

Research on teaching of Java Exception Handling

Xiaoyuan Zhou

Lecturer, Institute of Computer Science and Technology Xi'an University of Science and Technology

Abstract

Exception handling is an important and difficult part of the Java programming's teaching and training. The paper points out and discusses the two outstanding problems in details during teachings, that are how can make students master exactly the basic ideas of exception handling and locations need to be handled by exception handling methods in Java programs. Furthermore proposes a kind of teaching process which gradually shows the puzzles during the learning of students starting from the case by combining the heuristic teaching with the practical case of significance, and figures out step by step the solving methods, namely the knowledge points, which have to be learned about exception handling. Visualization and simplification are their advantages, which facilitates the students' learning and understanding. The teaching practice shows that the good teaching results have been achieved.

Index Terms: Java exception handling; heuristic teaching; case teaching; teaching methods

© 2012 Published by MECS Publisher. Selection and/or peer review under responsibility of the International Conference on E-Business System and Education Technology

1. Introduction

Nowadays, Java programming language is one of the most important software development technologies. According to the list of global computer programming languages published by TIOBE which is the authoritative agency in assessing the popularity of software programming languages, shows that Java is absolutely the top one in the long-term[3]. Therefore in order to improve the students' competitiveness in employing, the courses relative with Java have been set in colleges and universities from the early years. Exception handling is an important part of an object-oriented software system, also is difficult to learn [12]. Years of teaching practices show that the methods how to make programs running continuously by eliminating the errors maximized are learned not well, because most of the students couldn't understand completely the basic thoughts of exception handling and realize the importance of programs' reliability, stability and fault tolerance. So, aimed at the problems rise from students' learning process, the paper proposes a kind of teaching method combining the heuristic methods with the case teaching ways, and then connecting and integrating all key knowledge points of exception handling into only one example. Hope these ideas and thoughts can be discussed with colleagues.

Corresponding author:
E-mail address: yuanyuanzhou@hotmail.com

2. OVERVIEW

According to the Java language's specification, the logic of exception handling is showed in the Fig. 1.

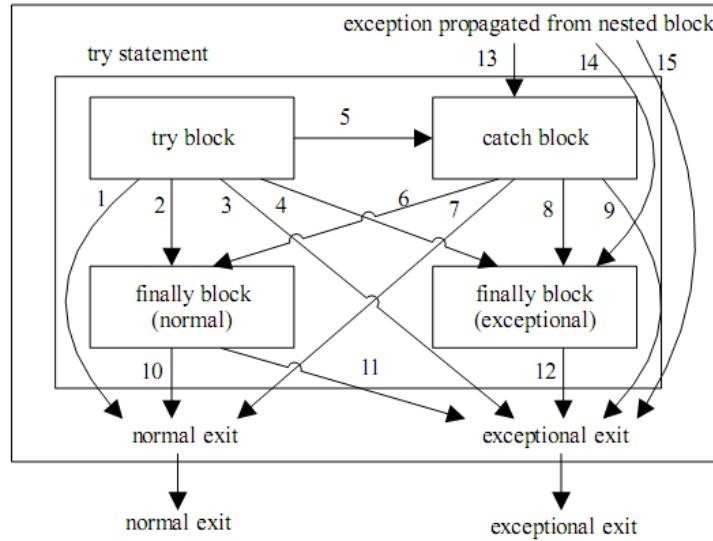


Figure 1. the logic of Java language's exception handling

Of which the numbers represent various different situations when need to handle exceptions. For example the number 1 represents try block raises with no exception and no finally blocks, the number 2 represents try block raises with finally block specified but no exception block, and so on[1]. In order to highlight the paper's focuses and shorten the paper size as well, the numbers' meanings (can be consulted from Java languages' specification as needed) are omitted. So, if interpreting the Java exception handling mechanism from the specification, the learning process would be boring and difficult. Moreover, university students are the group that lack of programming experiences and without any software projects' development experiences further.

Therefore, most of students only know the statements such as *try... catch... finally* after learning exception handling, whereas the main ideas of exception handling are not get and understand well. The following problems explain why.

- The point that the programs' reliability, stability and fault tolerance are more important than efficiency could not be realized thoroughly.
- When to need avoid or employ exception handling could not be distinguished completely, and which specific way should be used as needed is not known.
- For practical applications, the reasonable customized exception classes could not be figured out and throw the objects of them.

