



# Chapter 0 Introduction



Xiang Zhang  
javaseu@163.com  
<https://wdsseu.github.io/java/>



# Something about Teaching in English

2

- Motivation
  - For a better preparation for bilingual communication in your long career;
- Something special
  - We will teach and study at a slow pace;
  - Teaching / Question Answering / Examination in English;
    - ✦ For teaching, ~100% English;
    - ✦ For question answering, up to you;
    - ✦ For examination, 100% English;

# About QA

3

- I will ask you a lot of questions in this class;
  - Some questions are open questions;
- Voluntarily answering questions will gain **extra** class performance points!!!
- Being shy is OK, but you may lose a good opportunity to gain a high score easily
  - A full usual score is 30 points





# About the Attendance

4

- I NEVER call the roll !! I promise!!
- Feel free to skip the class
  - We are adults, and we have the right to choose;
  - It will NOT affect your final grade;
  - But you will have no chance to gain extra class performance points;
  - And you need to work hard on your assignments;



# Course Information

5

- Syllabus

- Java Grammar (a few classes)
  - ✦ Similar to C++
- OOP (less than one month)
  - ✦ Abstraction | Inheritance | Polymorphism
- Advanced Topics (two months)
  - ✦ The focus
- ~~○ Flipped Classroom (one month)~~
  - ~~✦ Students give presentations on stage~~



# Course Information

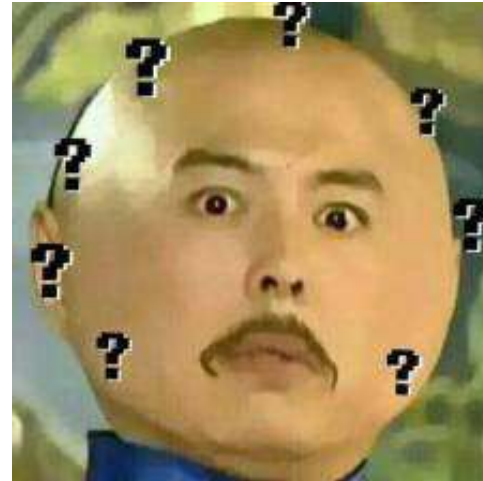
6

- Professor
  - Xiang Zhang 张祥
  - [javaseu@163.com](mailto:javaseu@163.com)
  - Teaching Assistant
- Remember to bring your laptop!!

# First of All

7

- Currently, learning Java (or any language) is easy
  - plenty of textbooks, videos and online materials
  - open source communities like Github and Kaggle
  - QA communities like Quora and StackOverFlow
- So the question is:
  - why we still sit in this classroom to learn Java??





