 Exception Class

Handling exceptions using Exception Class for safe side

In some case, when unexpected exceptions are thrown from the try block, the specific exceptions may not handle the unexpected exception which need to be handled by a different exception Class. So, for safe side, we can write the catch blocks which can handle the expected exceptions using the provided specific Class and at the end of specific catch blocks, we can write the catch block which can handle all the unexpected exceptions using the generic Exception Class.

Example -

Lets say, you have not expected ***ArithmeticException***and created a catch block believing that ***ArrayIndexOutOfBoundsException****will occur* as shown below -

try

{

      int a[ ] = new int[2];

      a[5] = 8/0;  //Throws Arithmetic Exception

}

catch(***ArrayIndexOutOfBoundsException***object1)

{

      System.out.println("Exception is handled by ArrayIndexOutOfBoundsException");

}

System.out.println("rest of the code");

Output -

On executing the above code, Arithmetic Exception will be thrown by try block and as there is no specific catch block to handle the Arithmetic Exception, the program will terminate without executing the program steps which are after the catch block in the program.

In order to avoid this kind of situations, we have to write an extra catch block having Exception Class at the end of all the catch block. As Exception Class can handle all kinds of exceptions handled by its sub-classes like ArithmeticException Class etc. Using that as a safe side can help us in handling the unexpected if occur any as shown in the below example program -

try

{

      int a[ ] = new int[2];

      a[5] = 8/0;  //Throws Arithmetic Exception

}

catch(***ArrayIndexOutOfBoundsException***object1)

{

      System.out.println("Exception is handled by ArrayIndexOutOfBoundsException");

}

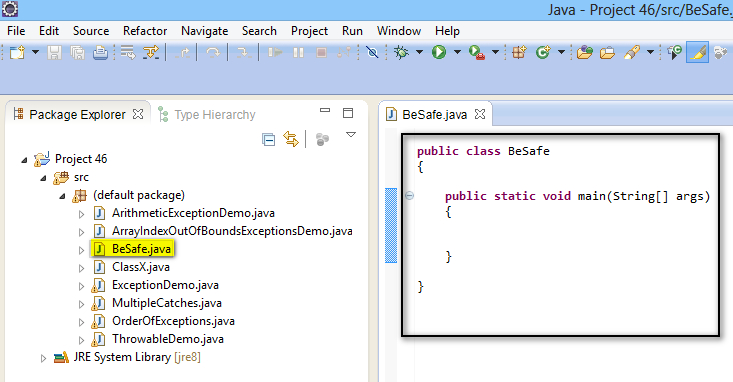
catch(***Exception***object1)

{

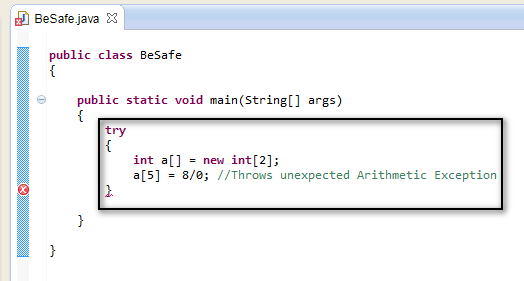
     System.out.println("Exception handled by Exception Class - "+object1);

}

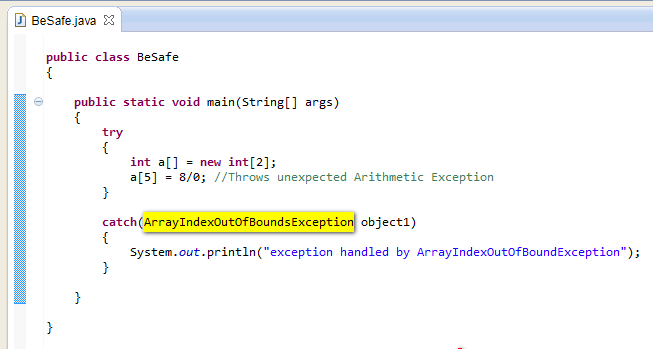
Output -  
  
Though the unexpected specific ***ArithmeticExcption*** is not handled by any catch block, the occurred unexpected exception will be handled by the catch block having the Exception Class. Hence the program wont get terminated and the steps after the catch block in the program will get executed.  
  
