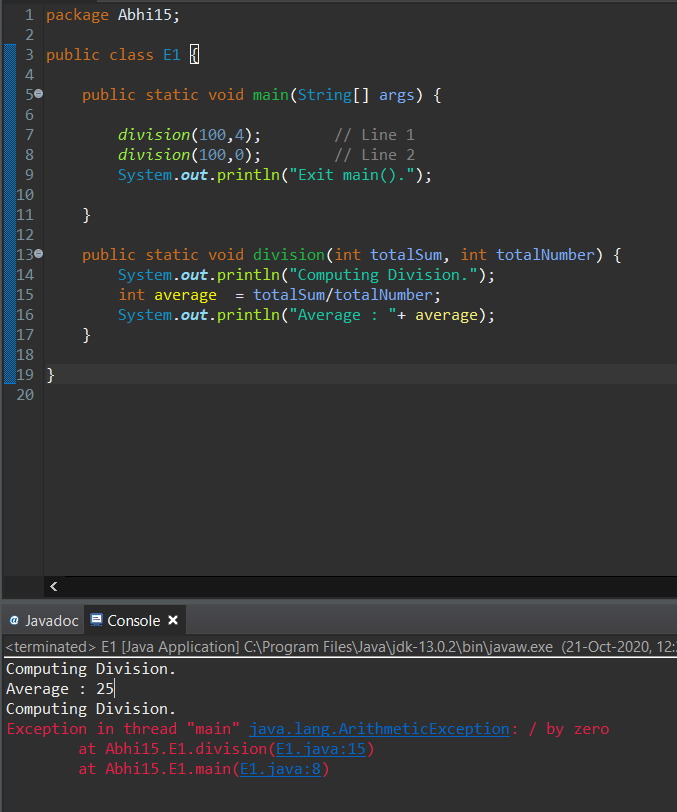
**LAB SHEET 9 -** Exception

S. Abhishek

AM. EN. U4CSE19147

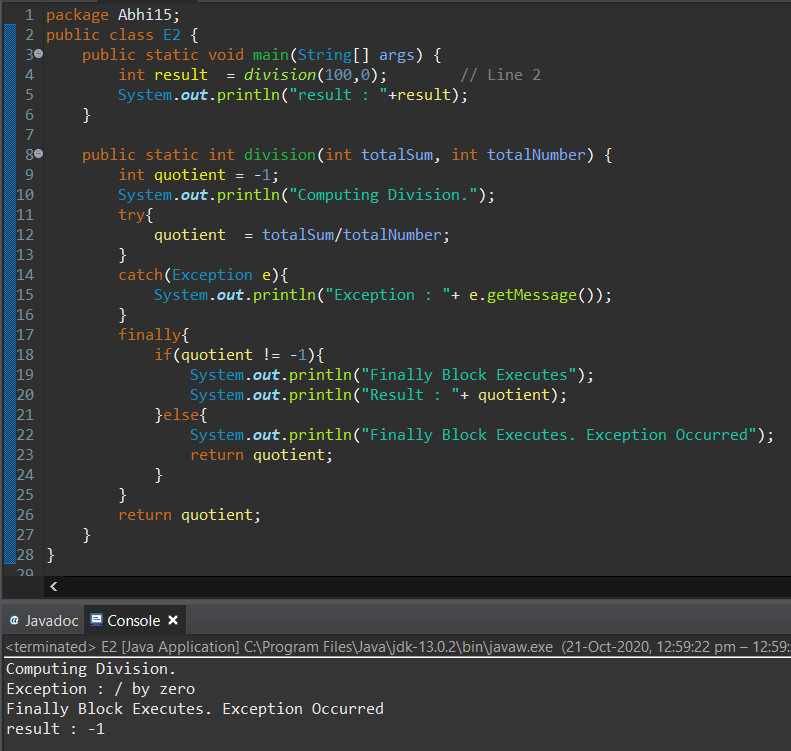


Here we have defined the method and the print statement in the class “division”.

We are trying to compute the value from the Main class.

For the First Testcase it works fine and for the second it throws the error because we are trying to divide the given input by 0 which basically comes under the “Arithmetic Exception”.

Here we have not provided any Exception and by default the JVM compiler throws an error.



Here we have defined the method and the print statement in the class “division”. We are trying to compute the division from the Main class.

In the division class we have defined “try”, “catch”, “finally” keywords which basically tries the given computation given inside the “Try” block and then catches the Exception and prevents the program from further termination.

