

# An in-depth JAVA Teaching Exploration into the Software Engineering Curriculum

Qing-Wei Xu<sup>1</sup>

<sup>1</sup>Computer Science and Technology, HUBEI University of Education Wu Han, China

## Abstract

Currently, the setting of JAVA can not meet the need of enterprises' employment demand. In this paper, we divide the JAVA learning process into four stages: basic knowledge, junior programming, web development, and senior programming. Step by step, we will introduce the students into in-depth understanding, and integrate JAVA and related technologies into software engineering curriculum.

**Keywords:** JAVA, Teaching-Topics, Project-Driven

## 1. Introduction

According to related statistics, at present, the demand for software professionals has reached 20 million in China, while it is still increasing at an annual growth rate of about 20%. In 2009, the supply gap of software talent in our country has reached 425,000, especially lack of JAVA talents. According to IDC statistics, among all needs of software development engineer, the demand for JAVA talent amount to 60% to 70%. Despite the wide gap on JAVA talent, it is difficult for IT companies with vigorous demand to recruit qualified JAVA software engineers. The most fundamental reason lies in that many graduates on computer sciences do not master the practical skills and experience during the academic study, resulting in a huge gap to the actual needs of these enterprises [1].

It is believed that the main reasons on the gap between the actual needs and the JAVA curriculum given by the universities lies in the following:

First, the course construction on JAVA programming is seriously lagging. In the curriculum construction of software engineering, C language programming is given in the second semester, to train the student the basic programming skills by taking the process programming as the main content. In the third semester, objective-oriented programming takes C++ as the tool, aim to add the objective-oriented ideas on the basis of traditional programming techniques. Usually the JAVA language would be given at the fifth semester or later, mainly as a manner of practical technology. Although JAVA and C are both objective-oriented programming languages, there are a

lot of differences in between. It results in poor grasp for the students to give this course so late. And further it is not conducive to develop the professional courses based on JAVA, for example, JSP dynamic web design, JAVAEE enterprise applications.

Secondly, JAVA and C++ programming curriculums have many repetitive contents. Currently, most JAVA teaching materials mostly lay emphasis on its objective-oriented characteristics, which don't reflect the advantages of cross-platform, safety, openness as a mainstream development technology for enterprise applications. The teaching materials on C#, VB and VC also have the same problems. "We study many programming languages, but grasp a little. Once we undertake a project, we can do nothing at all", a typical complains from many students.

Thirdly, JAVA teaching is isolated. Many main courses on computer subject, such as Data Structure, Computer Network, Operating System, Software Project, Database System Concepts, could be explained using related JAVA knowledge. Also, JAVA knowledge would be applied in some application courses, such as Software Test, Security in Computing, Computer Graphics with OpenGL, Embedded Technology, and Multimedia Programming. Unfortunately, only a few courses such as Data Structure are explained with JAVA language. Students just study each course in isolation, and are lack of parallel relationship and integration with other knowledge. They have to face the large system, various new technology when start the JAVA courses with comprehensive requirements such as JavaEE enterprise application. Without a very solid foundation, they will feel confused and lost interest in learning JAVA.

## 2. Resolution

As shown in Figure 1, taking the conventional software engineering courses as an example, the study process of JAVA can be divided into four stages, i.e., JAVA Basics, the junior programming, Web programming and Senior Programming [2]. In each stage, all kinds of related JAVA technologies are inserted into the learning of professional courses. There are close ties between the different courses in the same stage [3].

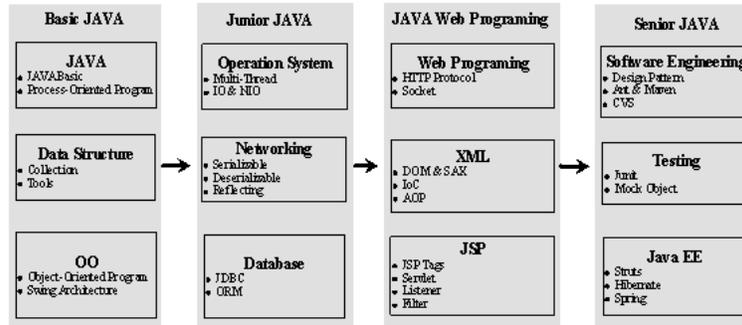


Figure 1. Curriculum System Construction based on JAVA

JAVA basic knowledge takes the traditional objective-oriented technology as the main content. Students will study in depth the following two aspects. Firstly, by using the source code of the classic java.lang and java.util core software packages, the course explains how to achieve such as linear tables, linked lists, stacks, queues, binary trees and other common data structures, and related finding and sorting algorithms. Secondly, it takes structure design of the java.util and javax.swing software packages as an example enable students to understand in-depth of the objective-oriented programming ideas.

