Chapter 0   Introduction

Xiang Zhang

javaseu@163.com

http://wds.ac.cn/java/

- 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;

- I will ask you a lot of questions in this class;
  - o 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
  - o A full usual score is 30 points

- I NEVER call the roll !! I promise!!

- Feel free to skip the class

  o We are adults, and we have the right to choose;

  o It will NOT affect your final grade;

  o But you will have no chance to gain extra class performance points;

  o And you need to work hard on your assignments;

- 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~~

- Professor
  - Xiang Zhang 张祥
  - javaseu@163.com
- Remember to bring your laptop!!

First of All

- 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??
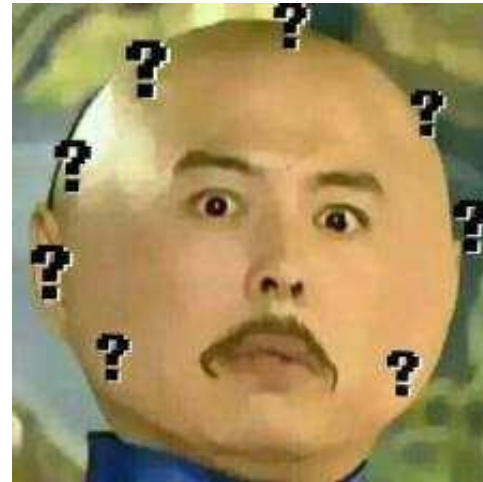
for fun

- For Fun

- For Communication
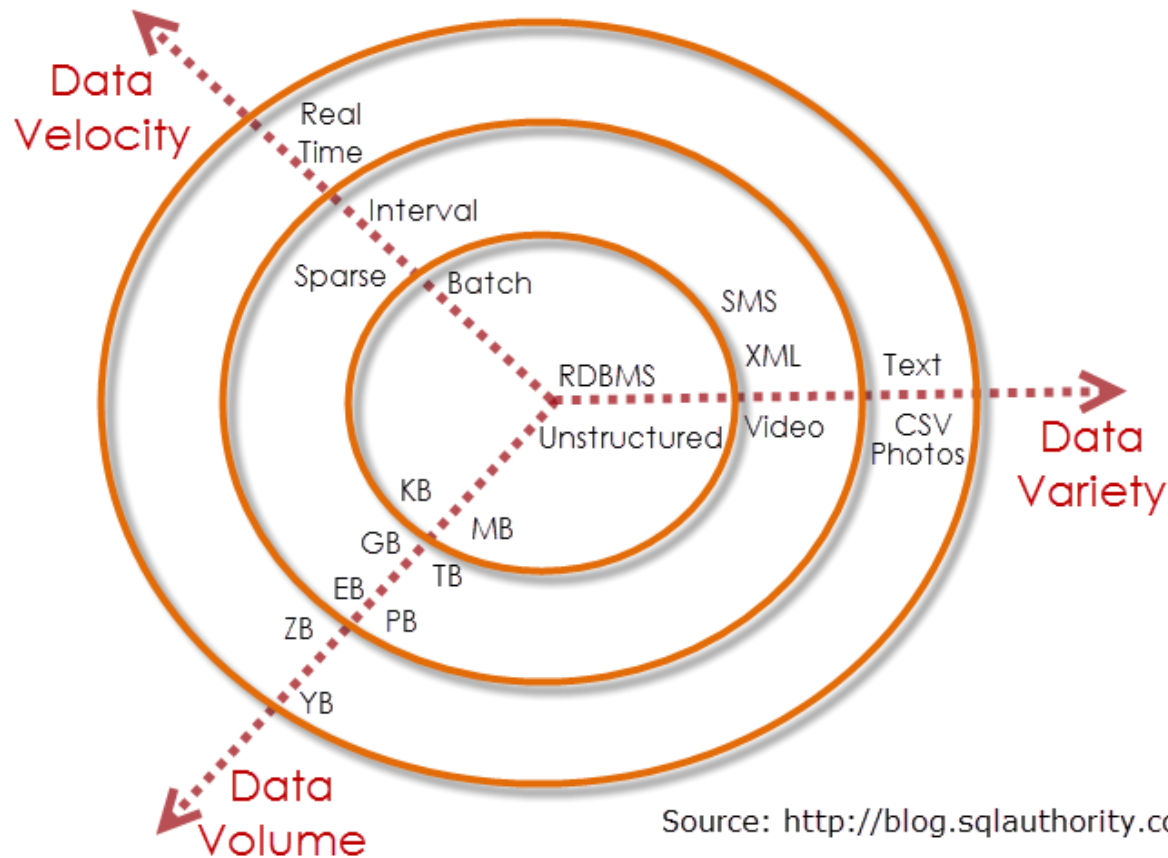
- For a marathon with the friends

