S.No: 19	Exp. Name: ***Write the code to create an exception using the predefined exception***	Date: 2023-11-05

Aim:

Write a Java code to create an exception using the predefined exception

Source Code:

q223/exception2.java

```
package q223;
public class exception2
{
        public static void main(String args[])
        {
                int d,a;
                try
                {
                        d=0;
                        a=42/d;
                }
                catch(ArithmeticException e)
                {
                        System.out.println("Exception raised -Division by zero.");
                }
                System.out.println("After catch statement.");
        }
}
```

Execution Results - All test cases have succeeded!

Test Case - 1
User Output
Exception raised -Division by zero.
After catch statement.

ID: 224G1A0557

Srinivasa Ramanujan Institute of Technology

S.No: 20	Exp. Name: ***Write the code for creating your own exception***	Date: 2023-11-05

Aim:

Write a Java code for creating your own exception

Source Code:

q224/demo.java

```
package q224;
class MyException extends Exception {
        private int ex;
        MyException(int a){
                ex=a;
        }
        public String toString(){
                return "MyException["+ex+"] is less than zero";
        }
}
public class demo{
        static void sum(int a,int b)throws MyException{
                if(a<0)
                throw new MyException(a);
                else
                System.out.println(a+b);
        }
        public static void main(String args[]){
                try{
                        sum(-10,10);
                }
                catch(MyException e){
                        System.out.println(e);
                }
        }
}
```

Execution Results - All test cases have succeeded!

Test Case - 1
User Output
MyException[-10] is less than zero

S.No: 21	Exp. Name: ***program that takes inputs 5 numbers, each between 10 and 100***	**Date: 2023-11-27**

Aim:

Write java program that inputs 5 numbers, each between 10 and 100 inclusive. As each number is read display it only if it's not a duplicate of any number already read. Display the complete set of unique values input after the user enters new values

Source Code:

Duplicate.java

```
import java.util.Scanner;
class Duplicate{
	static boolean isDuplicate(int ele,int arr[]){
		for(int i=0;i<5;i++){
			if(ele == arr[i]){
				return true;
			}
		}
		return false;
	}
	public static void main(String[] args){
		Scanner inp = new Scanner(System.in);
		int num[]=new int[5];
		System.out.println("Enter 5 unique values between 10 & 100 ");
		int c=0;
		while(c<5){
			int element = inp.nextInt();
			if(element>10 && element<100){
				if(isDuplicate(element,num) == true){
					System.out.println("Duplicate value found, retry");
				}else{
					num[c]=element;
					c++;
				}
			}else{
				System.out.println("Entered value must be in between 10 &
100");
			}
		}
		System.out.print("The five unique values are :");
		for(int i=0;i<5;i++){
			System.out.print(num[i]+" ");
		}
	}
}
```

Execution Results - All test cases have succeeded!

Test Case - 1

ID: 224G1A0557
2022-2026-CSE-A
Srinivasa Ramanujan Institute of Technology

Enter 5 unique values between 10 & 100
25
15
30
0
Entered value must be in between 10 & 100
34
89
The five unique values are :25 15 30 34 89

Test Case - 2
User Output
Enter 5 unique values between 10 & 100
48
92
34
92
Duplicate value found, retry
39
23
The five unique values are :48 92 34 39 23

S.No: 22	Exp. Name: ***A program to illustrate threads***	Date: 2023-11-27

Aim:

Write Java program(s) on creating multiple threads, assigning priority to threads, synchronizing threads, suspend and resume threads

Source Code:

TestThread.java