Assignment No.6: Practice Questions

1 Handle exception in number Problem statement: Get the input String from user and parse it to integer, if it is not a number it will throw number format exception Catch it and print "Entered input is not a valid format for an integer." or else print the square of that number. (Refer Sample Input and Output). Sample input and output 1: Enter an integer: 12 The square value is 144 The work has been done successfully Sample input and output 2: Enter an integer: Java Entered input is not a valid format for an integer.

```
import java.util.*;

class Q1{


	public static void main(String[] args){

		Scanner sc = new Scanner(System.in);

		System.out.println("Enter the String: ");

		String s = sc.next();

		int squareOfNumber;

		try{

			int i = Integer.parseInt(s);

			squareOfNumber = i*i;

			System.out.println("Square of "+i+" = " +squareOfNumber);

			System.out.println("The work has been done successfully.");

		}catch(NumberFormatException e){

			e.printStackTrace();

			System.out.println("Entered input is not a valid format for an integer.");

		}


	}


}
```

2 Write a program that takes as input the size of the array and the elements in the array. The program then asks the user to enter a particular index and prints the element at that index. This program may generate Array Index Out Of Bounds Exception. Use exception handling mechanisms to handle this exception. In the catch block, print the class name of the exception thrown. Sample Input and Output 1: Enter the number of elements in the array 3 Enter the elements in the array 20 90 4 Enter the index of the array element you want to access 2 The array element at index 2 = 4 The array element successfully accessed

Sample Input and Output 2: Enter the number of elements in the array 3 Enter the elements in the array 20 90 4 Enter the index of the array element you want to access 6 java.lang.ArrayIndexOutOfBoundsException

```
import java.util.*;


class Q2{
	public static void main(String[] args){
		Scanner sc = new Scanner(System.in);
		System.out.println("Enter the size of the array:");
		int n = sc.nextInt();
		int arr[] = new int[n];
		int i;
		try{
			System.out.println("Enter the elements of an array:");
			for(i=0; i<n; i++){
				arr[i] = sc.nextInt();
			}
			System.out.println("Enter a index to access the array element:");
			int index = sc.nextInt();
			System.out.println("Element at index "+index+ " = " +arr[index]);
			System.out.println("The array element successfully accessed");

		}catch(ArrayIndexOutOfBoundsException e){
			//e.printStackTrace();
			System.out.println("java.lang.ArrayIndexOutOfBoundsException");
		}

	}

}
```

3 Write a class MathOperation which accepts integers from command line. Create an array using these parameters. Loop through the array and obtain the sum and average of all the elements. Display the result. Check for various exceptions that may arise like ArithmeticException, NumberFormatException, and so on. For example: The class would be invoked as follows: C:>java MathOperation 1900, 4560, 0, 32500

```
import java.util.*;


class MathOperation{
	public static void main(String[] args){


		int n = args.length;
		int arr[] = new int[n];
		int i, sum=0, avg=0;
		try{
			for(i=0; i<n; i++){
				arr[i] = Integer.parseInt(args[i]);
				sum = sum + arr[i];
			}
			avg = sum/n;
			System.out.println("Sum = "+sum);
			System.out.println("Average = "+avg);
		}catch(ArithmeticException e){
			System.out.println("ArithmeticException");
		}catch(NumberFormatException e1){
			System.out.println("NumberFormatException");
		}catch(ArrayIndexOutOfBoundsException e2){
			System.out.println("ArrayIndexOutOfBoundsException");
		}finally{
			System.out.println("Done.");
		}
	}
}
```

4 Write a program to accept name and age of a person from the command prompt(passed as arguments when you execute the class) and ensure that the age entered is >=18 and < 60. Display proper error messages. The program must exit gracefully after displaying the error message in case the arguments passed are not proper. (Hint : Create a user defined exception class for handling errors.)

```
class AgeException extends Exception{
		AgeException(String errorMessage){
		super(errorMessage);
	}
}


class Q4{


	public static void main(String args[]){
		String name = args[0];
		int age = Integer.parseInt(args[1]);


		try{
			if(age >= 18 && age < 60)
				System.out.println("Name: "+name+ " and Age = "+age);
			else
				throw new AgeException("Invalid age");
		}catch(AgeException e){
			System.out.println(e.getMessage());
		}


	}
}
```