**LAB9\_assignment**

|  |  |  |  |
| --- | --- | --- | --- |
| Name: | 신민선 | Department: | 미디어학과 |
| Student ID: | 201723307 | Room Number: | 팔달관 333호 |
| Due Date: | **Saturday, May 11, 2019** | | |

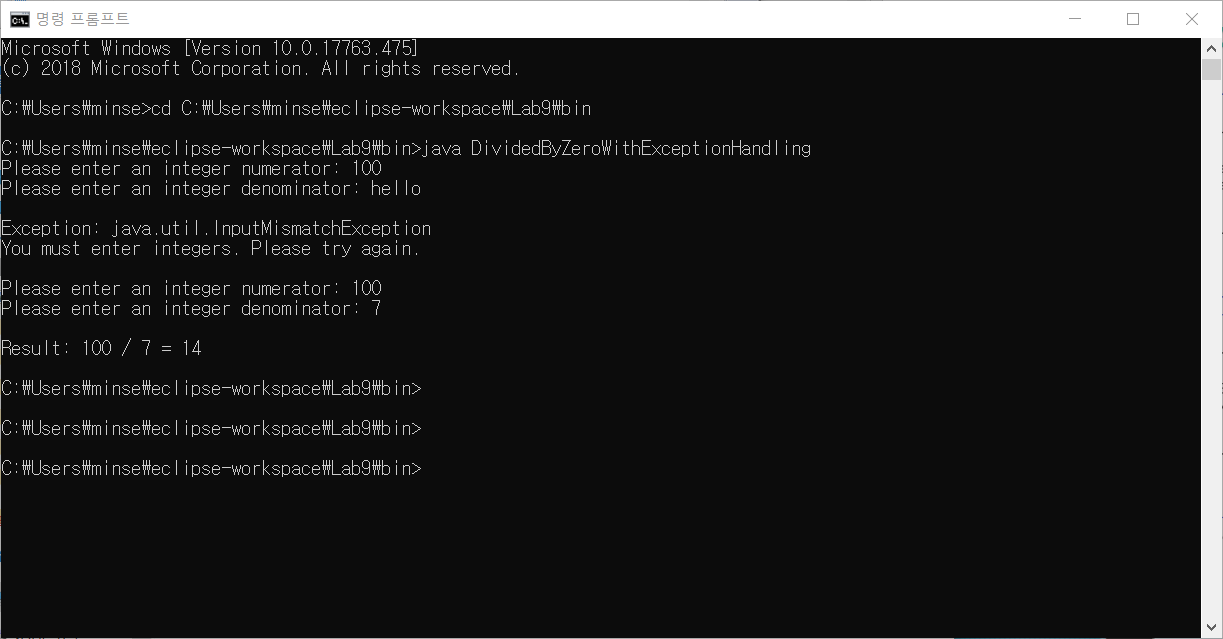
1. **Answer the following questions after executing the code in Fig 11.3**
   1. **At which line number the “ InputMismatchException ” occur? (1 point)**

Line 27: scanner.nextInt();

* 1. **Why the “InputMismatchException” occurs? (1 point)**

InputMismatchException is about the input data that don’t same type of input(int).

In this code, we have to get data type integer, but if we get other data type like string, we got this Error.