# First of All

8

- For Fun
- For Communication
- For a marathon with the friends







# Content

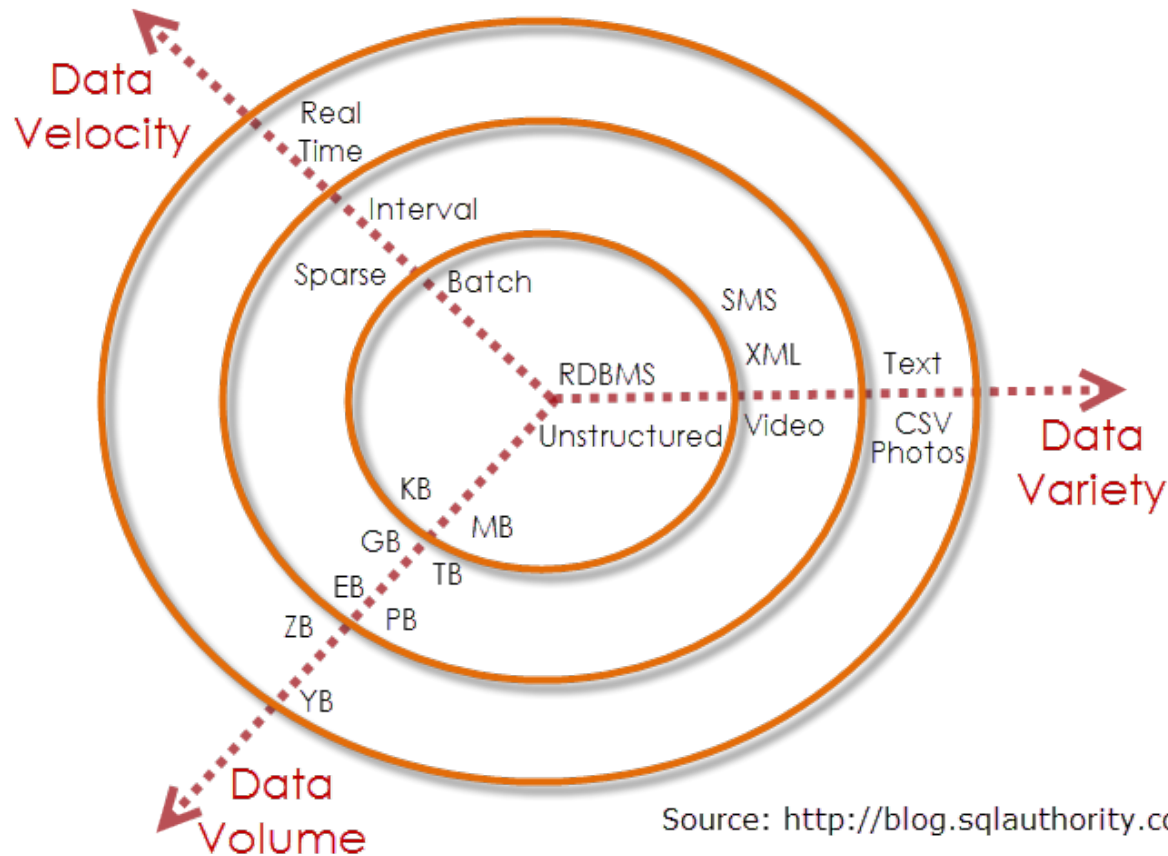
9

- Bloating of Complexity
- Java vs. C++
- About this course
  - Course Arrangement
  - Materials
  - How to Learn
  - Self-study
- Course Design

# Era of Big Data

10

## 3Vs of Big Data





# Challenge: Complexity

11

- Increasing Complexity
  - Complexity of both problems and data
  - Managing the complexity by programming
    - ✦ More abstract models
    - ✦ Easier grammar
- Example:
  - OICQ vs. QQ. vs. WeChat



# Open Discussion

12

How can we estimate  
how many web pages  
are on the WWW?



# Open Discussion

13

- Find some applications that need to make use of extremely large data

## BIG DATA





# Open Discussion

14

- What do you think is hard to understand in C and C++?
- What do you think is not easy to use in C and C++?
- What features do you most look forward to in future programming languages?



# Dilemma

15

- Dilemma in programming

- Increasing user need (Security, Transaction...)
- Increasing size of software modules (from 10s to thousands)
- Decreasing efficiency of software development
- Increasing cost of maintenance (a week coding, but a year maintenance)

Open Discussion:  
what happens to  
an ATM in case  
of power off

- The goal of Java since its birth

- Using precise, easy-to-read, secured program for problem solving
- Thinking in Java – Pure Object-Oriented Thinking