After referencing, comparing and synthesizing Java teachings of various universities and training institutions, the general teaching process of exception handling can be described in Fig. 2.

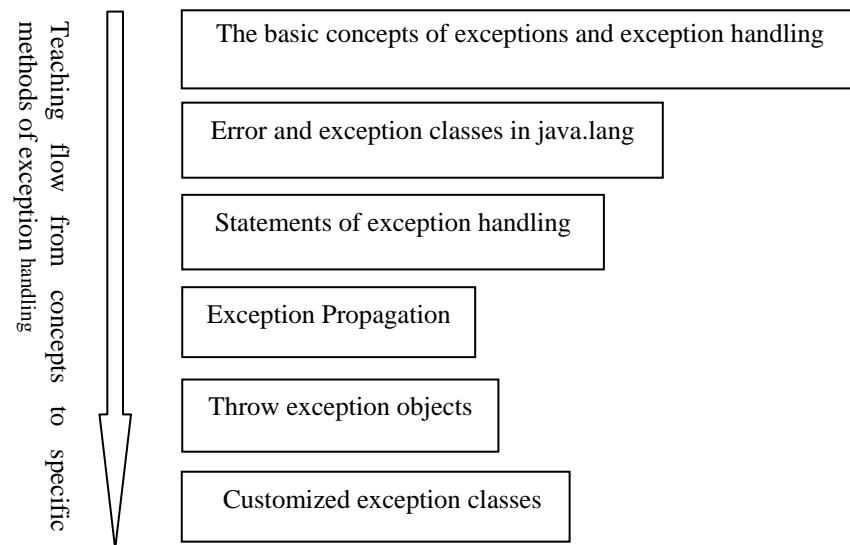


Figure 2. the general teaching process of exception handling in various universities and training institutions

From the aspect of natural learning laws and teaching practices, the teaching process above mentioned is reasonable in general. Then the following questions focus on how to visualize those abstract ideas and concepts into teaching process, and simplify the complex ideas. Additionally how to make students understand Java language's exception handling mechanism thoroughly, and even apply the main idea solving problems learned from Java lessons to many other ways in life.

3. methods of solving the problems

3. 1 Clarify the concept

“Exceptions represent problems or unusual situations that may occur in a program. Java provides various ways to handle exceptions when they occur”, “Java has a predefined set of exceptions and errors that may occur during the execution of a program, and those can be used to solve problems timely when exceptions occurs”, “The advantages of Java exception handling is making the program itself has the ability handling the exceptions, and separating the codes handling exceptions from the normal codes performing the required logic.”

In the normal teaching process, the similar description about the basic thoughts and advantages of exception handling as above shows are made by most of the teachers. Meanwhile teachers also emphasize the reliability and stability of the programs usually more important than efficiencies, and illustrate how exceptions occurred do harm to programs with an example showed in the Fig. 3 (a). The program terminates at 5th row when it attempts to divide by zero with no codes to handle the exception explicitly. And then to solve the problem, the statement (try...catch...) can be used to modify the program as shown in Fig. 3 (b).

Although the example or others like that explain what is the exception and how to handle it, fault impressions that use the statement(try...catch...) to improve the reliability under any circumstances left to the beginners and result in the abuse of it.

Actually some exceptions are checked, whereas others are unchecked. Unchecked exceptions refer to those caused by the program itself, and most of them can be avoided by expecting and processing ahead during programming. So there is one point should be emphasized, that is, exception handling is not necessary in any cases

Comparing the two pieces of codes showed in Fig. 3 (b) and (c) enable students know exactly that those errors caused by the program itself such as dividing by zero or array operations(actually belong to unchecked exceptions) should be expected and handled ahead. On the contrary, those errors caused by the program environments such as the data input from users, to open the specified file or internet link, etc. should be dealt by Java exception handling. So, an important conclusion which illustrate various situations and the corresponding handling methods, conducted together with students. As Fig. 4 shows, exception handling is a kind of methods solving problems after exceptions occur.