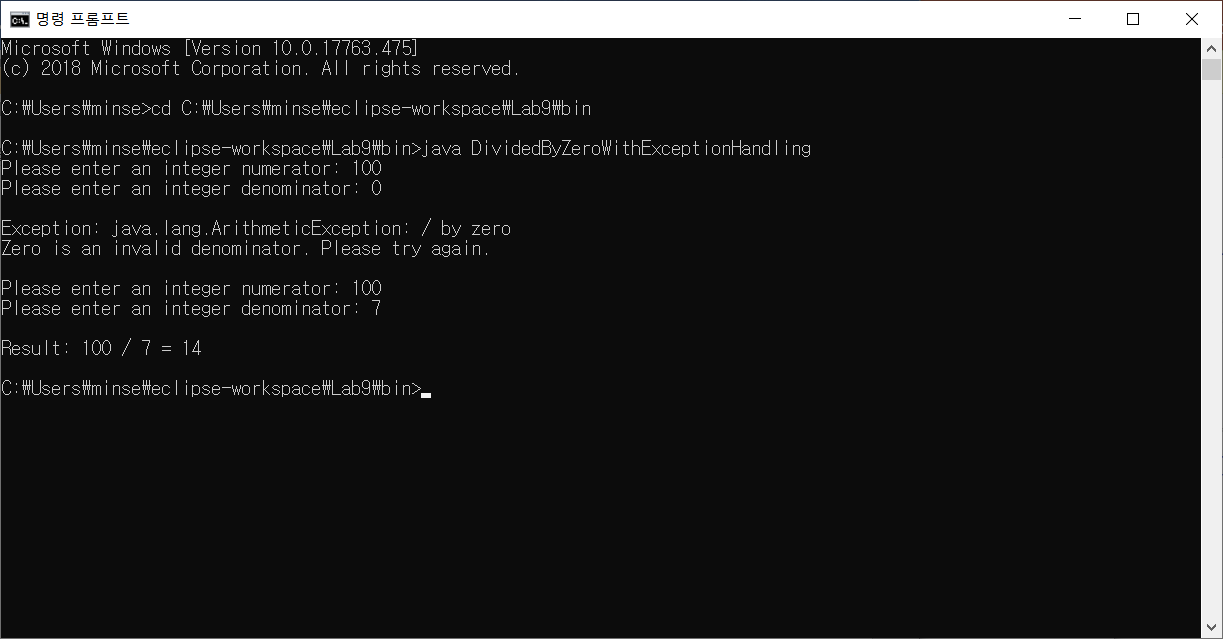


* 1. **At which line number the “ArithmeticException” occurs? (1 point)**

Line 29 : int result = quotient(numerator, denominator)

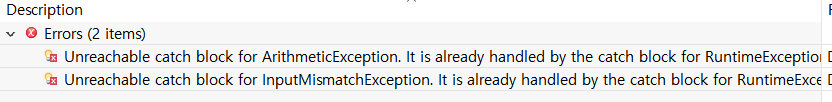
* 1. **Why the “ArithmeticException” occurs. (1 point)**

When we divide number to another number, if one divided zero, the result came infinity. At that time ArithmeticException occers. So if complier will try to divide any number to zero, it will occur.



1. **Answer the following questions about the program in Fig 11.3**
   1. Before line 34, **add** the following catch block and compile the program. Explain what happens and explain why this happens (including only explanation, **2 points**)

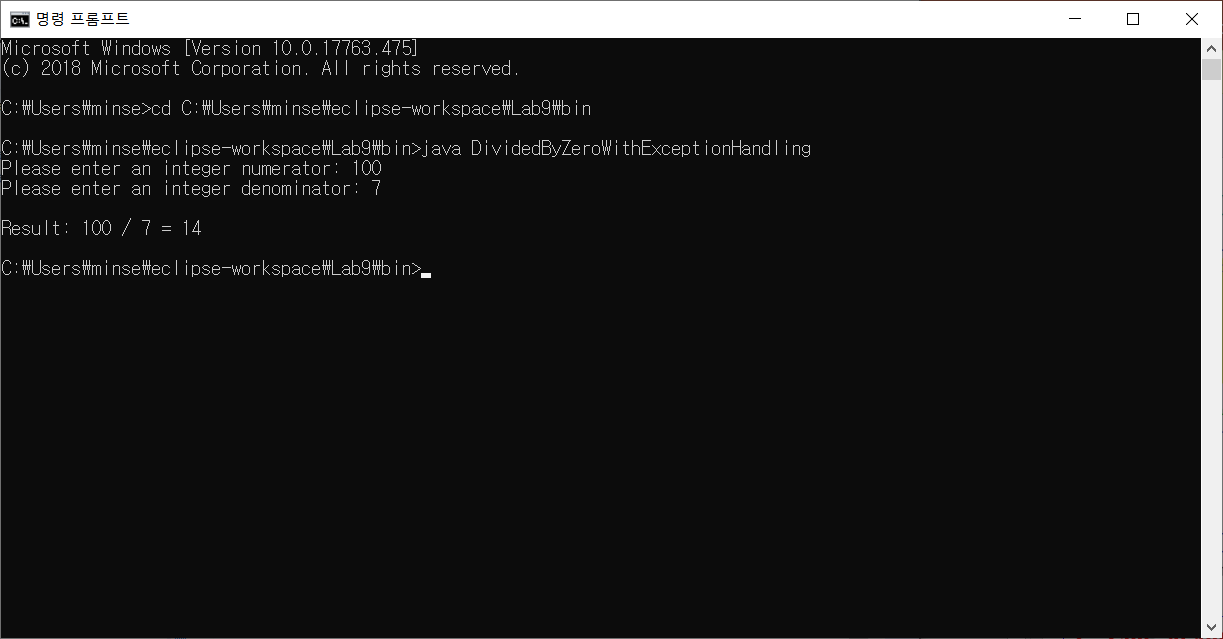
It comes Error.