# Evolution of Programming

16

- Evolution of program language

- Machine、 Assembly
- Procedure oriented
- Object -oriented / Aspect-oriented



Increase of  
capsulation

- Evolution of capsulation

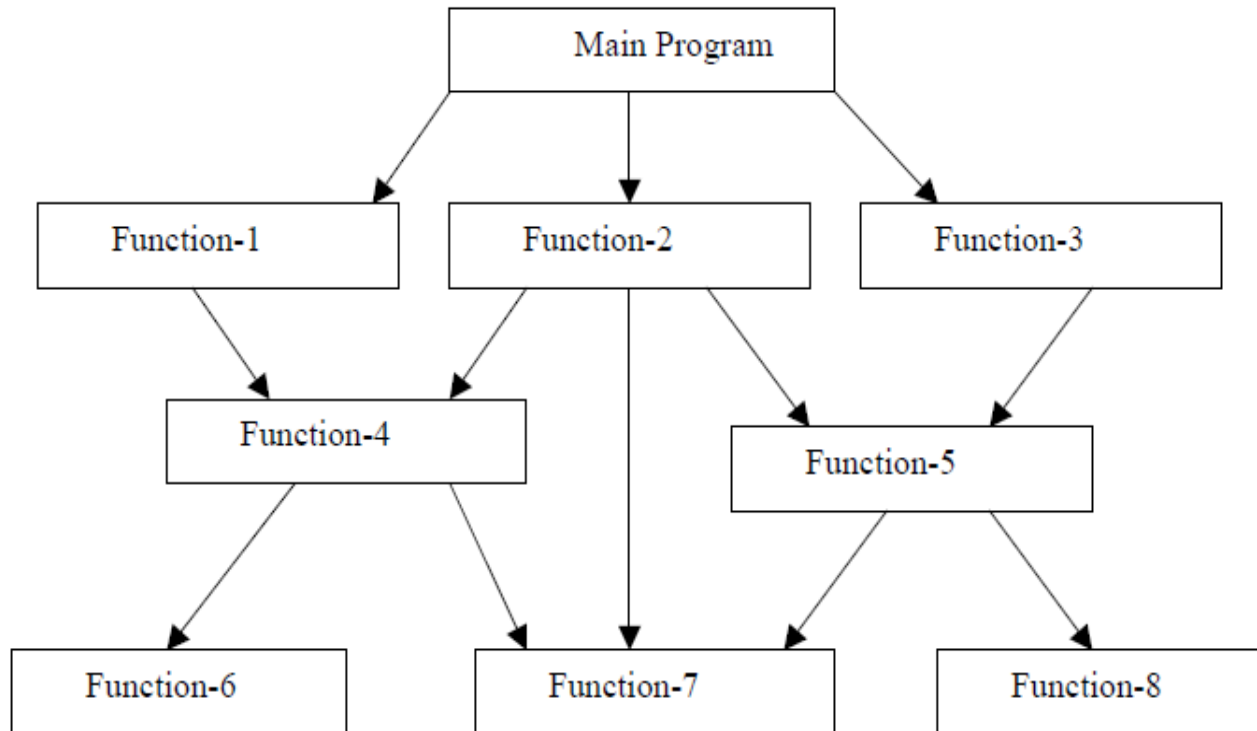
- Class Library
- Design pattern
- Framework
- Service





