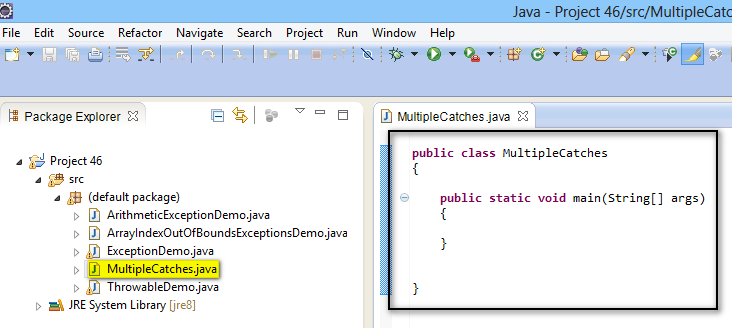
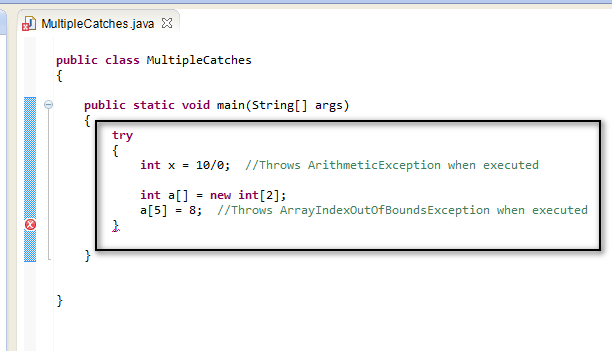
 Using multiple catch blocks

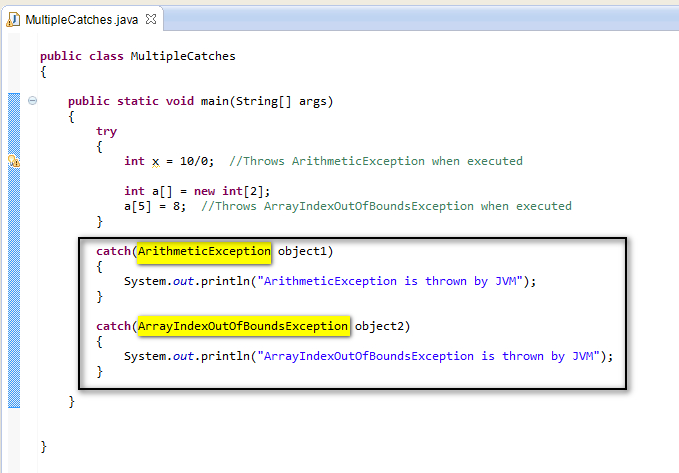
When your statements in the try{ } block can throw more than one exception as shown below -  
  
try  
{  
     int x = 10/0; *//Throws ArithmeticException*  
     int a[] = new int[2];  
     a[5] = 8;  *//Throws ArrayIndexOutOfBoundsException*  
}  
  
To handle the above situation, you can specify two or more catch blocks after the try block as shown below -  
  
catch(***ArithmeticException*** object1)  
{  
      System.out.println("ArithmeticException is thrown by JVM");  
}  
catch(***ArrayIndexOutOfBoundsException*** object2)  
{  
      System.out.println("ArrayIndexOutOfBoundsException is thrown JVM")  
}  
  
System.out.println("rest of the code");  
  
  
On executing the above try{ } and multiple catch{ } blocks, the following will occur -  
  
1. First statement in the try block gets executed and throws Arithmetic exception  
2. First catch block handles the Arithmetic exception  
3. Second exception in the try{ } block i.e. ArrayIndexOutOfBoundsException wont be thrown as after the first catch block handles the arithmetic exception , the control directly goes to the System.out.println("rest of the code"); statement  
  
*Conclusion* - Even though the code in the try block has statements which can throw more than one exception, only one exception will be thrown by the try block and hence only one catch block will get executed. So, try block throws an exception which occurs first in its code.  
  
Lets Implement the above example on Eclipse IDE -  
  
1. Launch Eclipse IDE, create a Java Class 'MultipleCatches' with main( ) method in the existing Java Project 'Project 46' as shown below -

[](https://1.bp.blogspot.com/-7Os2fNVMyCQ/U-tgtoZrQbI/AAAAAAAAbwk/9ggEVDyOPl8/s1600/1.jpg)

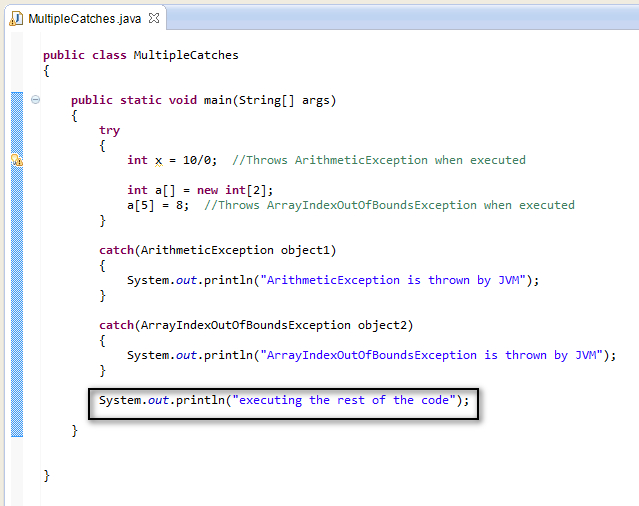
2. Write statements in try block which can throw more than one exception as shown below -

[](https://1.bp.blogspot.com/-9pUaIiyO7Cc/U-t4B0D6TQI/AAAAAAAAbw0/DHdWlA6xiyU/s1600/2.jpg)

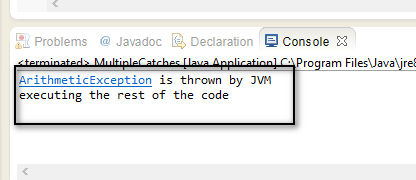
3. Create two catch blocks which can handle the above two expected exception as shown below -

[](https://4.bp.blogspot.com/-3dr-fV6urxo/U-t50X8KajI/AAAAAAAAbxA/CqD1DyY0hNo/s1600/4.jpg)

4. Write a statement which need to be executed after handling the exception after the catch blocks as shown below -

[](https://4.bp.blogspot.com/-nVlVCukmPBw/U-t6YuILcdI/AAAAAAAAbxI/FuHurZZWeJM/s1600/5.jpg)

5. Save and Run the 'MultipleCatches.java' file and observe that the following output is displayed in the console as shown below -

[](https://1.bp.blogspot.com/-hV2_pkfBAro/U-t646M-2XI/AAAAAAAAbxQ/RjSRQEhYwtw/s1600/6.jpg)

Understanding the above output -  
  
Only one exception is thrown by the JVM i.e. ArithmeticException, even though the code in try block can throw two exceptions. Hence try block throws the exception which occurs first in the code. Once the catch blog gets executed, the control will go to the statement that is written after the catch blocks. Hence second catch block is not executed and 'executing the rest of the code' is printed.