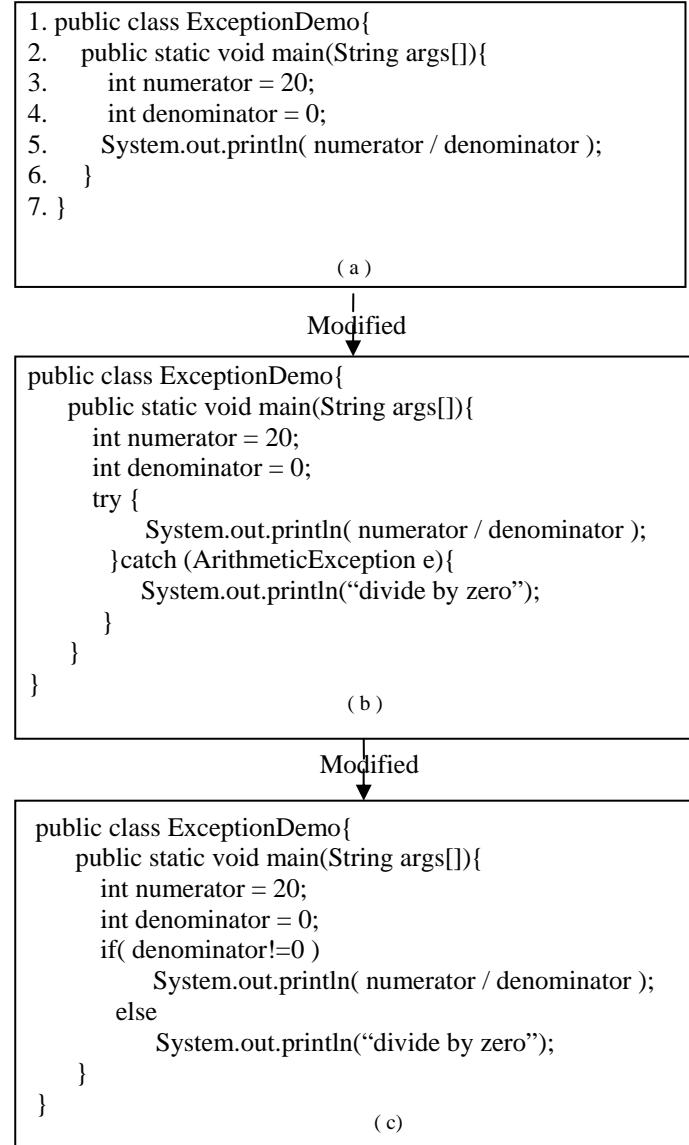


Figure 3. the usual example used to explain the harms to the program once exceptions occurred and a better way is to use if else statement

3.2 Appropriate teaching cases

Java provides the mechanism of run-time errors at language-level, including exception classes, the specified statement-*try...catch... finally*, handling exceptions, the *throw* statement used to throw an exception object specifically and a *throws* clause appended to the header of a method definition[2].