# POP vs. OOP

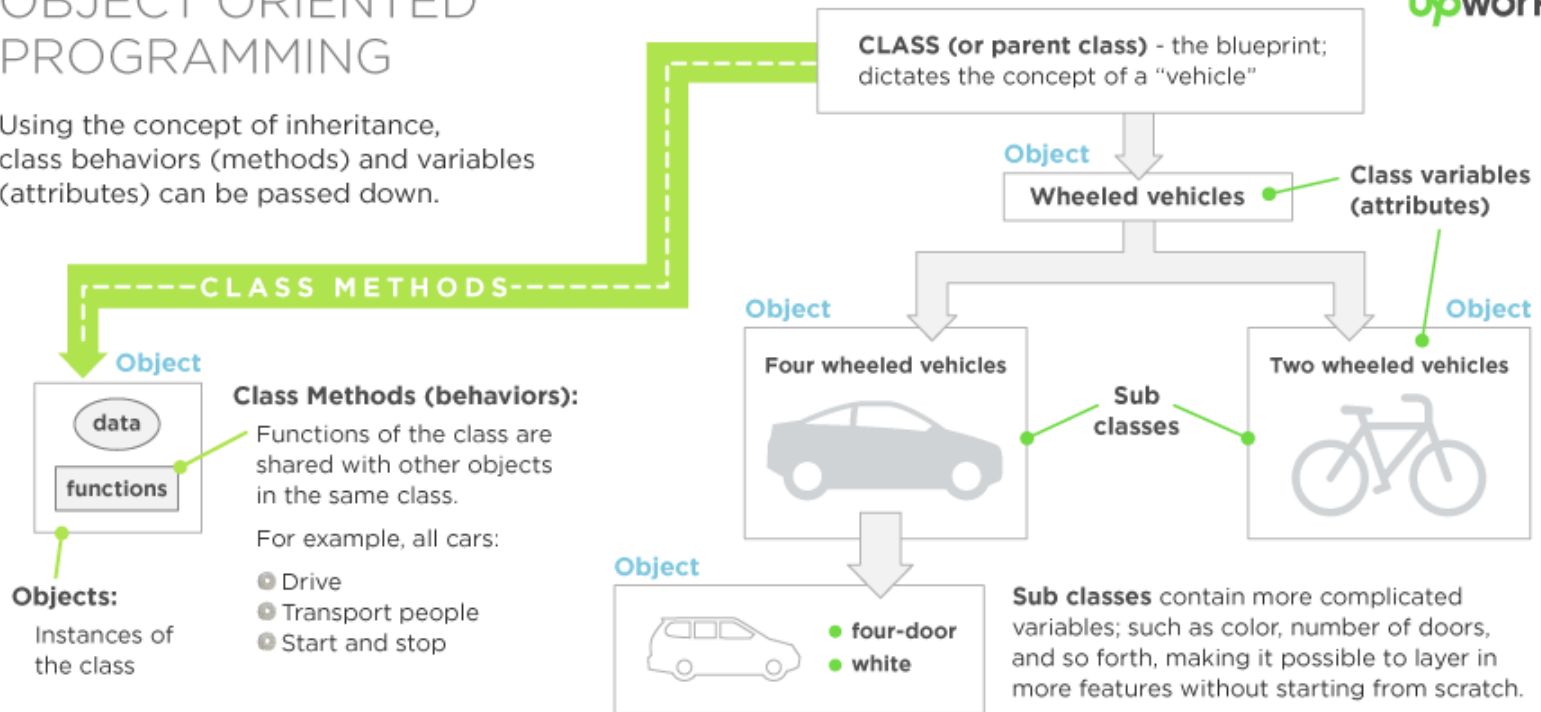
17



## OBJECT ORIENTED PROGRAMMING

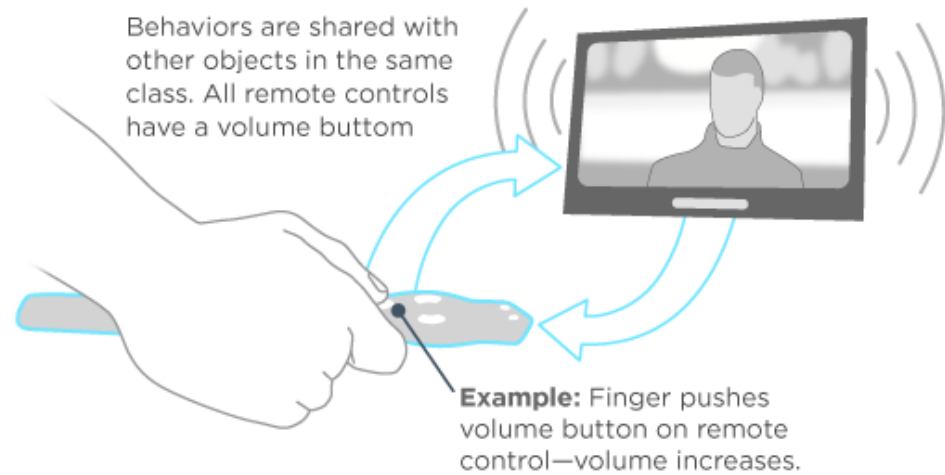
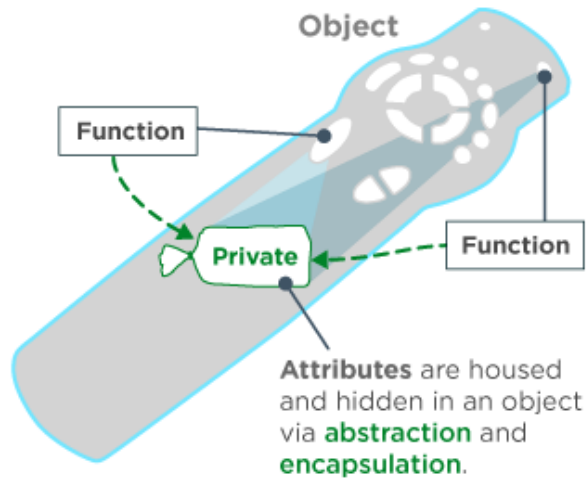
Using the concept of inheritance, class behaviors (methods) and variables (attributes) can be passed down.