Lets implement this on Eclipse IDE -  
  
1. Launch Eclipse IDE, create a Class 'BeSafe.java' with main( ) method in the existing Java Project 'Project 46' as shown below -

[](https://2.bp.blogspot.com/-iPzxWaf4gnw/U-xasod5bVI/AAAAAAAAbyk/QEMNxFCda-g/s1600/1.jpg)

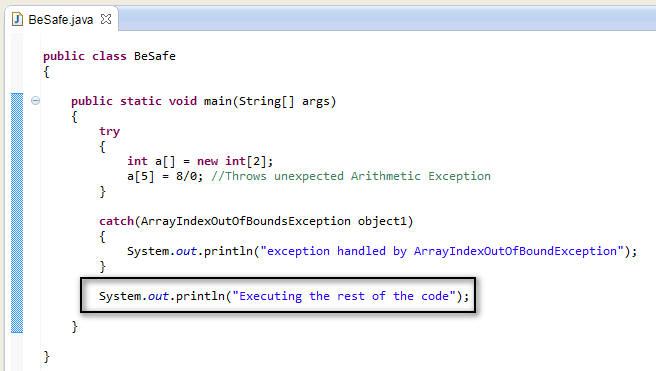
2. Create a try block which can throw two types of exceptions a) ArrayIndexOutOfBoundsException (Expected by User) and b) ArithmeticException (Not expected by user that it will trigger) as shown below -

[](https://2.bp.blogspot.com/-Aduxn2b81tE/U-xbdb4nLOI/AAAAAAAAbys/lw5PSRG6Phc/s1600/2.jpg)

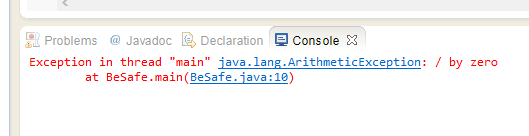
3. As User is only expecting that the statements in try block will throw ***ArrayIndexOutOfBoundsException***only,  write the catch block for it as shown below -

[](https://1.bp.blogspot.com/-VdI_Fqrhzdw/U-xdhGgbHLI/AAAAAAAAby4/ZdW_-__aFmg/s1600/3.jpg)

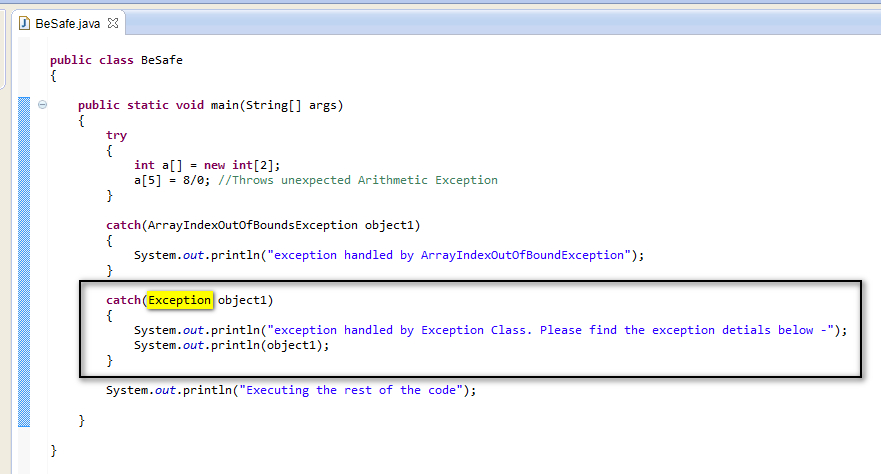
4. Write a statement after the catch block, which need to be executed if the occurred exception is handled by the catch block as shown below -

[](https://3.bp.blogspot.com/-XooVr9bD3xc/U-xe0l-XXRI/AAAAAAAAbzA/gg3GYCAfztc/s1600/4.jpg)

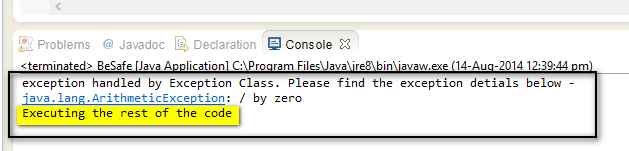
5. Save and Run the 'BeSafe.java' Class file and observe that Arithmetic Exception is thrown by the statements in try block and the program got terminated without execugting the rest of the code by displaying the error details in output as shown below -

[](https://3.bp.blogspot.com/-nBMZFRzoVaw/U-xfF8hCNRI/AAAAAAAAbzI/t7tI4E3PL38/s1600/5.jpg)

6. Now for safe side, write a catch block having Exception Class at the end of the existing catch blocks to make sure that the program wont get terminated when any unexpected exception is not handled by specific catch blocks as shown below -

[](https://1.bp.blogspot.com/-LBzkX9TiQSs/U-xgjp54iCI/AAAAAAAAbzY/vIdSG8FyoK0/s1600/6.jpg)

7. Save & Run the 'BeSafe.java' and observe that the catch block having Exception Class has handled the occurred unexpected arithmetic exception as shown below -

[](https://2.bp.blogspot.com/-sJG-DYGDTG4/U-xg7l8iTrI/AAAAAAAAbzg/mOY25T1IwaU/s1600/7.jpg)

Hence we can use, catch block having Exception Class to handle all the unexpected exceptions which are not handled by the catch blocks having specific exception classes.