DOI: 10.23977/aetp.2024.080629
ISSN 2371-9400 Vol. 8 Num. 6

Exploration on the Teaching Model of Java Programming and Practice for Students with No Programming Background

Qi Wang[a,*], Lulu Zheng[b], Ruijin Hong[c]

School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, 516 Jungong Rd, Shanghai, 200093, China
[a]wangqi580@usst.edu.cn, [b]llzheng@usst.edu.cn, [c]rjhong@usst.edu.cn
[*]Corresponding author

Keywords: Java; Program design; Teaching mode

Abstract: Java Programming and Practice is a compulsory course for students majoring in Computer Science and Technology and an important elective course for those of other majors. Based on the author's own teaching practice, along with a meticulous study of the teaching content and curriculum system, and with a clear positioning of the course goals, taking both social learners and college learners into account, this paper explores how to implement an efficient and effective teaching model and method for the Java programming and practical courses for undergraduate students with no programming foundation. To enable students to better understand the concept of object-oriented programming, the teaching significance, teaching content, teaching methods, and student practice forms of the course teaching are discussed, thereby expanding students' thinking space in software design, enhancing their learning effect and interest, meeting the requirements of personalized teaching, and strengthening students' software development capabilities.

1. Introduction

With the increasing demand for software talents and the necessity to enhance the technical level of students specializing in software-related fields, the demand for talents in society is not only stricter but also more diverse. To cultivate and foster a large number of high-quality technical application-oriented talents with strong innovation ability and adaptability to economic and social development needs, it is essential for schools to train talents based on the requirements of society, adjust the structure of personnel training, improve the quality of personnel training, and enhance the employ ability of graduates. Java programming technology is an especially significant part of computer software design and development. To better apply Java programming technology, it is of great significance and necessity to study the characteristics of Java programming in computer software and its technical application [1-2]. Java programming courses not only focus on the fundamentals but also emphasize students' practical skills during classes.

Through the study of Java Programming Design and Practice, students can not only understand the programming environment, establish project awareness, master operational and practical skills, stimulate the spirit of innovation, improve comprehensive quality, and cultivate the sense of innovation and a rigorous scientific work style. It also lays a preliminary practical foundation for students to learn other related courses and engage in programming design and software development in the future.

2. Java Programming Teaching and Learning Content

JAVA is a software engineering programming platform and is also regarded as a computer software programming language. It possesses the advantages of cross-platform operation and dynamic development and enjoys broad application prospects in the field of computer software development. The JAVA programming language can be considered as a modified version of C++, featuring many similarities to C++, such as being an object-oriented programming language. Simultaneously, it effectively resolves numerous problems and errors in the C++ language and is not prone to operator overloading and pointer ambiguity, which effectively enhances the quality of software development. According to the course content, this course is divided into three modules, namely Java syntax and programming basics, Java object-oriented programming, and Java advanced features. Among them, the Java syntax and programming basic module mainly consists of four parts: Java overview, language basics, strings and vectors, and arrays; the Java object-oriented programming module mainly includes three parts: classes and objects, inheritance and poly-morphism, and exception handling. The Java advanced feature modules mainly comprise five parts: graphical user interface, "file, stream and I/O technology", multimedia and multi-threading, Java network program design, and JDBC access database, forming a complete knowledge system [3-4]. Computer program design is an indispensable course in the practical teaching for undergraduate students of optoelectronic engineering. The aim is to assist students in acquiring basic program design skills through programming and debugging, foster the ability to integrate theory with practice, enhance the ability to analyze and solve problems, lay a foundation for the subsequent study of computer courses, and strengthen the ability of independent thinking and teamwork. Java programming is a highly practical teaching course, with theory serving as the basis of practice. The nature of this course necessitates the combination of classroom teaching and practical teaching in the teaching process to cultivate students' comprehensive application ability of analyzing, designing, coding and testing Java products. However, this leads to some students being negative and fearing difficulties in the learning process, lacking learning initiative, and resulting in poor teaching effects for teachers. It also affects the study of subsequent professional courses. Firstly, for the zero-basis Java programming and practical teaching content design: Based on the learning objectives and difficulty levels, develop a clear syllabus and course content, and integrate with practical computer courses to provide a comprehensive learning experience. In practical teaching, teachers typically adopt the teaching method that combines theory with practice to deepen students' understanding of knowledge points and provide timely feedback when students have questions, thereby increasing the interaction between students and teachers. Secondly, it is necessary to assess students' learning outcomes by using online tests, programming assignments, project practice and other means to evaluate learning results and adjust teaching strategies in a timely manner. Thirdly, it is necessary to establish a student support and feedback mechanism: Set up a student support system, provide channels for learning guidance and doubt-solving, collect feedback from students regularly, and constantly optimize teaching methods.

3. The challenges and strategies of practical methods for Java programming

We need to cultivate students capable of addressing practical problems. Therefore, we should not only enhance students' learning ability but also improve their comprehensive ability to analyze and solve problems. We should take students as the center and application as the orientation to better promote the combination of theory and practice.