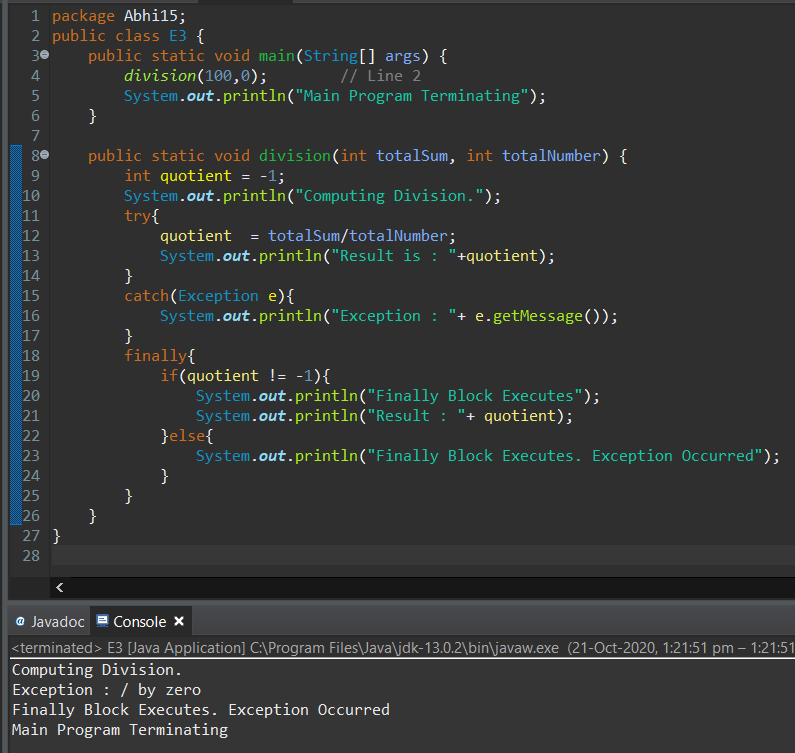
The **finally** keyword is used to create a block of code that follows a try block.

A finally block of code always executes, whether or not an exception has occurred.

Here the Exception is “/ by Zero which comes under the arithmetic type” and “e.getmessage()” which the calls the “getmessage()” inbuilt function from the instantiated object “e” and prints the type of Exception occurred.

At last “finally” is executed and since the “quotient = -1” it prints the statement mentioned inside the “Else” block which is "Finally Block Executes. Exception Occurred".

Then finally it returns back to the main class and there it prints the value returned value “-1”.



Here we have defined the method and the print statement in the class “division”. We are trying to compute the division from the Main class.

In the division class we have defined “try”, “catch”, “finally” keywords which basically tries the given computation given inside the “Try” block and then catches the Exception and prevents the program from further termination.

The **finally** keyword is used to create a block of code that follows a try block.

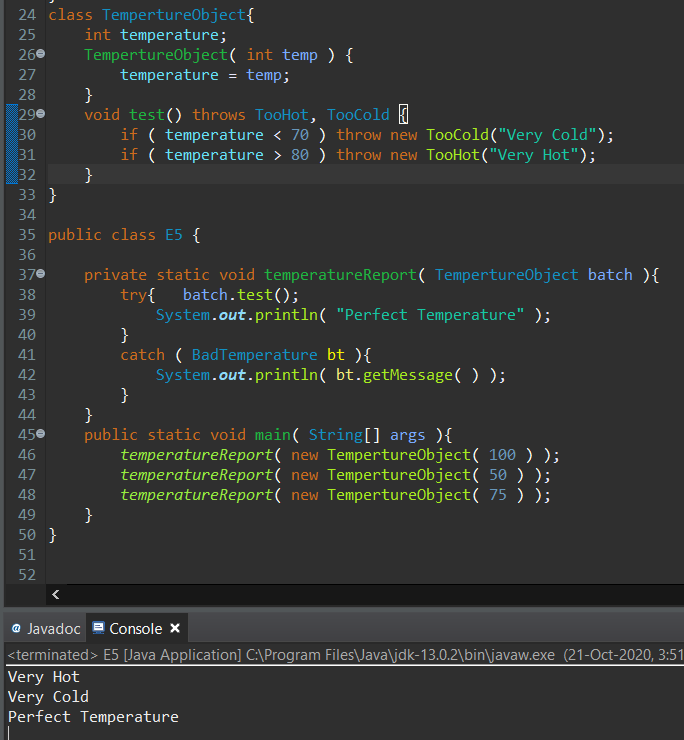
A finally block of code always executes, whether or not an exception has occurred.

Here the Exception is “/ by Zero which comes under the arithmetic type” and “e.getmessage()” which the calls the “getmessage()” inbuilt function from the instantiated object “e” and prints the type of Exception occurred.

At last “finally” is executed and since the “quotient = -1” it prints the statement mentioned inside the “Else” block which is "Finally Block Executes. Exception Occurred".

Then finally it returns back to the main class and there it prints the statement “Main Program Termination”.





Here we have created our own exception classes and we are calling it based on the given condition.

