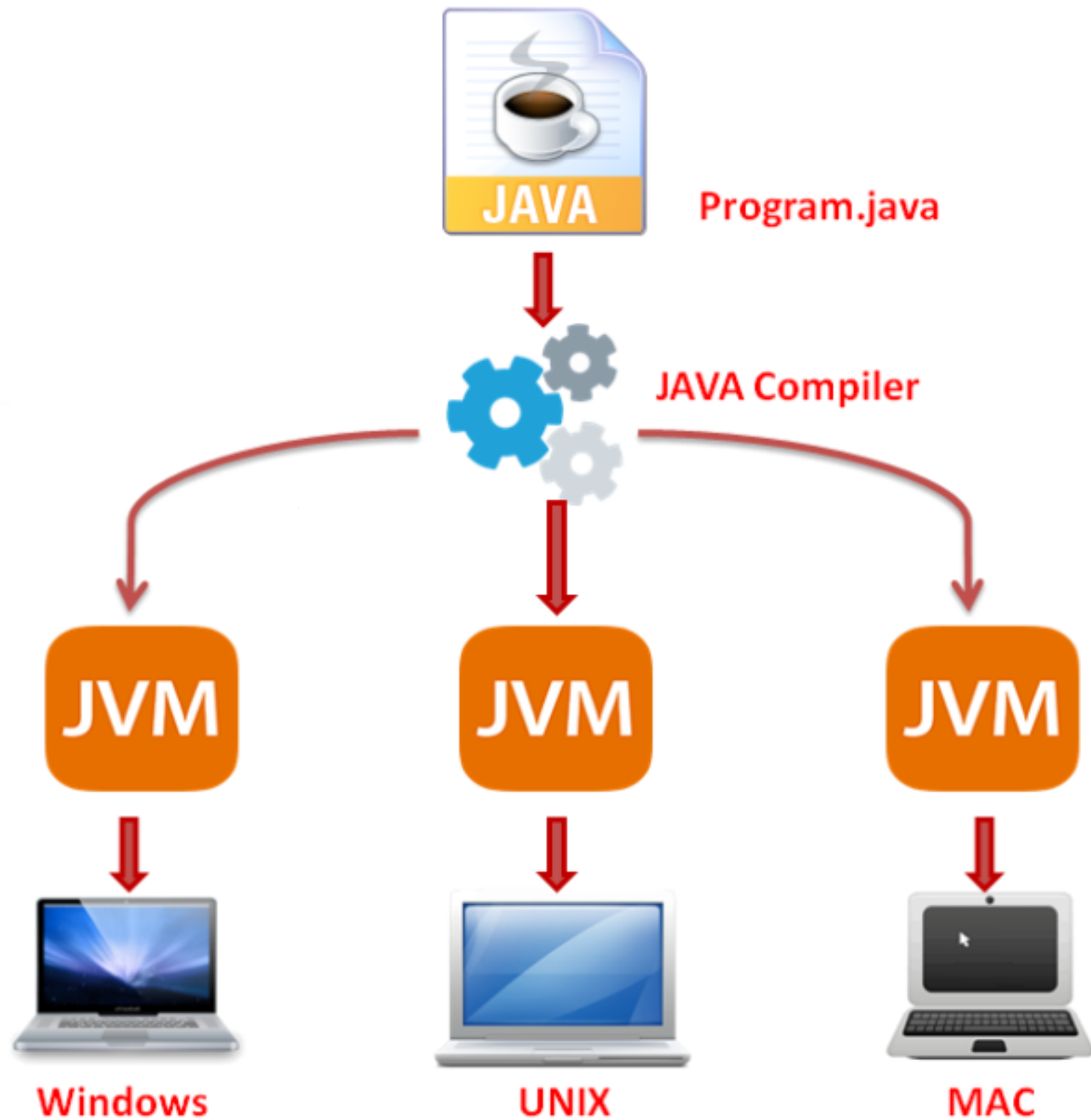


Java™ Platform, Standard Edition (Java SE)



Java Platform at a Glance

JDK Overview



Write Once, Run Anywhere

The Java Declaration

Java Overview — Java Development Toolkit

- Download and Installation
- PATH setup
- CLASSPATH setup
- Commands
 - javac, java, appletviewer, javadoc