It means that Runtime Error is more comprehensive than ArithmeticException, InputMismatchException. So, when you write catch line, you have to write down form intensive to comprehensive.

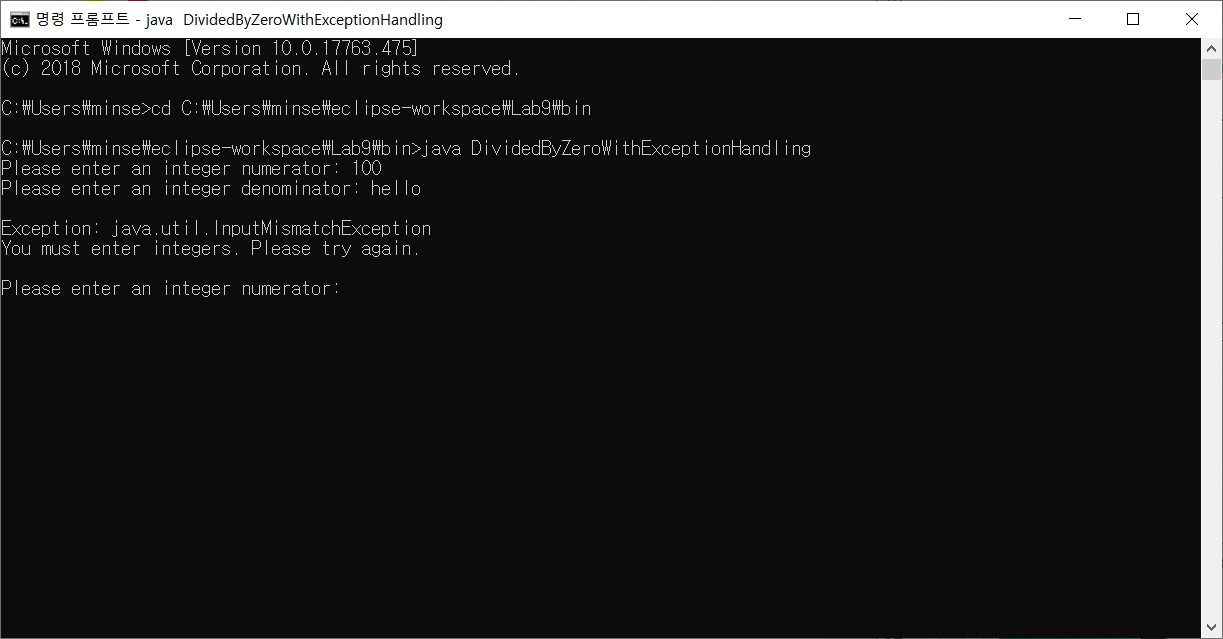
* 1. Replace the **second** catch block (Fig.11.3) by the following catch block. Then **enter** input similar values as shown in the example and execute the program. **Explain** what result you observed and explain why such type of results occur (including the **reason** and **screen shot of the result , 2 points**)