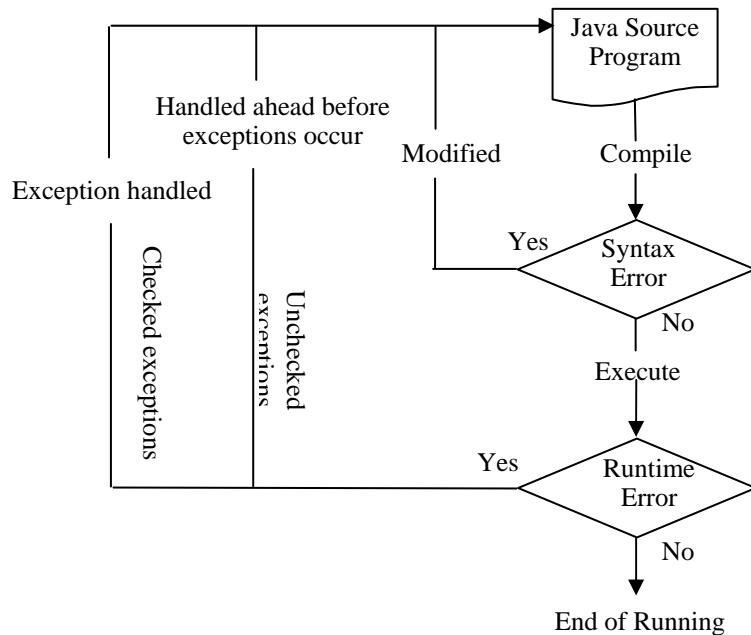


Figure 4. Various exceptions occurred in programs and the handling methods

Generally from the teaching effects' point of view, the try statement is mastered well as long as students are aware of the timing and location of exceptions handling, while the throw statement and throws clause are understand worse contrarily. Thus appropriate and practical teaching case throughout the whole knowledge points is good for learning. From the examples surrounding students, the paper design and implement a Student class that abstract and encapsulate all students, of which an instance variable with type *int* declared to store the age. Since the range of the data expressing age maybe too large, a conditional action has to be done to substitute the inappropriate age with the value of 0 as Fig. 5 (a). However the value setting like this without notice users is forbidden absolutely in practical applications.

Consequently a better way is throw an exception object with error information when an argument age is not appropriate just like showed in Fig. 5 (b). Additionally, how to implement that in the code is the next question need to be discussed further, which resulting in teaching *throw* statement.

The code in Fig. 5 (b) actually leads to three knowledge points: (1) throw an exception actively; (2) the exception propagation; (3) customized exception classes. Java asks that the exception object thrown actively must be catch or handle, so where on earth to add try statement in the code? As it is users who enter the information age, the problem of age out of bounds can not be solved within the method set(), then it must pass the exception object to the calling program to deal with. Fig. 5 (c) shows how to declare the exceptions at the header of method definitions which is the problem of exception propagations. The calling program has to be modified as Fig. 5 (d), because of the inappropriate data rise inside itself.