Content

- Bloating of Complexity

- Java vs. C++

- About this course
  - Course Arrangement
  - Materials
  - How to Learn
  - Self-study

- Course Design

3Vs of Big Data

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

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

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

- 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

- 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)

- The goal of Java since its birth
  - Using precise, easy-to-read, secured program for problem solving
  - Thinking in Java – Pure Object-Oriented Thinking

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

Evolution of Programming

- Evolution of program language
  - Machine、Assembly
  - Procedure oriented
  - Object -oriented / Aspect-oriented

Increase of capsulation
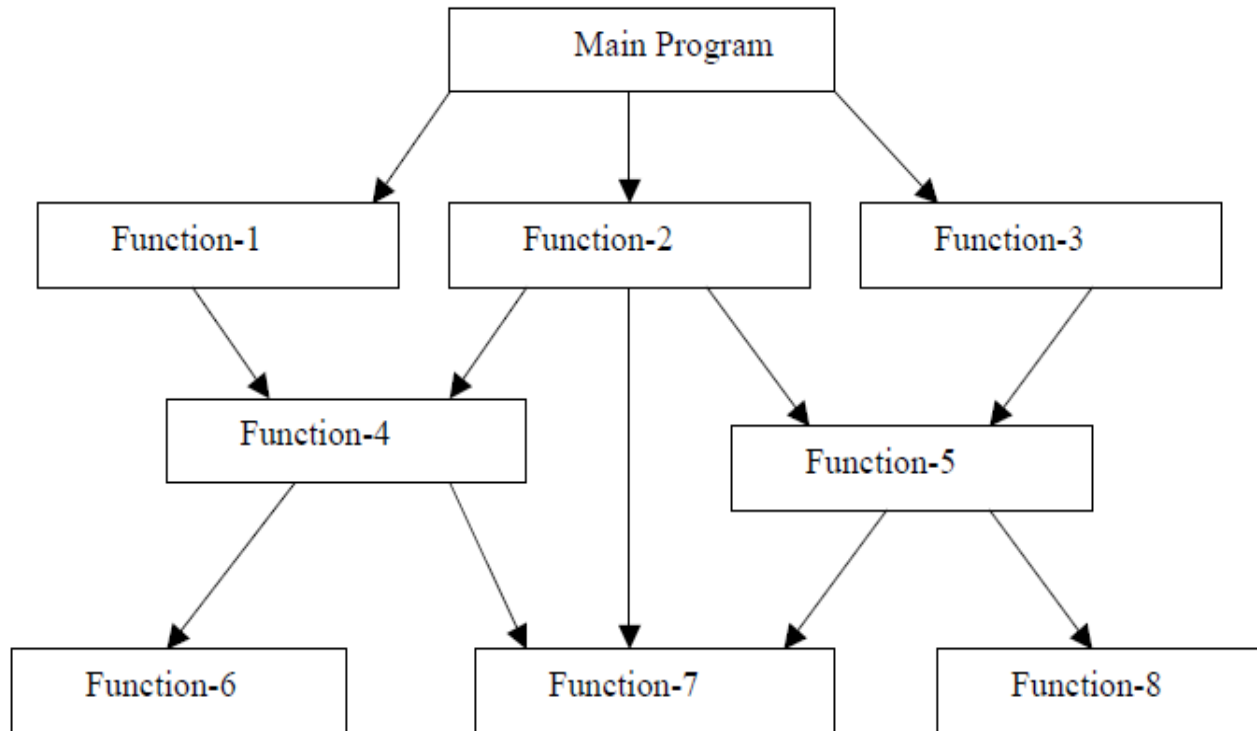
- Evolution of capsulation
  - Class Library
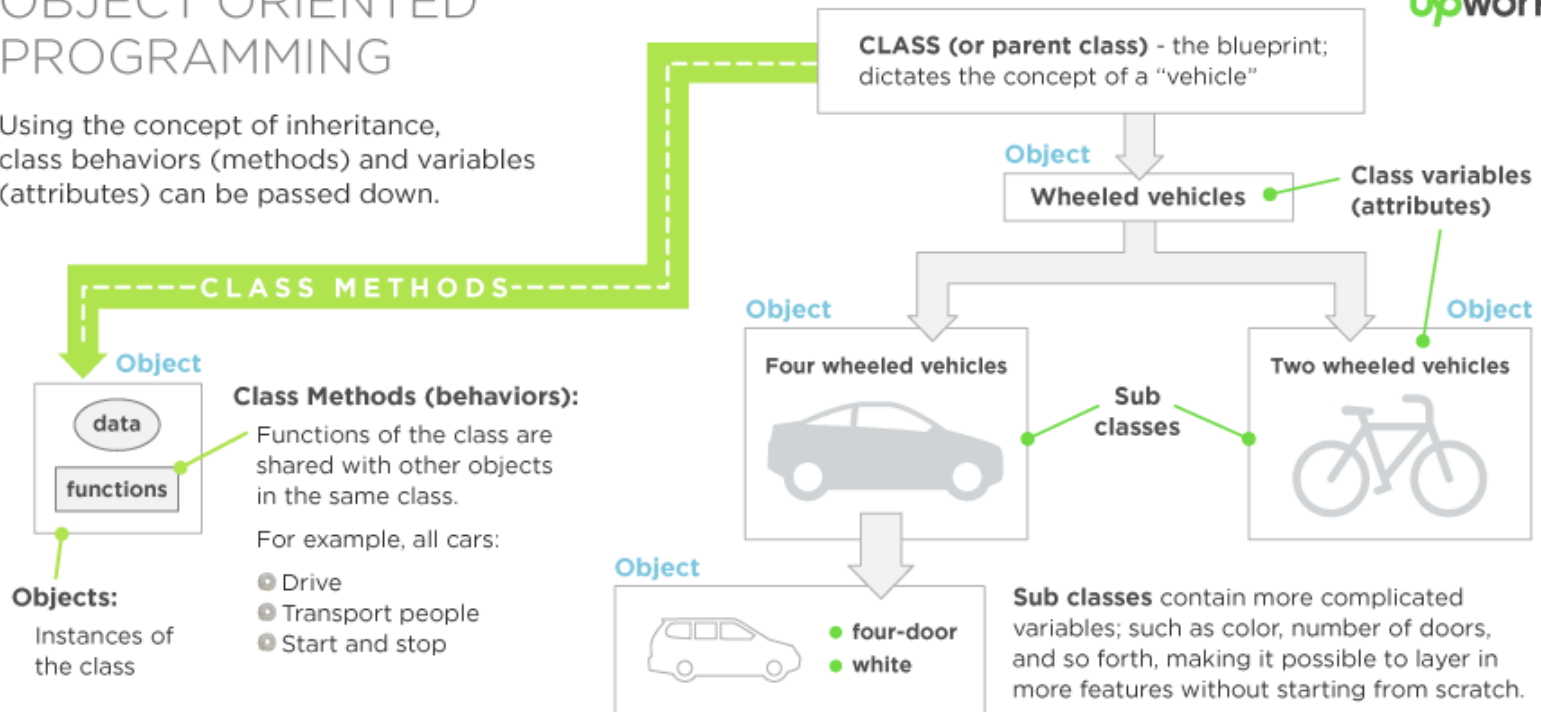 - Design pattern
  - Framework
  - Service