First, no Error

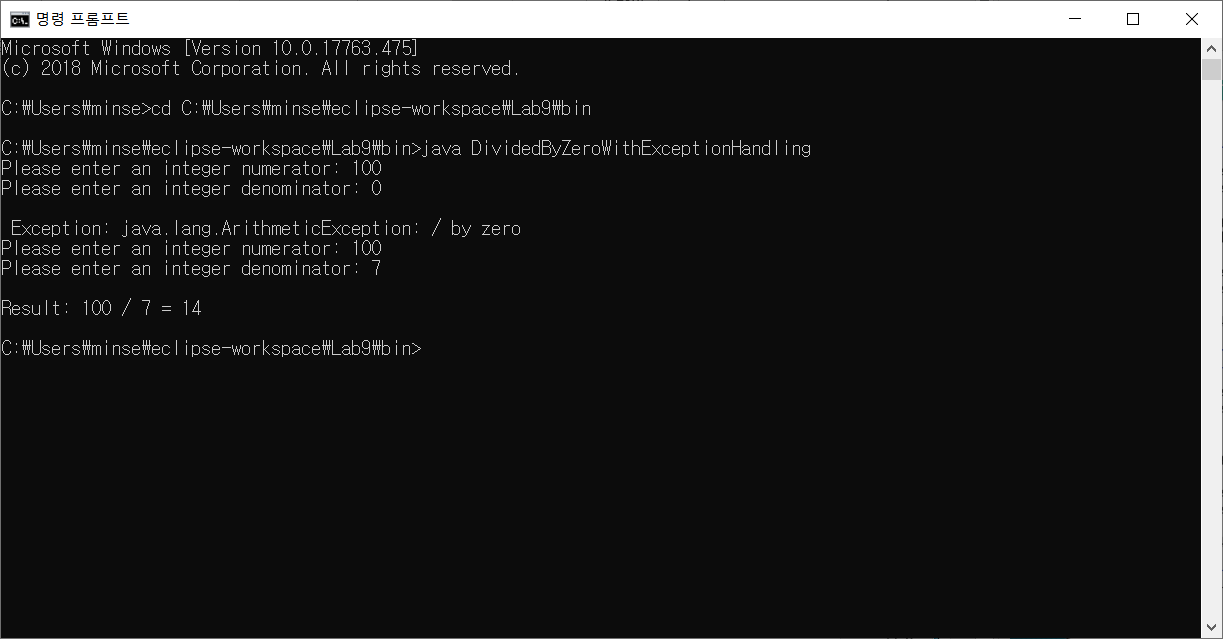


Second, If I input hello not integer number,

they are Error throws **InputMismatchException**



Third, after revise, they don’t have ArithemeticException. But they came out the Error throws ArithemeticException. Because quotient method throws ArithemeticException from java. So they came out that codes.

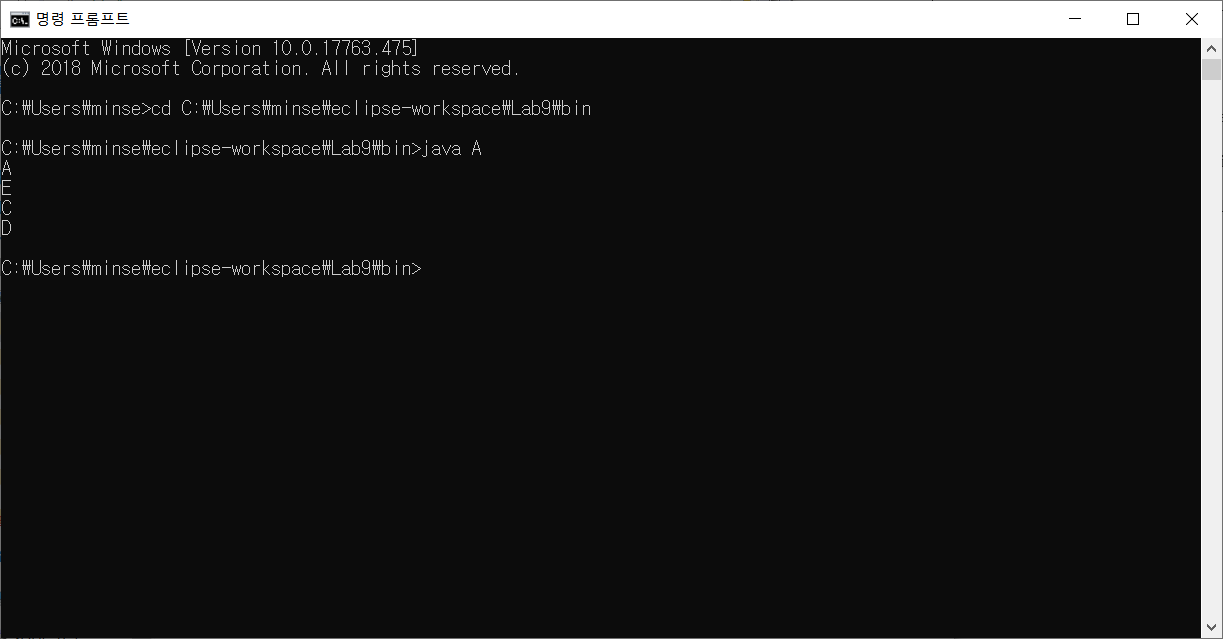


1. After deleting “**throws Exception**” at line 21(Fig 11.5), compile the program. Explain why a problem happens (**including only explanation, 2 points**)

An Error occurs. When we delete "throw Exception", it can't execute by Exception at try-catch in throwException method. In throwException method, it tries new exception(). After that, if it has an Error, it passes throw catch(Exception exception) line.

