

Started on	Sunday, 31 December 2023, 4:41 PM
State	Finished
Completed on	Sunday, 31 December 2023, 5:36 PM
Time taken	55 mins 20 secs

Question 1

Correct

Marked out of 50.00

Exception handling is the process of responding to the occurrence, during computation, of exceptions – anomalous or exceptional conditions requiring special processing – often changing the normal flow of program execution. (Wikipedia)

Java has built-in mechanism to handle exceptions. Using the *try* statement we can test a block of code for errors. The *catch* block contains the code that says what to do if exception occurs.

This problem will test your knowledge on try-catch block.

You will be given two integers x and y as input, you have to compute x/y . If x and y are not 32 bit signed integers or if y is zero, exception will occur and you have to report it. Read sample Input/Output to know what to report in case of exceptions.

Sample Input 0:

```
10
3
```

Sample Output 0:

```
3
```

Sample Input 1:

```
10
Hello
```

Sample Output 1:

```
java.util.InputMismatchException
```

Sample Input 2:

```
10
0
```

Sample Output 2:

```
java.lang.ArithmeticException: / by zero
```

Sample Input 3:

```
23.323
0
```

Sample Output 3:

```
java.util.InputMismatchException
```

For example:

Input	Result
10 3	3
10 Hello	java.util.InputMismatchException
10 0	java.lang.ArithmeticException: / by zero
23.323 0	java.util.InputMismatchException

Answer: (penalty regime: 0 %)

```
1 import java.util.*;
2 public class Exception1{
3     public static void main(String[] args)
4     {
5         Scanner sc = new Scanner(System.in);
6         try
7         {
8             int x = sc.nextInt();
```

```

9      int y = sc.nextInt();
10     System.out.print(x/y);
11     }
12     catch(Exception e)
13     {
14         System.out.print(e);
15     }
16 }
17 }

```

	Input	Expected	Got	
✓	10 3	3	3	✓
✓	10 Hello	java.util.InputMismatchException	java.util.InputMismatchException	✓
✓	10 0	java.lang.ArithmeticException: / by zero	java.lang.ArithmeticException: / by zero	✓
✓	23.323 0	java.util.InputMismatchException	java.util.InputMismatchException	✓

Passed all tests! ✓

Question **2**

Correct

Marked out of 50.00

Minimum eligibility to register driving license is

1. Age should be above 18 years old
2. Person should pass the road rules eligibility test(with above 80 marks)

Write a program to approve or display suitable exceptions whenever a person tries to register for a driving license.

Create two exception `InvalidAgeForDrivingLicenseException` and `InvalidMarkForDrivingLicenseException` to handle above scenarios.

Create Main class to get Name, Age and Marks from user , check the data and approve or throw suitable exception.

Note : Refer sample output for exact statement and format.

Input Format :

Name
Age
Marks obtained

Output Format :

Approved or suitable exception

Note : Sample test cases for exact exception statement

Sample testcases :

Testcase 1 Input

Guru
33
95

Testcase 1 Output

Approved

Testcase 2 Input

Guru
2
95

Testcase 2 Output

`InvalidAgeForDrivingLicenseException`: Age should be more than 18 years old

Testcase 3 Input

Guru
-3
95

Testcase 3 Output

`InvalidAgeForDrivingLicenseException`: Invalid age

Testcase 4 Input

Guru
33
75

Testcase 4 Output

InvalidMarkForDrivingLicenseException: Mark should be more than 80

Testcase 5 Input

Guru

33

-45

Testcase 5 Output

InvalidMarkForDrivingLicenseException: Invalid mark

For example:

Input	Result
Guru 33 95	Approved
Guru 2 95	InvalidAgeForDrivingLicenseException: Age should be more than 18 years old
Guru -3 95	InvalidAgeForDrivingLicenseException: Invalid age
Guru 33 75	InvalidMarkForDrivingLicenseException: Mark should be more than 80
Guru 33 -45	InvalidMarkForDrivingLicenseException: Invalid mark

Answer: (penalty regime: 0 %)

```
1 import java.util.*;
2
3 class InvalidAgeForDrivingLicenseException extends Exception{
4     InvalidAgeForDrivingLicenseException(String str)
5     {
6         super(str);
7     }
8 }
9 class InvalidMarkForDrivingLicenseException extends Exception{
10     InvalidMarkForDrivingLicenseException(String str1)
11     {
12         super(str1);
13     }
14 }
15
16 public class Exception2
17 {
18     public static void main(String[] args)
19     {
20         Scanner sc = new Scanner(System.in);
21         String name = sc.nextLine();
22         int age = sc.nextInt();
23         int marks = sc.nextInt();
24         //System.out.println(age+" "+marks);
25         try
26         {
27             if(age>18 && marks>80 && marks<=100)
28             {
29                 System.out.println("Approved");
30             }
31             else if(age<0)
32             {
33                 throw new InvalidAgeForDrivingLicenseException("Invalid age");
34             }
35             else if(age<=18)
36             {
37                 throw new InvalidAgeForDrivingLicenseException("Age should be more than 18 years old");
38             }
39         }
40     }
41 }
```

```

39         else if(marks<0 || marks>100)
40         {
41             throw new InvalidMarkForDrivingLicenseException("Invalid mark");
42         }
43         else if(marks>=0 && marks<=80)
44         {
45             throw new InvalidMarkForDrivingLicenseException("Mark should be more than 80");
46         }
47     }
48 }
49 catch(InvalidAgeForDrivingLicenseException e)
50 {
51     System.out.println(e);
52 }

```

	Input	Expected	Got	
✓	Guru 33 95	Approved	Approved	✓
✓	Guru 2 95	InvalidAgeForDrivingLicenseException: Age should be more than 18 years old	InvalidAgeForDrivingLicenseException: Age should be more than 18 years old	✓
✓	Guru -3 95	InvalidAgeForDrivingLicenseException: Invalid age	InvalidAgeForDrivingLicenseException: Invalid age	✓
✓	Guru 33 75	InvalidMarkForDrivingLicenseException: Mark should be more than 80	InvalidMarkForDrivingLicenseException: Mark should be more than 80	✓
✓	Guru 33 -45	InvalidMarkForDrivingLicenseException: Invalid mark	InvalidMarkForDrivingLicenseException: Invalid mark	✓

Passed all tests! ✓