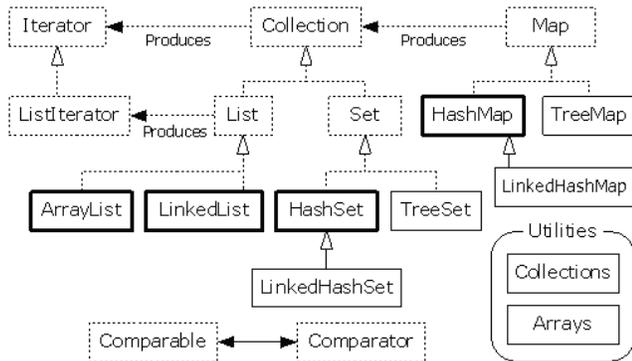


Figure 2. Curriculum System Construction based on JAVA

Programming in JAVA primary stage, the author advocates the spirit of "to learn to use". After the completion of the JAVA basics, the instructor should intersperse JAVA related realization technology into professional fundamentals, to enable students to integrate theory with practice while to avoid boring isolated and abstract concept. Take the operating system program as an example, the process management is the key point and difficulty. In the learning process students often just stay at the theoretical level. Correspondingly, JAVA thread management introduces in the life cycle of threads, synchronization mechanisms, deadlocks, queue congestion and other related technologies. Students can experience in-depth synchronization and locking mechanisms in the process of programming of multi-threaded JAVA applications. Further, JAVA SDK provides a set of thread-safe and thread-unsafe containers (realized by data

Figure 2 shows the inherit relations between part of the categories in java.util package, from which the students can clearly understand that as a linear form, List (interface) has two specific implementations, i.e., ArrayList and LinkList. And further collections tools can provide binary search and sorting services for all Collection. The students will understand not only the concept of inheritance and reference in objective-oriented ideas, but also the intrinsic relationship between different data structures, so as to understand the organic links between these two professional foundation courses.

When analyzing the source code students can link together the operating systems and data structures courses, to understand deeply how to solve thread safety issues for multi-threads. The same method can also be used in the study of the two courses of computer networks and database systems.

The third stage takes the B/S three-tier architecture as the core, to explain how to use JAVA technology to develop simple dynamic Web applications. For WEB programming curriculum, there are two options in colleges and universities, i.e., ASP.NET provided by Microsoft and JSP by Oracle Corporation (PHP Course less opens). Microsoft Visual Studio integrated with development environment is quick and easy to use, and is more vulnerable to the students. In contrast, to do the same thing, the code for JSP is more than twice long than ASP. JSP also involves a number of knowledge points and complex configuration, which result in poor effects for such a university courses. To solve this problem, in the learning of WEB program design and XML related courses, it is also essential to add the content of JAVA related technologies. Otherwise, by the JSP course alone (72 hours for JSP, and 32 hours for curriculum design in the author's school), time will certainly be stretched to complete JSP course better, including the teaching of all knowledge points while guiding the students to develop a complete project.

For the JAVA senior programming, i.e. the fourth stage of the learning process, the authors advocate the so-called project-driven teaching methods[4, 5], which means breaking the previous passive learning mode as "teacher lecturing and students learning". The instructor should

comprehensive train the students independent problem-solving skills and team spirit. Mostly, program developed by using JAVA usually aim to enterprise application systems. The long development cycle and the large number of personnel all make the traditional teaching model unworkable. Take the Project-oriented method, the teacher would guide students to participate in the whole process from the topics selection, needs analysis, module design, coding, functional testing, and final deployment. Students will integrate all previously acquired knowledge and skills, experience the real development process and accumulate experiences.

### 3. Conclusions

The author argues that a teaching system of JAVA should be set up through a set of JAVA related technologies. In JAVA basics and Primary Programming stages, the study will focus on linking the knowledge points scattered in different computer science courses. Students can reference crossly and improve scrolling throughout the whole learning process. As an old saying says: "knowledge coming from the paper is always superficial, one needs practical practice to understand completely a thing". In the last two stages of the curriculum, teacher should introduce the project-driven method to improve student's abilities of analyzing and solving problems.

While emphasizing JAVA technology, the author does not ignore the importance of C++ and .NET courses in the teaching of the computer science. In contrast, it is also a good choice to apply C++, .NET and the related technologies to the software engineering curriculum construction. Any kind of computer programming languages is just a tool for software development, the focus of teaching should not set out the types of tools, but should lay emphasis on method and skills of analyzing and solving problems.

### References

- [1] Wu-Jing and Ren-Liyong, Thinking in Education of University Software Engineer Education. *Computer Education*, 2007(14): p. 67-70.
- [2] Cheng-Lei and Yao-Nansheng, Investigation of Java Courses Setting. *Journal of HuaiNan Teachers College*, 2009(003): p. 45-48.
- [3] Li-Ming, The Curve of Java Learning. *Computer Education*, 2009(24).
- [4] Luo-Ling, Project-Driven Curriculum Innovation in "ASP NET Programing". *Computer Education*, 2009(13).
- [5] Wang-Yuan, Wen-Lihui, and Zhi-Tingsheng, The Research in Project-Driven Based "JavaEE framework" Curriculum. *Software Engineer*, 2010(007): p. 50-52.