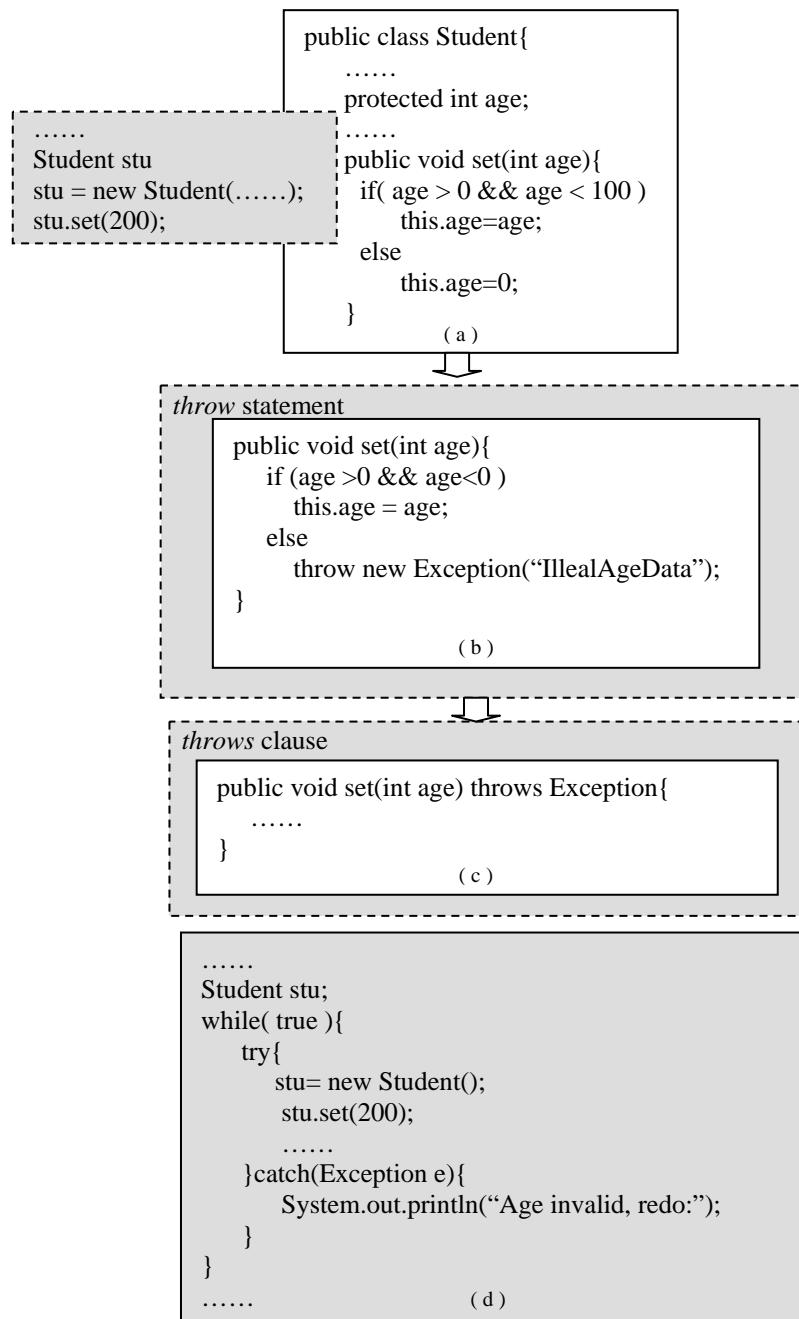


Figure 5. Appropriate and practical teaching case throughout the whole knowledge points

Since the class of the object thrown from the method set() is the subclass of Exception that is the super class of all exception classes, the calling program do not recognize at all what kind of exception happened while an exception object involving age out of bounds is expected. So the exception classes provided in Java are not

inadequate for programming needs actually. Then the need for the customized exception classes is educed as shown in Fig. 6.

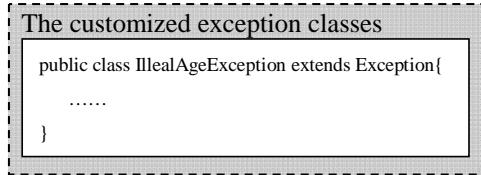


Figure 6. The customized exception class-IllealAgeException

Finally, in GUI programming, the users do not see any responses once an exception occurred without catching, which resulting in the program has terminates actually. So the normal method adopted in GUI programs to treat exceptions should be discussed further.

4. Conclusion

Exception handling is an important and difficult part of the Java programming's teaching and training, which involving many aspects from the concept of exceptions, the main idea of exception handling to many knowledge points in exception handling mechanism. General ideas and structures of teaching exception handling are outlined in the paper after referencing, comparing and synthesizing Java teachings of various universities and training institutions. Aimed at the outstanding problems in the teaching, the paper proposes a kind of teaching process which combines the heuristic teaching with the practical case of significance and then figures out gradually the solving methods have to be learned in Exception handling. The teaching practice shows that the good teaching results have been achieved.

References

- [1] James G, Bill J, Guy S, "The Java Language Specification", 3rd ed. [S. l.]: Addison Wesley, 2006.
- [2] Ye HeYa, "The practical tutorials of Java programming", 2nd ed. Publishing House of Electronics Industry, 2007
- [3] Chen Fen, He Hongjie, "Application on Heuristic Teaching of Java Course", Computer Education, Chinese, No.16, pp. 81–83, August 2010.
- [4] Cao Zhiwei, Yang Keqiao, WangWei, Zhou Xun, "Algorithm for Impementing Java Exception Mechanism in Static Compiler", Computer Engineering, Chinese, Vol.35, No.15, pp.88–90, August 2009.
- [5] Bruce Eckel, "Thinking in Java", China Mechine Press, 1999
- [6] Liu Yue, Chen Huaiyi, "Research of Foundation of Software Technology Innovation", Journal of Jilin University(Information Science Edition), Chinese, Vol.23, No.S0, pp. 145–148, August 2005.
- [7] Liu Zhicheng, Weng Jianhong, "The Case-design of Java Programming Curriculum", Computer Learning, Chinese, pp.29–32, February 2008
- [8] Jiang Guoquan, Li Yamin, Zeng Lihua, Song Zhiguo, "Discussion on exception handling of Java", Information Technology, Chinese, pp. 109–111, October 2005
- [9] He Jianying, "How to Improve Practice of High Student about Java Lauguage", Computer Teaching and Educational informationization, Chinese, pp. 95–97, April 2008
- [10] Li Wenfeng, "The Characteristics of Object Oriented Programming in Java", Journal of Shanxi Institute of Economic Management, Chinese, Vol.13, No.3, pp.59–60, September 2005
- [11] Liao Jianwei, Cai Hongbin, Jiang Pandeng, Zhou Mingtian, "Design and Implementation of Java-based GridMonitoring System", Computer Studies and Applications, No.12, pp.234– 237, 2005
- [12] John Lewis, William Loftus, Java Software Solutins Foundations of Program Design, 6th ed. Publishing House of Electronics Industry, 2009