OBJECT ORIENTED PROGRAMMING

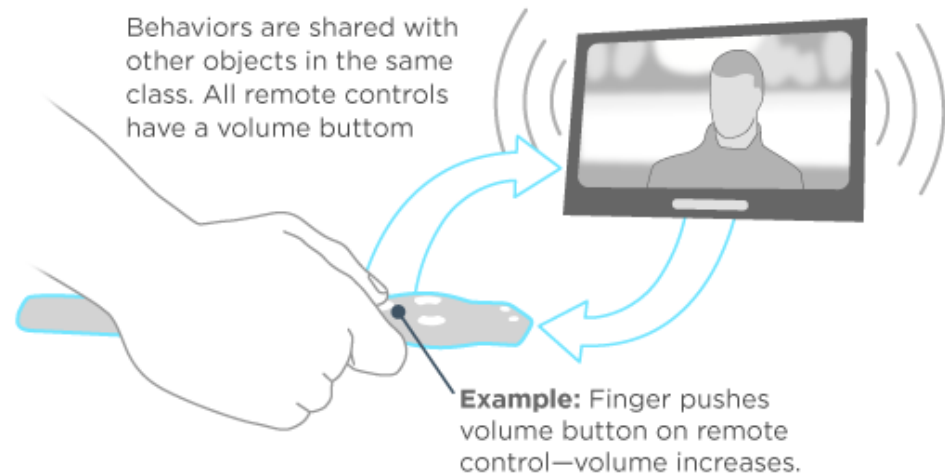**Upwork**

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

CLASS (or parent class) - the blueprint; dictates the concept of a "vehicle"

Object
Wheeled vehicles — **Class variables (attributes)**

- - - - - CLASS METHODS - - - - -

Object
data
functions

Class Methods (behaviors):
Functions of the class are shared with other objects in the same class.

For example, all cars:
- Drive
- Transport people
- Start and stop

Objects:
Instances of the class

Object
Four wheeled vehicles

Sub classes

Object
Two wheeled vehicles

Object
- four-door
- white

Sub classes contain more complicated variables; such as color, number of doors, and so forth, making it possible to layer in more features without starting from scratch.

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

Procedural:
- Top down design
- Limited code reuse
- Complex code
- Global data focused

VS.

Object-Oriented:
- Object focused design
- Code reuse
- Complex design
- Protected data