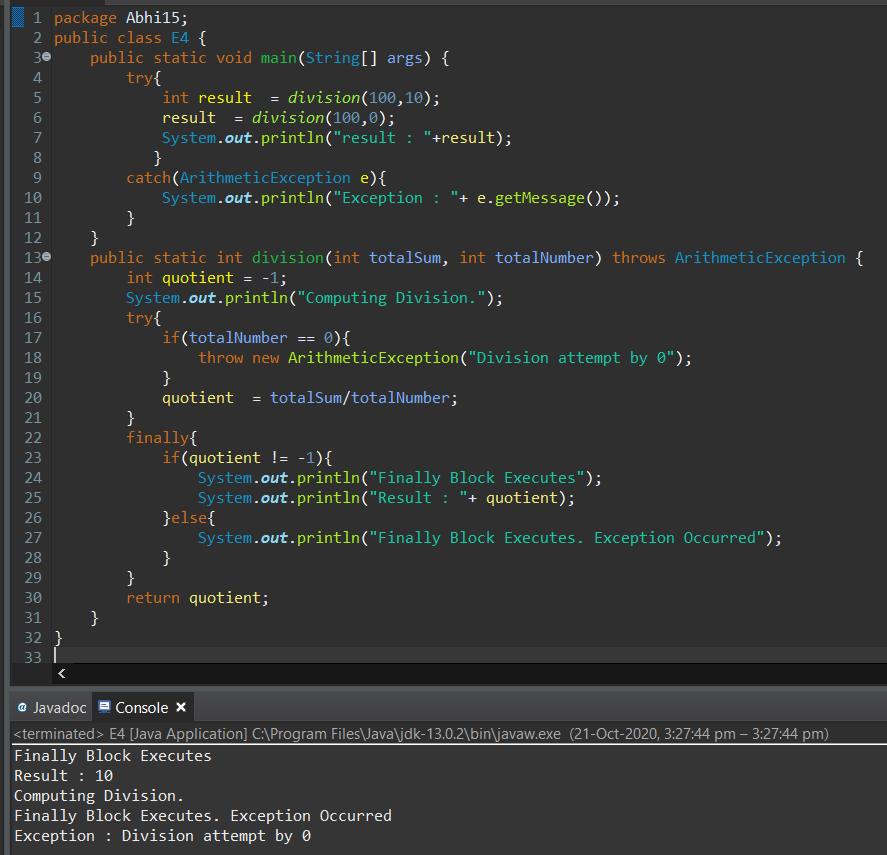
We are calling the “temperatureReport” class using constructor which inturn calls another class “TemperatureObject”.

There it checks the temperature and calls the required Exception accordingly.

We have extended the Exception with new two user defined Exceptions.

We have created the “badtemperature” by extending from the Exceptions and then we are extending two different other classes from the “badtemperature” class and then defining our own Exception message.

Then finally we are printing the report bsed on the temperature and Exception.



Here we have defined the method and the print statement in the class “division”.

We are trying to compute the division from the Main class. We have two test cases and for the first Testcase it works fine.

So it goes in to the final block directly since the denominator is not 0 and it prints the line “Finally Block Executes” and prints the Quotient there itself.

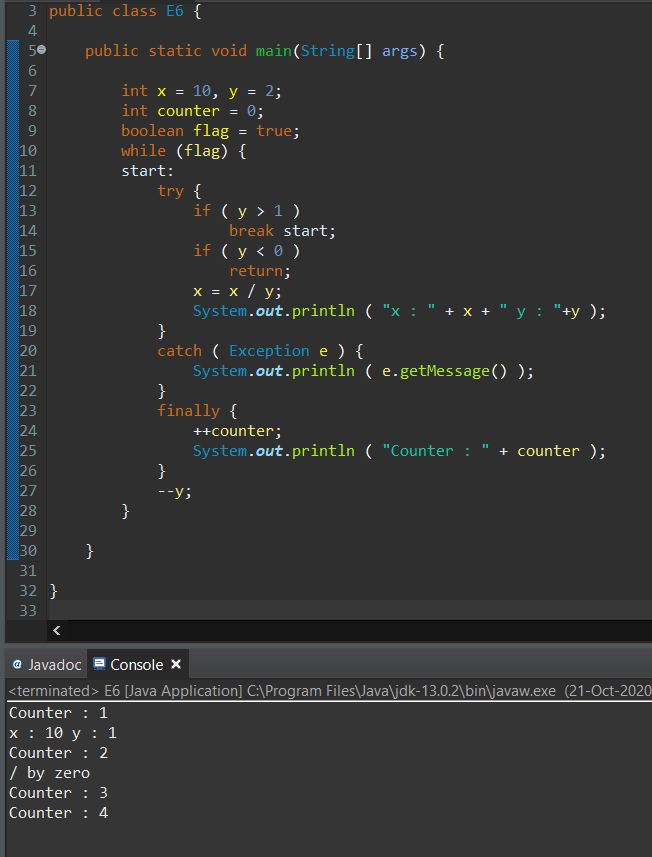
For the second test case it throws the error because we are trying to divide the given input by 0 which basically comes under the “Arithmetic Exception”.

Here it’s the user defined Exception is used.

Thethrows keyword is used to declare an exception. It gives an information to the programmer that there may occur an exception so it is better for the programmer to provide the exception handling code so that normal flow can be maintained.

And when it encounters the exception it prints the “Computing Division” line and enters into the else condition of the “Finally” block and then prints “Finally Block Executes. Exception Occurred”.

It returns back to the main class and then prints the result.



Iteration 1: Since “y” is greater than 1 it gets into first if and then gets break and comes to finally.

There the counter value increases to 1 and it gets printed. The value of “y” is decremented and again continues to Iteration 2.

There the value of “y” is 0 and thus throws the “/ by Zero” Exception and prints the Error.

In the next Iterations the value of “y” becomes less than 0 and gets returned.