Java Overview

1. Java History
2. Great Features of Java
3. Java Development Toolkit
- 4. Java Program Structure**
5. Hello Java World
6. Java Virtual Machine



**KEEP
CALM
AND
CODE
JAVA**

Java Overview — Java Program Structure

- In the Java programming language:
 - A program is made up of one or more classes
 - A class contains one or more methods
 - A method contains program statements
- These terms will be explored in detail throughout the course
- A Java application always contains a method called main

Java Program Structure — Lincoln.java

```
public class Lincoln
{
    //-----
    // Prints a presidential quote.
    //-----
    public static void main (String[] args)
    {
        System.out.println ("A quote by Abraham Lincoln:");
        System.out.println ("Whatever you are, be a good one.");
    }
}
```

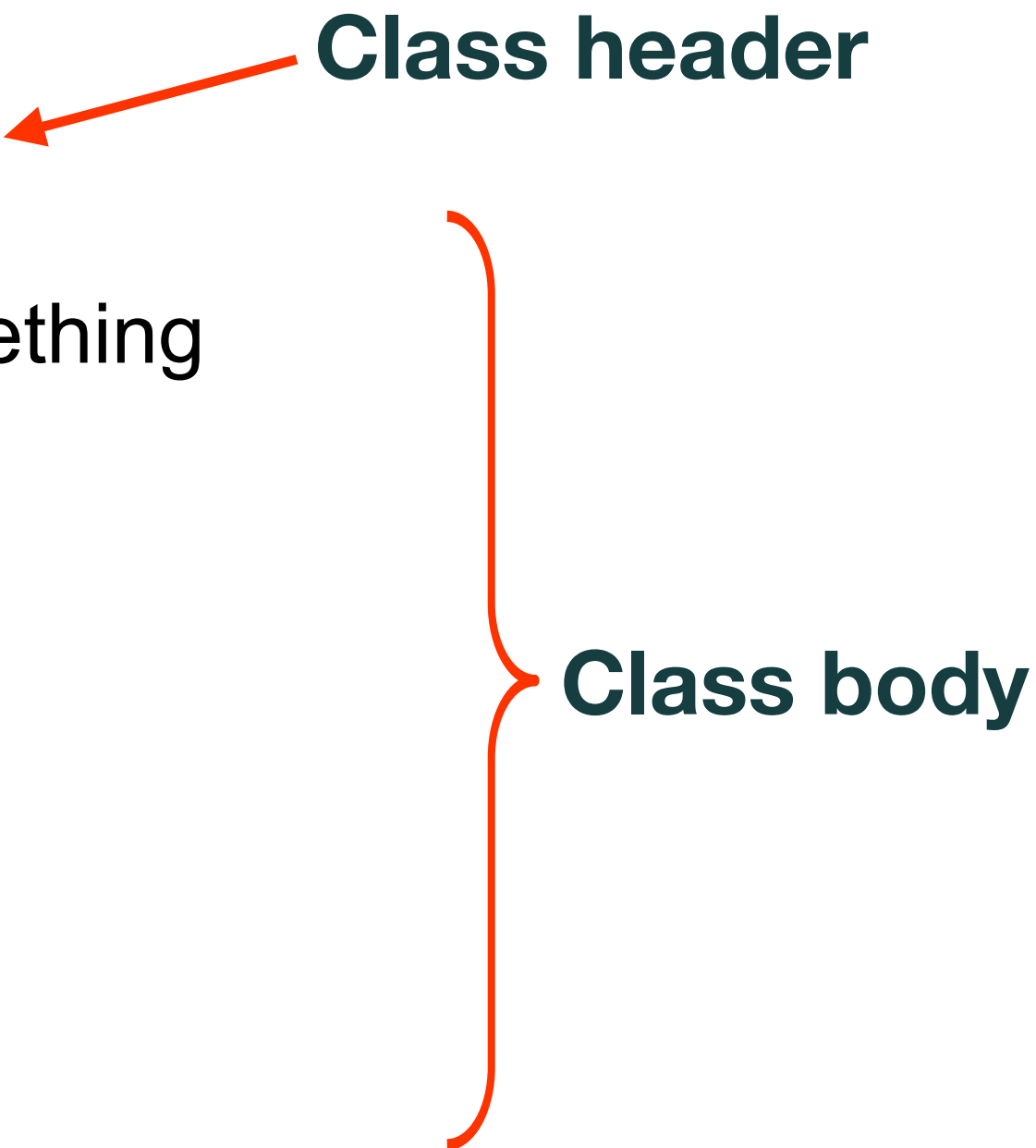
Java Program Structure — Class Structure

```
public class MyProgram
{
    //comments about something

    ...
    Statements...
    ...
}
```