credit: https://www.slideshare.net/HarisBinZahid/procedural-vs-object-oriented-programming
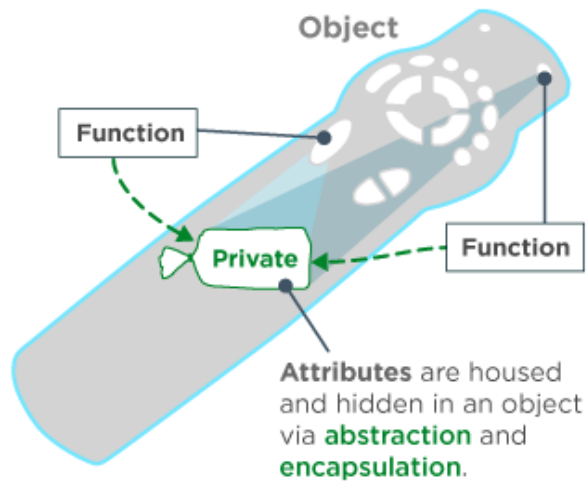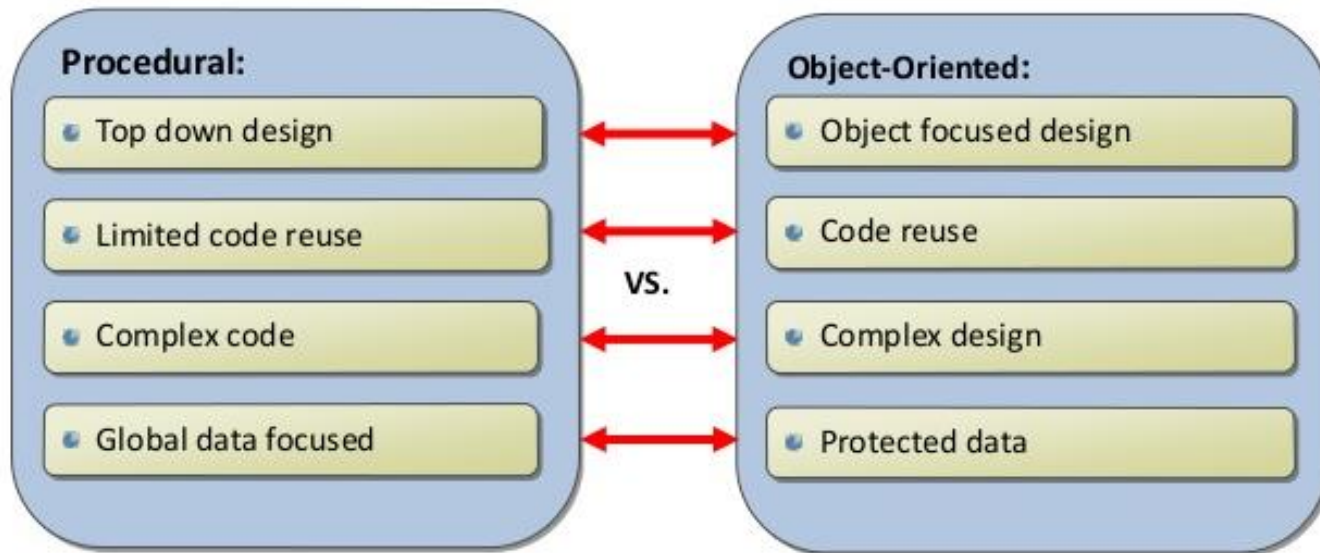
- The pointer