The traditional experimental teaching mode has the following three disadvantages: (1) Offline classroom teaching has limitations in time and space, which are mainly reflected in the insufficiency of experimental class hours, restricting the development of students' ability to solve practical problems by applying theoretical knowledge. (2) Confirmatory experiments are relatively numerous, while comprehensive design, innovative research, and open experiments are scarce, thus failing to meet the differentiated needs of students at different levels and reducing the interest of non-computer majors in learning programming. (3) The single process evaluation method and the lack of an effective process management mechanism are not conducive to the collection, sorting, and analysis of students' learning data, making it impossible to comprehensively and timely grasp students' learning situation [5-6].

Java programming is a course with strong practicality in teaching. Through practical teaching, students' innovative ability can be cultivated, their learning interest can be stimulated, and the class will no longer be dull and boring. The number of student absences will decrease, and the interaction between students and teachers will increase. In teaching practice, emphasis is placed on explaining the operation points where students are prone to make mistakes. For example: (i) Inheritance, construction methods in inheritance; we need to explain the difference between multiple inheritance and multiple inheritance in detail. (ii) Abstract classes. (iii) The function of polymorphism and method overloading. (iv) Exception handling mechanism. In the process of software development, exception handling is an inevitable part of the program. Exception handling is a process of automatically capturing and handling unexpected results or situations when a program encounters them at runtime. The exception handling mechanism in the Java course is to catch the exception at runtime and call the appropriate method to handle the exception. The types of exceptions can be classified into the following types: (1) Capture and handling of runtime exceptions. When a program encounters runtime exceptions (such as Out Of Memory Error) during runtime, the exception handling mechanism in Java courses will capture the exceptions during runtime and call the corresponding methods to handle the exceptions. (2) Capture and processing of compile-time exceptions. When the program encounters compile-time exceptions during compilation (such as: null pointer references, array out of bounds, etc.), the exception handling mechanism in the Java course will capture the exception at compile time and call the corresponding method to handle the exception. (3) Capture and processing of warning exceptions. When the program encounters warning exceptions at compile time or runtime (such as undefined variables, syntax errors, etc.), the exception handling mechanism in the Java course will capture the exception at compile time or runtime and call the corresponding method to handle the exception. Exception handling can not only improve the maintainability of the code but also the usability of the program.

Through multiple teachings and example guidance, the traditional model of "teachers talk and students listen" has been changed, and the teaching method of demonstration, explanation, and practice has been adopted, creating opportunities for students to actively participate and independently cooperate, and transforming the teaching form of cramming into a discussion-type research type. According to the differences in students' practical ability and problem-solving ability, students are encouraged to complete more challenging optional projects on the basis of completing the required projects. The student-oriented teaching concept is conducive to stimulating students' interest in learning and mobilizing teachers' enthusiasm in teaching, thereby improving the quality of teaching and talent training. According to the requirements of the course content, the assessment method of this course is to combine the usual scores with the computer practice scores, the practical operation assessment with the theoretical assessment, and the basic skills and comprehensive ability evaluation. During the computer practice, the instructor is required to evaluate and comment on the operation skills and design ability of the students and comment on the problems that students are prone to occur.

4. Conclusions

Computer practice aims to cultivate students as engineers to consider and analyze problems, enhance their problem-solving skills and practical ability literacy. Based on the learning characteristics of students during the learning process, the teaching content and resources of the course are further optimized. The course teaching emphasizes new methods, new models, new means, and new thinking. It combines Java application examples and explains knowledge and technology closely related to software productization in a targeted manner, which is easily accepted and understood by students, thereby generating learning motivation and vitality. Through computer practice, students can connect theoretical knowledge with practical application, deeply understand the application of theoretical knowledge, correct the errors in the program, and witness the test and verification of the program they wrote. This is a highly fulfilling experience, stimulating students' enthusiasm for learning, improving their interest in programming learning, and cultivating students' basic quality of engineering practice ability. It lays a solid foundation for future study and software design work. In the subsequent teaching, we will focus on improving the quality of programming practice and continue to explore practical course teaching forms suitable for the training needs of applied innovative talents and the characteristics of students, so as to meet the personalized learning needs of students and make the practical courses more perfect.

Acknowledgment

This article was funded by the Faculty Development Research Project of the University of Shanghai for Science and Technology (CFCTD2024ZD01).

References

[1] Yang Yanqing, Guo Xianchong, Characteristics of Java Programming in Computer Software and Technical Analysis [J].Wireless Interconnection Science and Technology, 2021(3):54-55.

[2] Zhao Jiang. Application of Java Language in Computer Software Development [J]. Digital Technology & Application, 2023, 41(03): 160-162.

[3] Li Ruige. Application of Java Programming Language in Computer Software Development—Review of "The Methodology of Java Programming" [J]. Chinese Science and Technology Papers, 2022, 17(01): 134.

[4] Wen Xiumei, Meng Fanxing, Fu Jianglong, Gao Liting, Construction and Practice of High-quality Online Open Courses [J]. Science and Technology Wind, 2021(2): 64-65.

[5] Feng Rui, Ren Laibao, Overall Research on the Reform of Experimental Course System in Educational Technology Major [J]. Educational Technology Research, 1999(1): 38-41.

[6] Zhang Weimai. Reform of Experimental Teaching of "Object-Oriented Programming (Java)" Integrated with Course Ideological and Political Education [J]. Journal of Weifang University, 2022, 22(05): 52-56.