At this situation, compiler occurs an error. so if we revise this code, we have to change from Exception to detail Exception point. Or we just add "throw Exception" in codes.

1. After **running** the following code, explain the reason why you got this result (including your reason and screen shot of the result, **3 points**)



First, in line “try”, print “A”

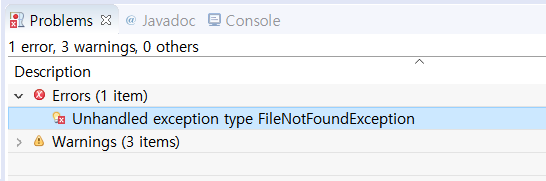
And then through m();, print “E” and then it is true, so throw new RuntimeException();

Re-throw for further processing.

Therefore from catch (exception e), they catch the Error so they print “C”

Finally they go finally area, print “D”, It is over.

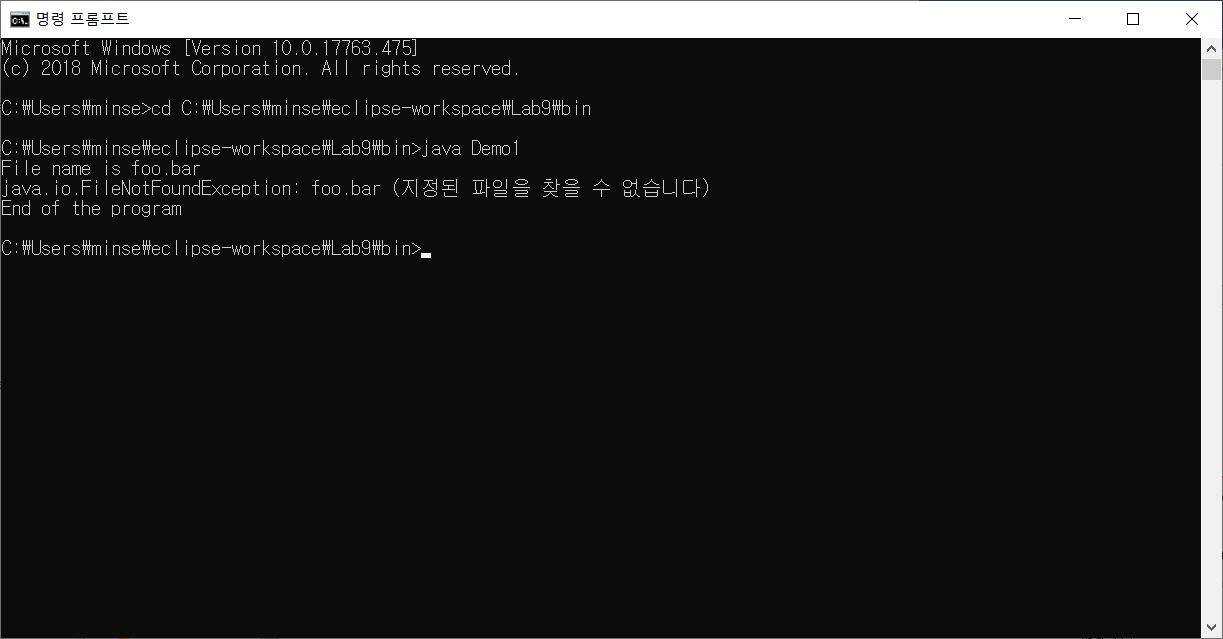
1. **When you compile the following program, an error occurs.**
   1. Explain **why** the error happens.