- String

- DLL

- Portability

- Multiple inheritance

- Garbage Collection

- 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?
}
```

- String

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

- DLL version compatibility

- Portability

- Multiple

  inheritance

Open Discussion: What is the problem here?

Mammal
Speak()

Man		Wolf
Speak()		Speak()
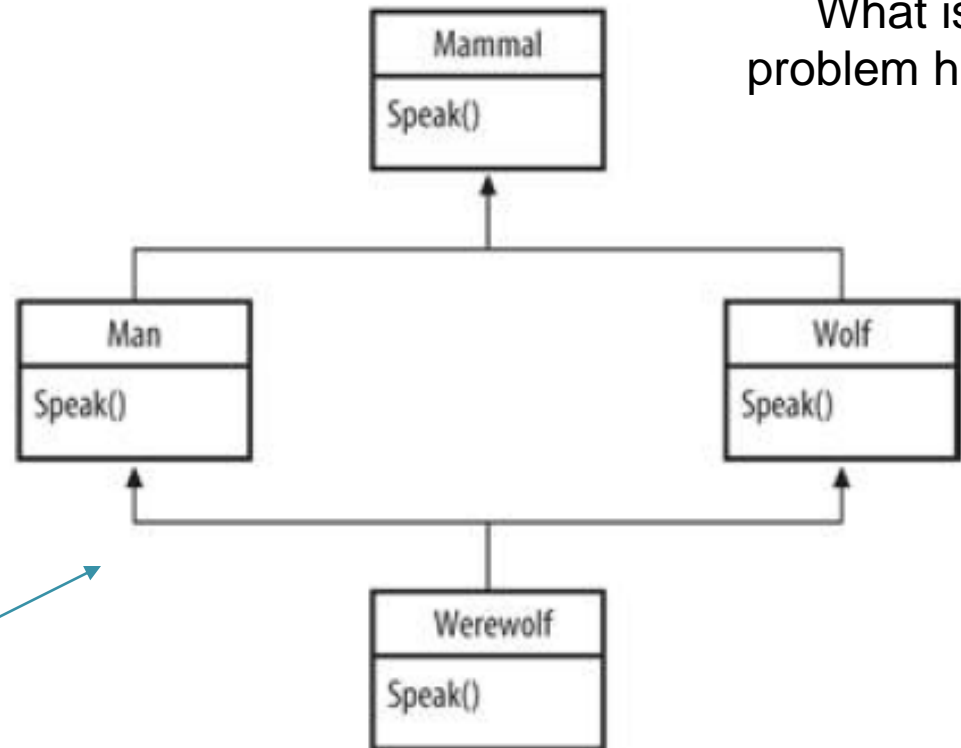
Werewolf
Speak()

Diamond Inheritance Problem
Or the Werewolf Problem

- Garbage Collection

`System.gc();`

delete
finalize

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

- JVM portability: Write Once, Run Anywhere

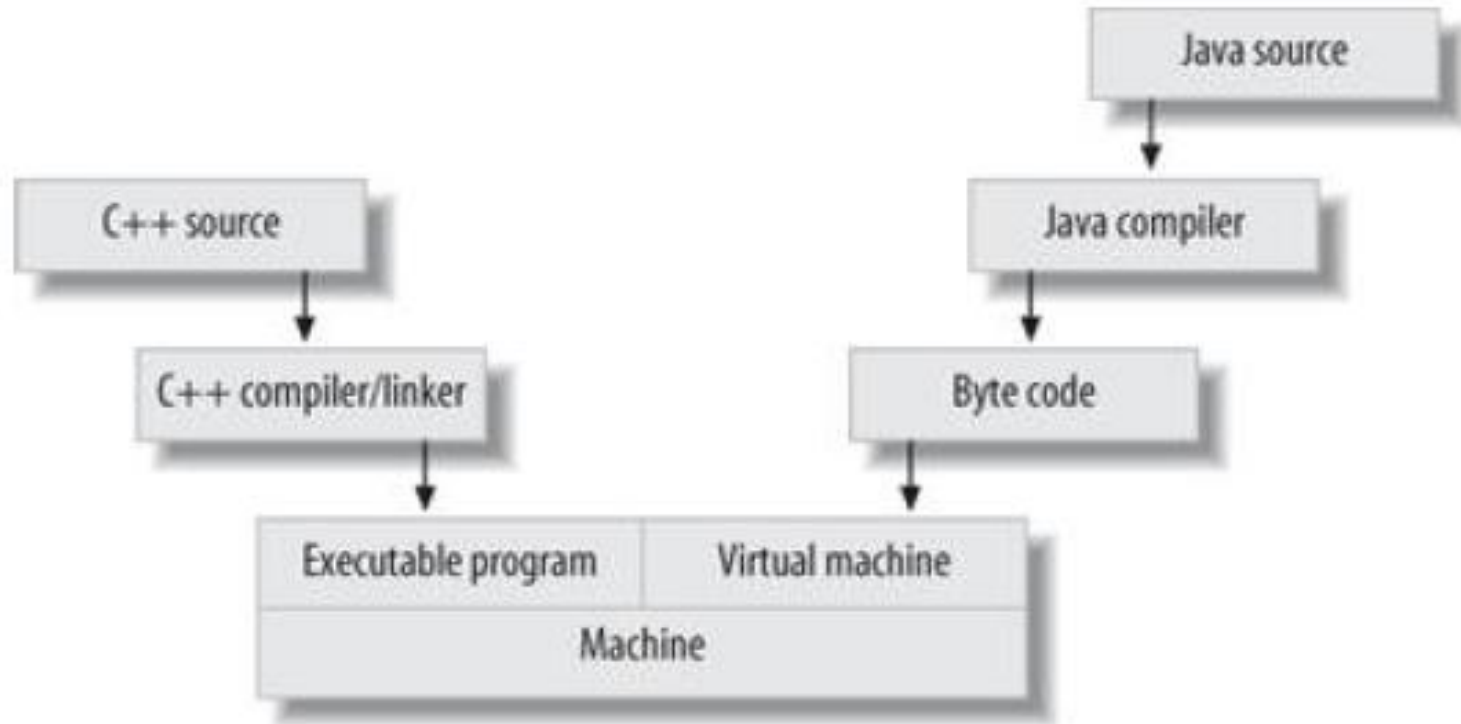- Byte code specification: specified JVM and Java language