upwork™



credit: <https://www.upwork.com/hiring/development/object-oriented-programming/>

## ABSTRACTION AND ENCAPSULATION

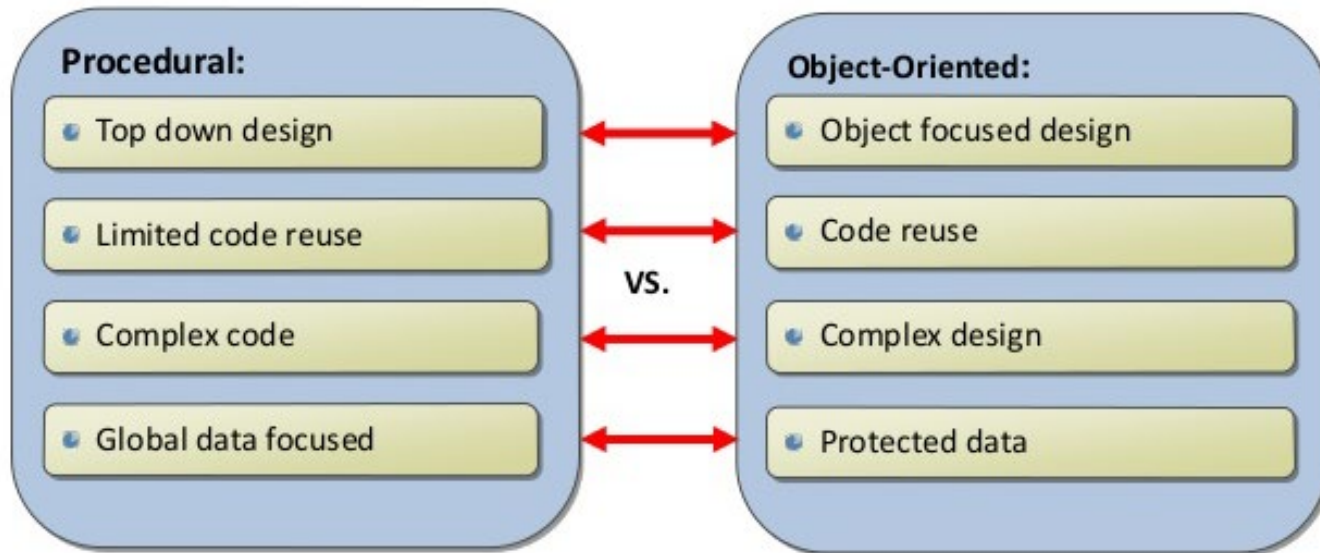


credit: <https://www.upwork.com/hiring/development/object-oriented-programming/>



# Procedural vs. OOP

20



credit: <https://www.slideshare.net/HarisBinZahid/procedural-vs-object-oriented-programming>



# Java and C++

21

- The pointer
- String
- DLL
- Portability
- Multiple inheritance
- Garbage Collection



# Java and C++ - Problems in C++

22

- The pointer

```
// Move and inverse memory
int i;
for(i=0;i<=size;i++){
    to_block[size-i] = from_block[i]; // Can you see the bug?
}
```



# Java and C++ - Problems in C++

23

- String

```
char str [] = "Hello"; //C-style
```

...

```
String str = "Hello, I'm feeling a little better."; //ANSI C++ from 1997
```

