ROLL	
NAME	

ASSIGNMENT: IV

Sl. No	Program Listing	Date
1.	Write a try block calculate the Fraction value 10 times where numerator is random number in range 1 to 10 and denominator is a random number in range 0 to 5. Write a catch block to catch the divide by zero exception may occurs in the try block. Print the origin of the exception caught.	
2.	Create an array of 10 integers and try to compute total of all the array elements from index 0 to 10. Put it in a try block and write a catch block to catch array index out of bounds exception. Print the appropriate message in the catch block.	
3.	Write a class Temperature that converts temperature between Fahrenheit and Celsius. The program should throw NumberFormatException if the degree is not legal floating-point number.	
4.	Consider the Stack class of Assignment III. Write two exception classes FullStackException and EmptyStackException. The program should throw FullStackException and EmptyStackException when required. Write try-catch blocks to handle exception.	
5.	Write a code segment that result a "NullPointerException". Write necessary catch block to catch it. Also write a 'finally block'. Write appropriate statements in it.	
6.	Create a superclass MathException and two subclasses OverFlowException and UnderFlowException from it. Write code segment that instantiate an integer variable and initialize it with some value. If the value is greater that 100, an OverFlowException is thrown, otherwise if the value is less that 0 an UnderFlowException is thrown. Now, write three catch blocks for MathException, OverFlowException and for UnderFlowException. Write appropriate message in the catch blocks and observe the result. In the first catch block re-throw the exception caught. In the other two catch blocks write appropriate statements to handle it.	
7.	Write a method validName(String name) which check if the name is valid name and return the name, throw InputMismatchException if invalid, In the main() method ask the user to enter a full name and check validity through validName() method and add try and catch block around appropriate statements. A valid name contains only alphabets and white space.	
8.	Write a class TeamPerformance, with private attributes name, gamesPlayed, gamesWon, gamesDrawn, goalsScored, goalsConceded. Write a parameterized constructor that accepts six parameter. Some validation must be done. For example, Games won cannot be more than games played. games drawn cannot be more than games played less games won. The program should throw IllegalArgumentException when the argument is invalid or inappropriate.	