- Limited memory access by JVM: sandbox

- Meta-model: Reflective Programming

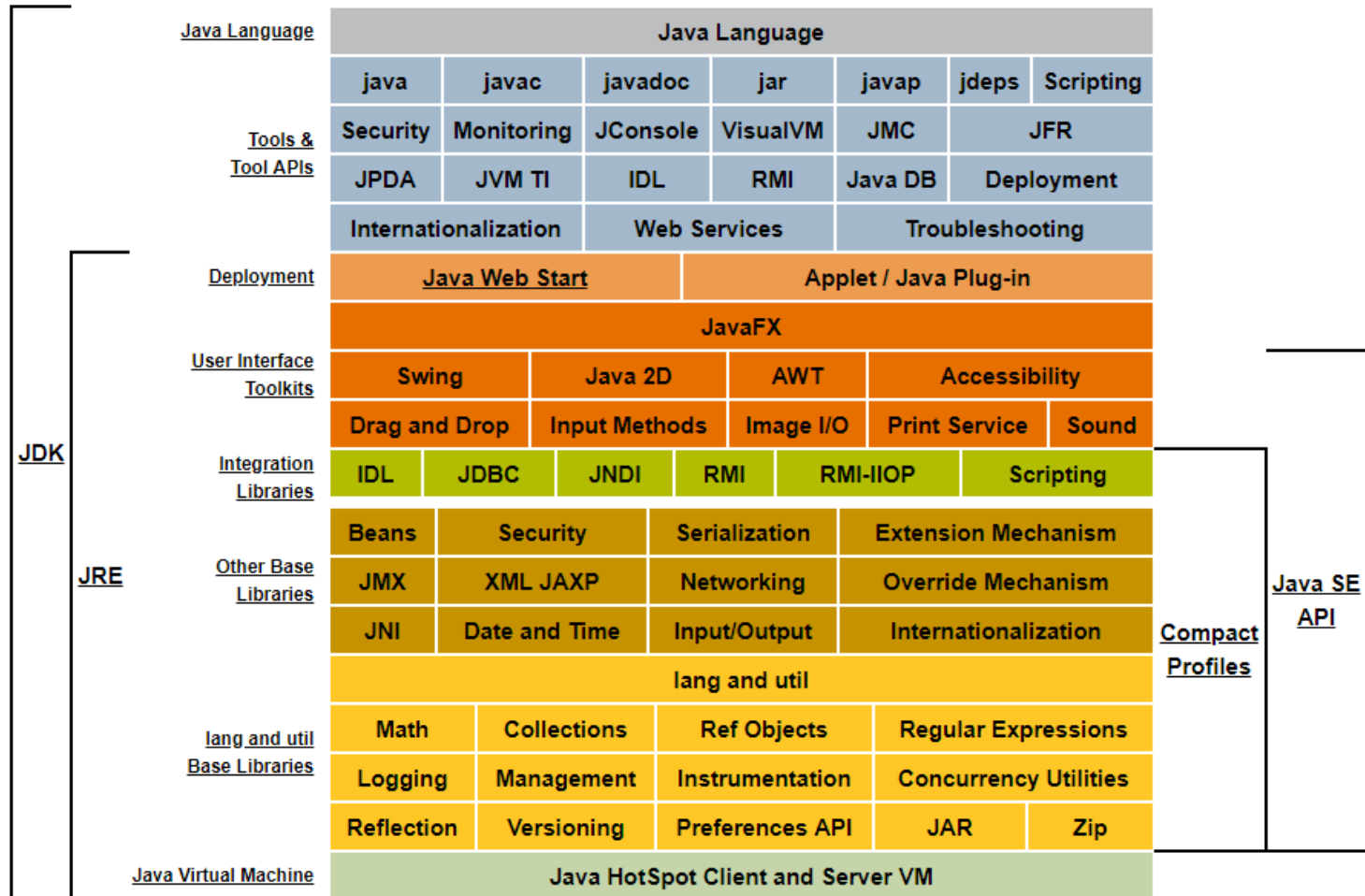- Automatic garbage collection

- Pointer removed / Reference is used

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

Java Language	Java Language						
	java	javac	javadoc	jar	javap	jdeps	Scripting
Tools & Tool APIs	Security	Monitoring	JConsole	VisualVM	JMC	JFR	
	JPDA	JVM TI	IDL	RMI	Java DB	Deployment	
	Internationalization		Web Services		Troubleshooting		
Deployment	Java Web Start			Applet / Java Plug-in			
	JavaFX						
User Interface Toolkits	Swing		Java 2D		AWT	Accessibility	
	Drag and Drop		Input Methods		Image I/O	Print Service	Sound
Integration Libraries	IDL	JDBC		JNDI	RMI	RMI-IIOP	Scripting
	Beans	Security		Serialization		Extension Mechanism	
Other Base Libraries	JMX	XML JAXP		Networking		Override Mechanism	
	JNI	Date and Time		Input/Output		Internationalization	
	lang and util						
lang and util Base Libraries	Math	Collections		Ref Objects		Regular Expressions	
	Logging	Management		Instrumentation		Concurrency Utilities	
	Reflection	Versioning		Preferences API		JAR	Zip
Java Virtual Machine	Java HotSpot Client and Server VM						

JDK

JRE

Compact Profiles

Java SE API

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 succesfully installed by typing "java -version" in your comannd line like this. If you see "java version "1.8.xxxxx"", congradulations!

