Java Exception Handling examples

BY CHAITANYA SINGH | FILED UNDER: EXCEPTION HANDLING

In this tutorial, we will see examples of few frequently used exceptions. If you looking for exception handling tutorial refer this complete guide: Exception handling in Java.

Example 1: Arithmetic exception

Class: `Java.lang.ArithmeticException`
This is a built-in-class present in java.lang package. This exception occurs when an integer is divided by zero.

```
class Example1
{
   public static void main(String args[])
   {
      try{
         int num1=30, num2=0;
         int output=num1/num2;
         System.out.println ("Result: "+output);
      }
      catch(ArithmeticException e){
         System.out.println ("You Shouldn't divide a number by zero");
      }
   }
}
```

Output of above program:

```
You Shouldn't divide a number by zero
```

Explanation: In the above example I've divided an integer by a zero and because of this `ArithmeticException` is thrown.

Example 2: ArrayIndexOutOfBounds Exception

Class: `Java.lang.ArrayIndexOutOfBoundsException`
This exception occurs when you try to access the array index which does not exist. For example, If array is having only 5 elements and we are trying to display 7th element then it would throw this exception.

```
class ExceptionDemo2
{
   public static void main(String args[])
   {
      try{
        int a[]=new int[10];
        //Array has only 10 elements
        a[11] = 9;
```

```
      }
      catch(ArrayIndexOutOfBoundsException e){
         System.out.println ("ArrayIndexOutOfBounds");
      }
   }
}
```

Output:

```
ArrayIndexOutOfBounds
```

In the above example the array is initialized to store only 10 elements indexes 0 to 9. Since we are try to access element of index 11, the program is throwing this exception.

Example 3: NumberFormat Exception

Class: `Java.lang.NumberFormatException`

This exception occurs when a string is parsed to any numeric variable.

For example, the statement `int num=Integer.parseInt ("XYZ");` would throw `NumberFormatException`because `String` “XYZ” cannot be parsed to int.

```
class ExceptionDemo3
{
   public static void main(String args[])
   {
      try{
         int num=Integer.parseInt ("XYZ") ;
         System.out.println(num);
      }catch(NumberFormatException e){
          System.out.println("Number format exception occurred");
       }
   }
}
```

Output:

```
Number format exception occurred
```

Example 4: StringIndexOutOfBound Exception

Class: `Java.lang.StringIndexOutOfBoundsException`

- An object of this class gets created whenever an index is invoked of a string, which is not in the range.
- Each character of a string object is stored in a particular index starting from 0.
- To get a character present in a particular index of a string we can use a method charAt(int) of java.lang.String where int argument is the index.

E.g.

```
class ExceptionDemo4
{
   public static void main(String args[])
   {
      try{
         String str="beginnersbook";
         System.out.println(str.length());;
         char c = str.charAt(0);
         c = str.charAt(40);
         System.out.println(c);
      }catch(StringIndexOutOfBoundsException e){
          System.out.println("StringIndexOutOfBoundsException!!");
       }
   }
}
```

Output:

```
13
StringIndexOutOfBoundsException!!
```

Exception occurred because the referenced index was not present in the String.

Example 5: NullPointer Exception

Class: `Java.lang.NullPointer Exception`
An object of this class gets created whenever a member is invoked with a “null” object.

```
class Exception2
{
   public static void main(String args[])
   {
         try{
                  String str=null;
                  System.out.println (str.length());
         }
         catch(NullPointerException e){
                  System.out.println("NullPointerException..");
         }
   }
}
```

Output:

```
NullPointerException..
```

Here, length() is the function, which should be used on an object. However in the above example `String` object str is null so it is not an object due to which `NullPointerException` occurred.