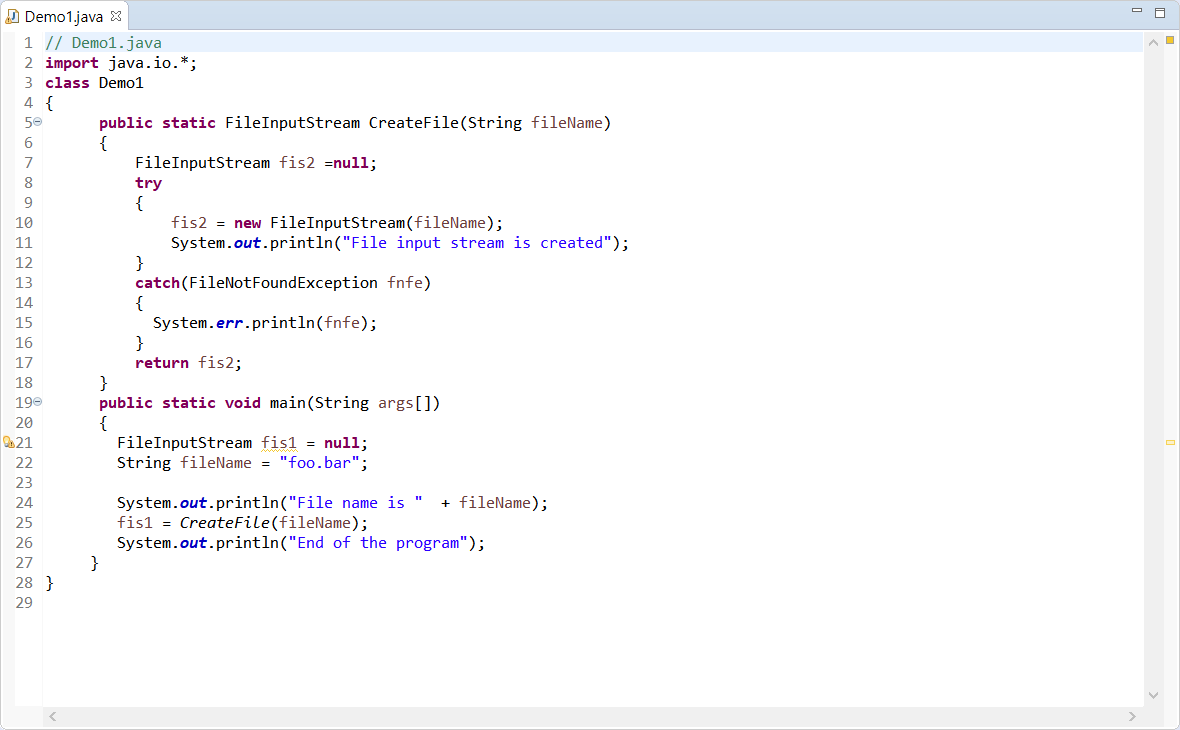


It can occur that they need Exception for File can’t normally open when compiler don’t know where file is or some kind of different reason. In this case, we have to add Exception.

* 1. In order to do **normal** operation, **add** the exception handling code using two ways. In your report, include screen shot of errors, explanation for the occurrence of the error, screen shot of the normal code (without error). Include also the source code of the normal code in the source file (**4 points**)