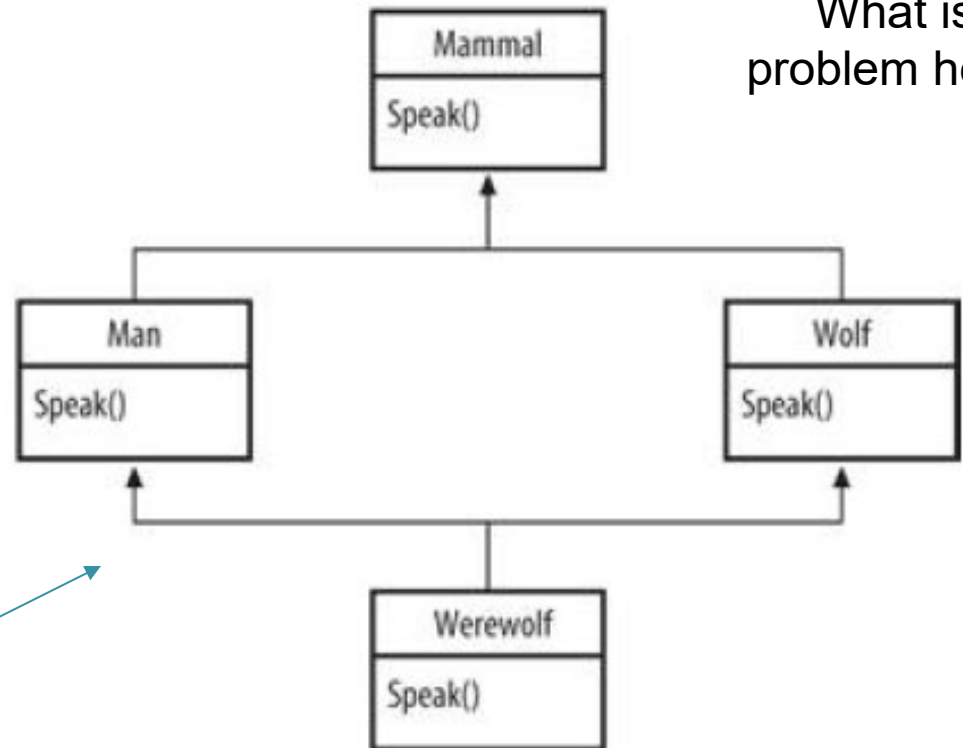


# Java and C++ - Problems in C++

24

- DLL version compatibility
- Portability
- Multiple inheritance

Open Discussion:  
What is the  
problem here?



Diamond Inheritance Problem  
Or the Werewolf Problem





# Java and C++ - Problems in C++

25

- Garbage Collection

`System.gc();`



## Open Discussion:

Is it possible that the JVM can automatically find out all possible garbage? How can it be done?



# Java and C++ - Java Solution

26

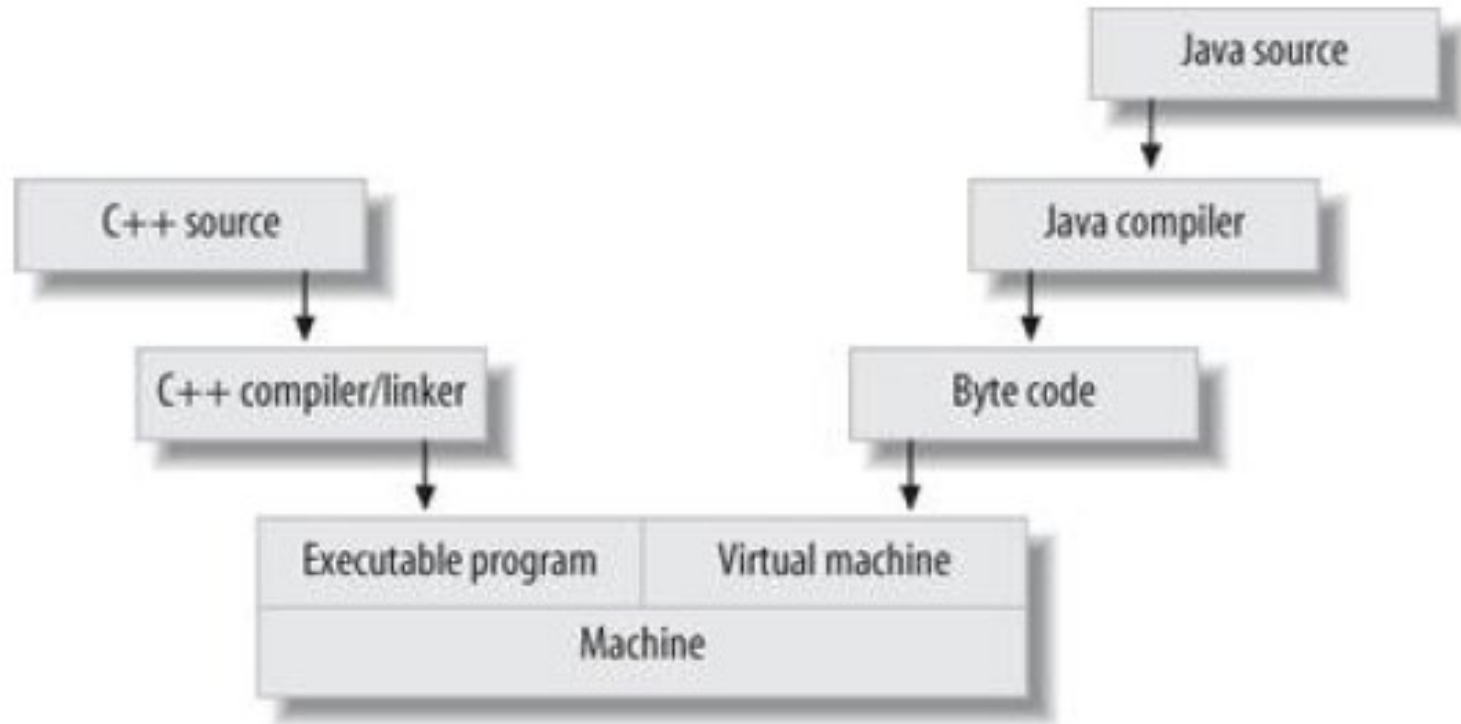
- JVM portability: Write Once, Run Anywhere
- Byte code specification: specified JVM and Java language
- Limited memory access by JVM: sandbox
- Meta-model: Reflective Programming
- Automatic garbage collection
- Pointer removed / Reference is used