Class header

Class body

The diagram illustrates the structure of a Java class. It shows a code snippet with two annotations. An orange arrow points from the text 'Class header' to the line 'public class MyProgram'. A large orange curly bracket on the right side of the code, spanning from the opening curly brace '{' to the closing curly brace '}', is labeled 'Class body'.

Java Program Structure — Method Structure

```
public class Lincoln  
{
```

```
    //comments about something
```

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

```
{
```

```
    ...
```

```
    Statements...
```

```
    ...
```

```
}
```

```
}
```

Method header



Method body



Java Program Structure — Package and Import

Package statement

```
package packagename;  
import OtherClassName;  
import ...
```

Imports statements

```
public class MyProgram  
{  
    //comments about something  
    ...  
    Statements...  
    ...  
}
```

Java Overview

1. Java History
2. Great Features of Java
3. Java Development Toolkit
4. Java Program Structure
- 5. Hello Java World**
6. Java Virtual Machine



**KEEP
CALM
AND
CODE
JAVA**

Java Overview — Hello World~

```
class HelloWorldApp {  
    public static void main(String args[]){  
        System.out.println("Hello, Java World !");  
    }//end of main method  
}//end of class
```

```
>>>javac HelloWorldApp.java
```

```
>>>java HelloWorldApp
```

Java Overview — Hello GUI~

```
import java.awt.*;
import java.awt.event.*;

public class HelloGUI {
    public static void main(String args[]) {
        Frame frame = new Frame("HelloGUI");
        Button button = new Button("Clicke me!");
        button.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                button.setLabel("Hello~~");
            }
        });
        frame.add(button);
        frame.pack();
        frame.setSize(200, 200);
        frame.setVisible(true);
    }
}
```

```
>>> javac HelloGUI.java
```

```
>>> java HelloGUI
```



Java Overview — Byebye Applet ~

```
import java.applet.Applet;
import java.awt.Graphics;
public class ByebyeApplet extends Applet{
    public void paint(Graphics g){
        g.drawString("Byebye, Java Applet ~ ",50,25);
    }
}
```

```
>>> javac ByebyeApplet.java
```

```
<HTML>
  <BODY>
    <APPLET CODE="HelloJavaApp.class" WIDTH=500 HEIGHT=200>
    </APPLET>
  </BODY>
</HTML>
```

```
>>> AppletViewer ByebyeApplet.html
```


Java Overview

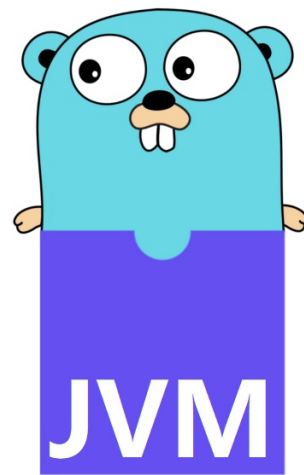
1. Java History
2. Great Features of Java
3. Java Development Toolkit
4. Java Program Structure
5. Hello Java World
6. **Java Virtual Machine**