```
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 installs 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

- 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	4th/5th	9787111212508
Core Java	11th	0135166306
Head First Java	2nd	0596009208
Inside Java Virtual Machine	2nd	0071350934
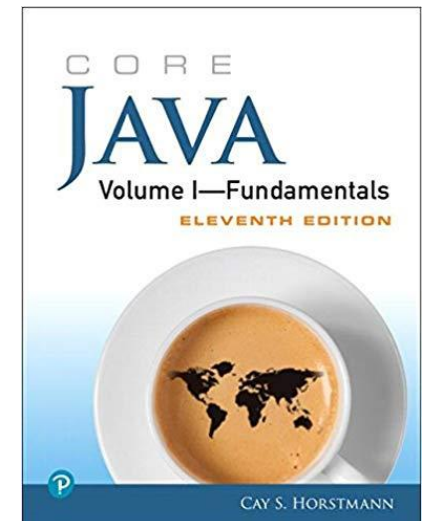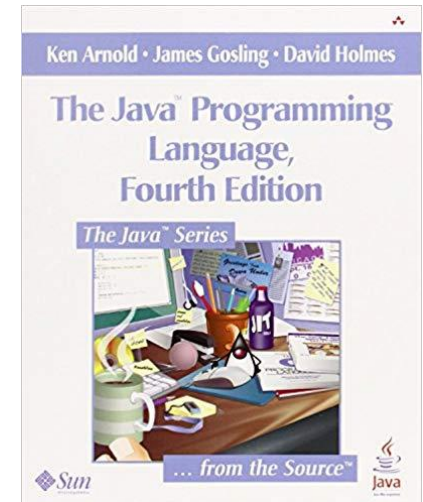
How to Learn

- How to study Java in a AGILE way?

  o TYPE the examples in the textbook

  o Learn to use Javadoc

  o Don't focus too much on the grammar details

  o Resources:

    ⚹ Github, Kaggle…

Lab work

- The installation of JDK/JRE

- Setup the PATH / Classpath

- Install Eclipse / IntelliJ …