# Open Discussion

27

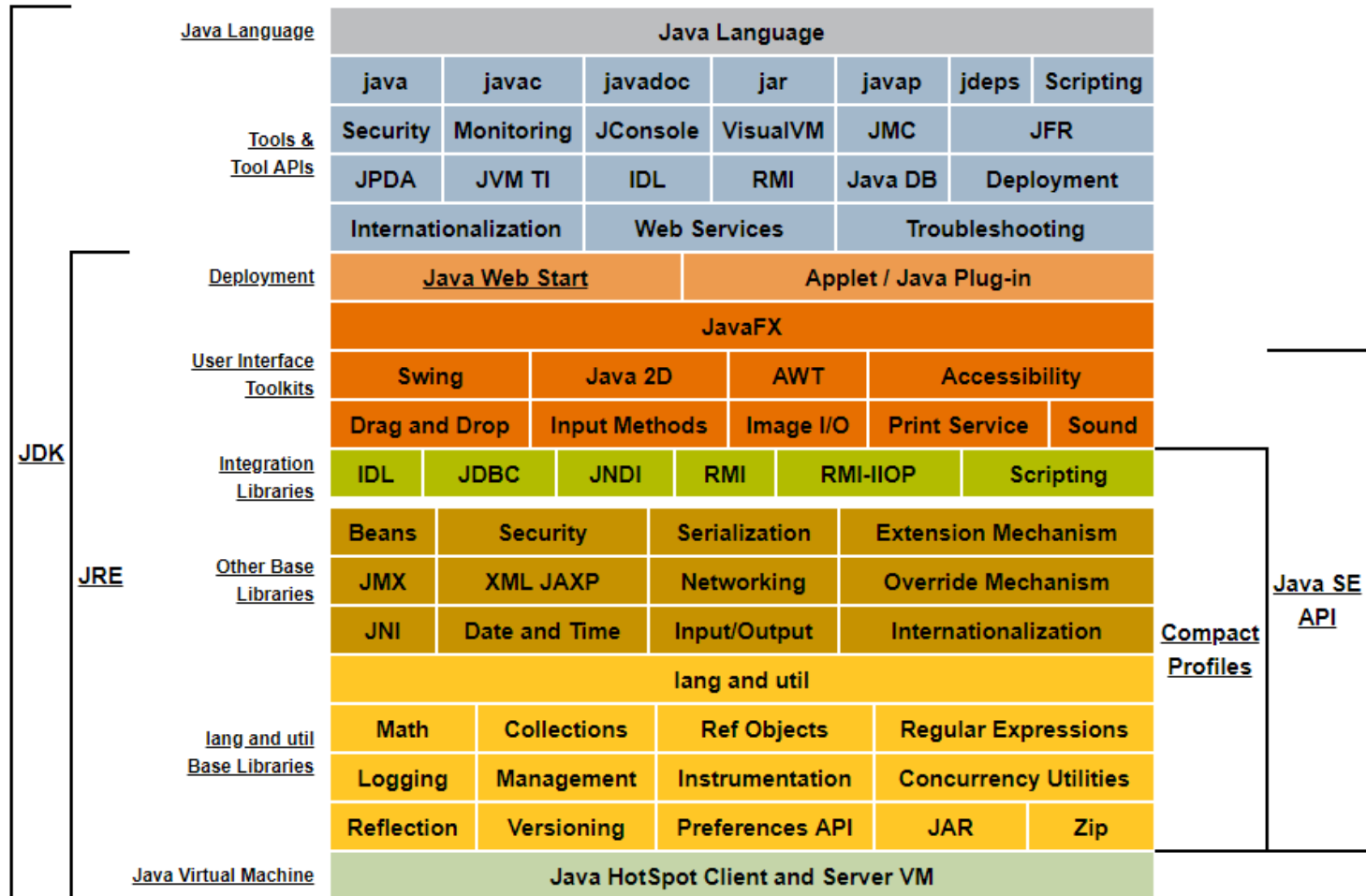
- *What is the pros and cons of Java?*





# JDK | JRE | JVM

28



Week	Content
W1	Introduction, Java Fundamentals
W2	Java OOP I (Classes and Objects)
W3	Java OOP II (Abstract, Inheritance, Polymorphism)
W4	Java OOP II (Abstract, Inheritance, Polymorphism)
W5	Java Exception
W6	Java I/O
W7	Java I/O
W8	Java I/O
W9	Java Collection
W10	Mid-term Lab Work
W11	Java Multithread
W12	Java Multithread
W13	Java Multithread
W14	JDBC
W15	Java Network Programming
W16	Course Project

homepage: <http://wds.ac.cn/java/>

## Java Programming

711191

Xiang Zhang

javacose@qq.com

### Preparing the Environment

---

- Installing JDK 8 (J2SE version 8) into your computer: <https://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>;
- Verifying if JDK 8 has been successfully installed by typing "java -version" in your command line like this. If you see "java version "1.8.xxxx"", congratulations!

```
C:\Users\Administrator>java -version
java version "1.8.0_221"
Java(TM) SE Runtime Environment (build 1.8.0_221-b11)
Java HotSpot(TM) 64-Bit Server VM (build 25.221-b11, mixed mode)
```