Case 1) using try-catch

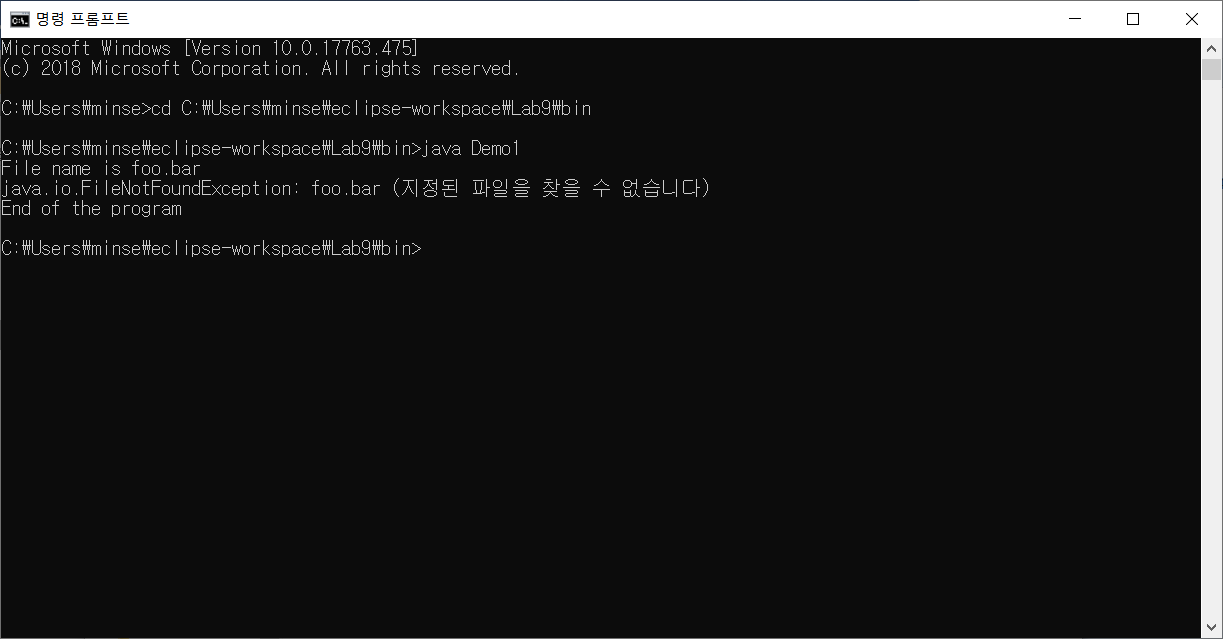


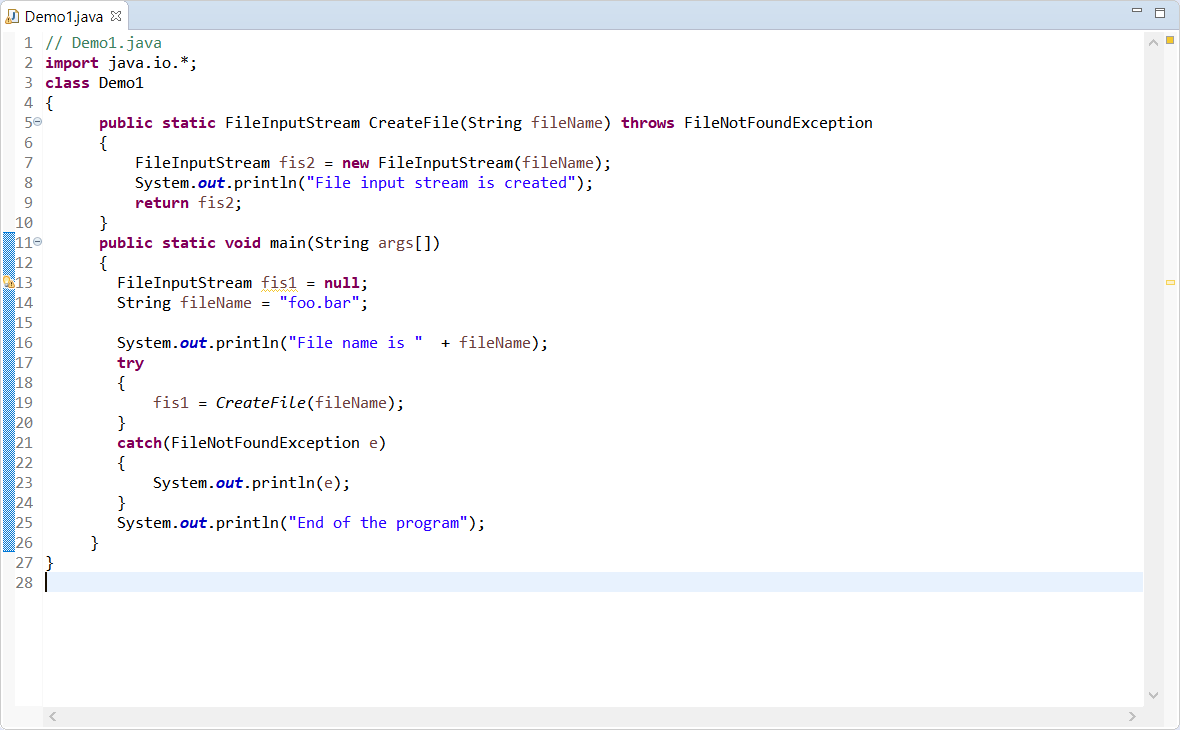


First, create FileInpuStream, and then print file name. After then, when they call Createfile function, move to line 5.

Line 7, they create new FileInputStream fis2 to null. And input new create FileInputStream that has parameter named filename(=foo.bar). But they can’t fine foo.bar file, so they catch from FileNotFoundException. So they print err. And then print End of the program in line 26.

Case2. Use throw Exception





Just use throws FileNotFoundException.

What is different is that just write on the main method in codes.

In the main method, create new FileInputStream and make it null. And then, print file name and use try-catch from throws FileNotFoundException. So first do "fis1 = CreateFile(fileName);" line. if it can't find file location or and so on, they pop up the error. "catch" line will catch the Error and then print exception e. Finally, print "End of the program."