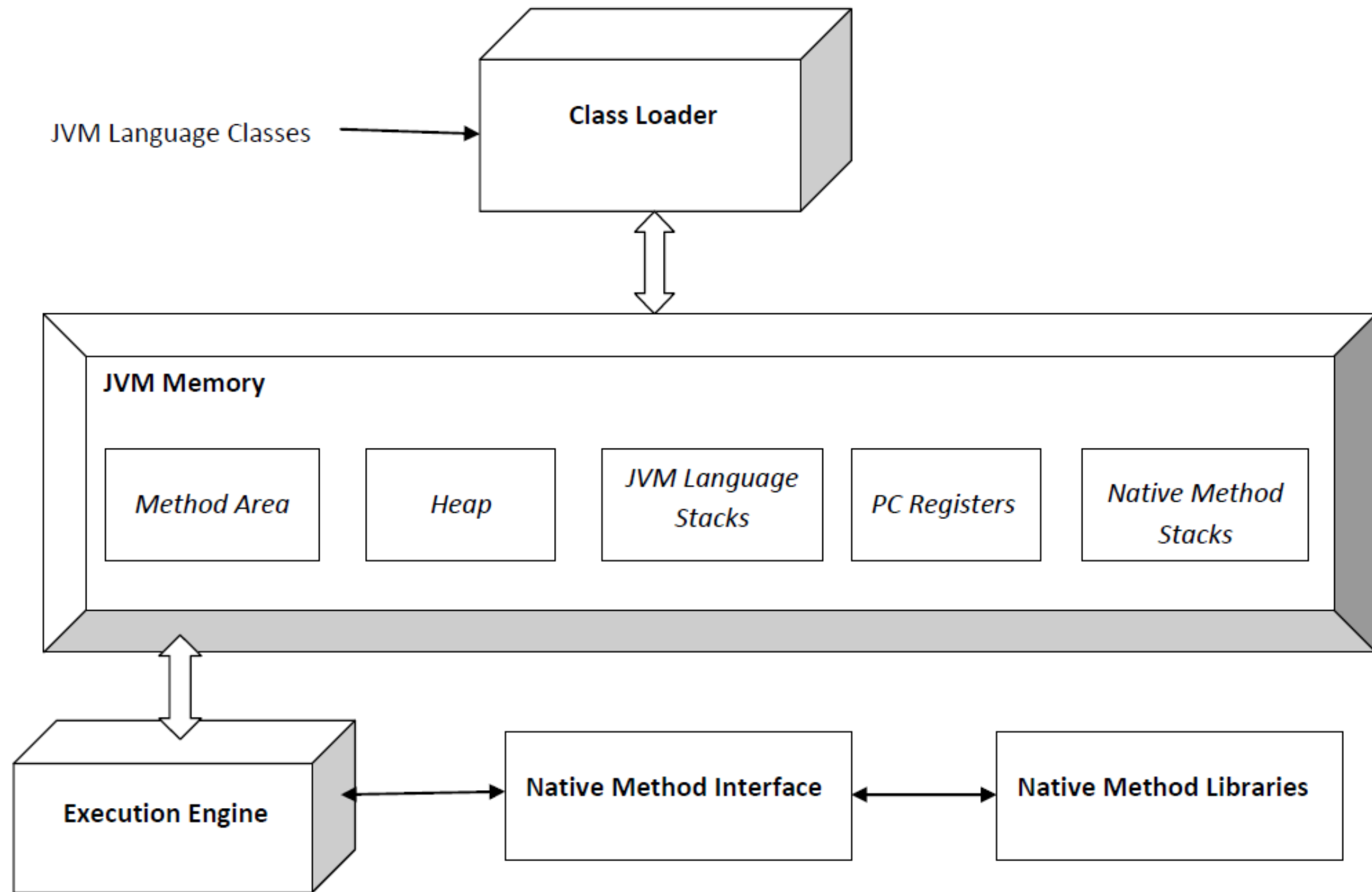


KEEP
CALM
AND
CODE
JAVA

Java Overview — Java Virtual Machine

- A Java virtual machine (JVM) is a virtual machine that enables a computer to run Java programs as well as programs written in other languages that are also compiled to Java bytecode.





Overview of a JVM architecture
based on The Java Virtual Machine
Specification Java SE 7 Edition

Java Virtual Machine

Java Virtual Machine — Features

- + Garbage collection
- + Security
- + Bytecode interpreter and Just-In-Time compiler
- Platform dependence

Homework

Setup Java and write a Helloworld.