- Then install Eclipse IDE for J2SE (<https://www.eclipse.org/>) (the latest version is 2019-06, "Eclipse IDE for Java Developers", please DO NOT download the "Eclipse IDE for Enterprise Java Developers", that is for J2EE, which is another edition of Java) OR IntelliJ IDEA (<https://www.jetbrains.com/idea/>).
- Verifying if the IDE is correctly installed by running a HelloWorld program, taking Eclipse as the example: [example](#)

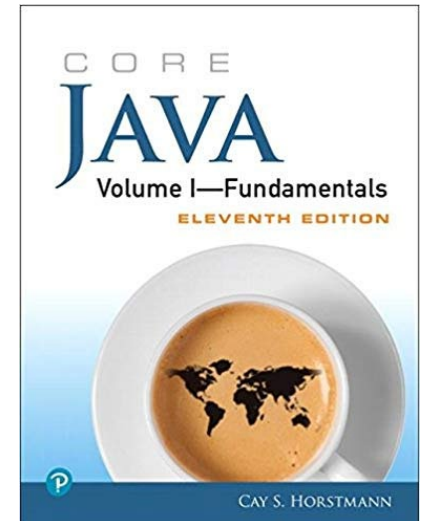
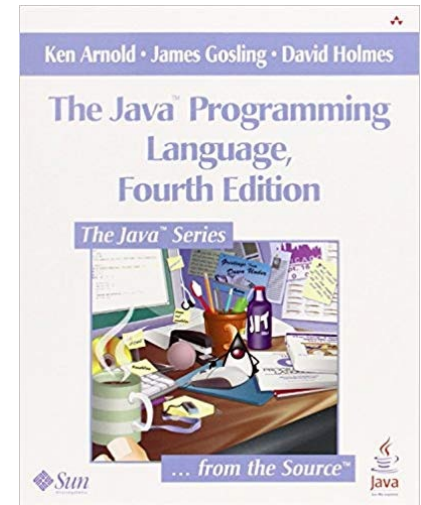


# Recommended Textbooks

31

- The best textbook is Javadoc
- The best professor is JDK/JRE
- Recommended Textbooks:

Title	Edition	ISBN
The Java Programming Language	4th	9780321349804
Thinking in Java	4 <sup>th</sup> /5 <sup>th</sup>	9787111212508
Core Java	11th	0135166306
Head First Java	2nd	0596009208
Inside Java Virtual Machine	2nd	0071350934





# How to Learn

32

- How to study Java in a AGILE way?
  - TYPE the examples in the textbook
  - Learn to use Javadoc
  - Don' t focus too much on the grammar details
  - Resources:
    - ✦ Github, Kaggle...





# Lab work

33

- The installation of JDK/JRE
- Setup the PATH / Classpath
- Install Eclipse / IntelliJ ...