Assignment-1

/*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. Index starts from zero.

This program may generate Array Index Out Of Bounds Exception or NumberFormatException . Use exception handling mechanisms to handle this exception.

Sample Input and Output 1:

```
Enter the number of elements in the array

2

Enter the elements in the array

50

80

Enter the index of the array element you want to access

1

The array element at index 1 = 80

The array element successfully accessed
```

Sample Input and Output 2:

```
Enter the number of elements in the array

2

Enter the elements in the array

50

80

Enter the index of the array element you want to access

9

java.lang.ArrayIndexOutOfBoundsException
```

Sample Input and Output 3:

```
Enter the number of elements in the array

2

Enter the elements in the array
```

```
30
j
java.lang.NumberFormatException
*/

import java.util.InputMismatchException;
import java.util.Scanner;

public class Solution {

	public static void main(String[] args) {
		// TODO Auto-generated method stub
		Scanner scan = new Scanner(System.in);

		System.out.println("Enter the number of elemets in the array");
		int len = scan.nextInt();
		int[] array = new int[len];
		System.out.println("Enter the elements in the array");
		try {
			for(int i = 0; i < array.length; i++)
				array[i] = scan.nextInt();
			System.out.println("Enter the index of the array you want to access");
			int keyIndex = scan.nextInt();
			System.out.println("The array element at index "+ keyIndex +" = "+ array[keyIndex]);
			System.out.println("The array element successfully accessed");
		}
		catch(ArrayIndexOutOfBoundsException e) {
```

```
            System.out.println(e);
        }
        catch(InputMismatchException e) {
            System.out.println("java.lang.NumberFormatException");
        }
    }

}
```

Assignment-2

```
/*Write a class MathOperation which accepts 5 integers through command line. Create an array using these parameters.

 Loop through the array and obtain the sum and average of all the elements and display the result.

Various exceptions that may arise like ArithmeticException, NumberFormatException, and so on should be handled.

*/


public class MathOperation {


    public static void main(String[] args) throws ArithmeticException,
NumberFormatException  {

        // TODO Auto-generated method stub

        if(args.length == 5) {

            int[] array = new int[args.length];

            int sum = 0;

            double avg = 0;

            try {

                for (int i = 0; i < args.length; i++)

                    array[i] = Integer.parseInt(args[i]);

                for (int i = 0; i < array.length; i++)

                    sum += array[i];

                avg = sum / array.length;

            }

            catch(Exception e){

                System.out.println(e);

            }

            System.out.println("sum: " + sum);
            System.out.println("avg: " + avg);
        }
        else {
            System.out.println("Enter 5 integers in command line");
        }
    }
}
```

Assignment-3

/*Write a Program to take care of Number Format Exception if user enters values other than integer for calculating average marks of 2 students.

The name of the students and marks in 3 subjects are taken from the user while executing the program.

In the same Program write your own Exception classes to take care of Negative values and values out of range (i.e. other than in the range of 0-100)

***/**

```
import java.util.Scanner;

class NegativeValuesException extends Exception {

    public NegativeValuesException() {

        System.out.println("NegativeValuesException occured");

    }

}


class ValuesOutOfRangeException extends Exception {

    public ValuesOutOfRangeException() {

        System.out.println("ValuesOutOfRangeException occured");

    }

}


public class Solution {


    public static void main(String[] args) {

        // TODO Auto-generated method stub

        Scanner scan = new Scanner(System.in);

        for (int i = 0; i < 2; i++) {

            String name = null;

            int subA = 0;
        int subB = 0;
        int subC = 0;
        try {
            name = scan.nextLine();
            if (scan.hasNextInt())
                subA = scan.nextInt();
            else
                throw new NumberFormatException();
            if (scan.hasNextInt())
                subB = scan.nextInt();
            else
                throw new NumberFormatException();
            if (scan.hasNextInt())
                subC = scan.nextInt();
            else
                throw new NumberFormatException();

            if (subA < 0) throw new NegativeValuesException();
            if (subA > 100) throw new ValuesOutOfRangeException();

            if (subB < 0) throw new NegativeValuesException();
            if (subB > 100) throw new ValuesOutOfRangeException();

            if (subC < 0) throw new NegativeValuesException();
            if (subC > 100) throw new ValuesOutOfRangeException();

}
catch (ArithmeticException e) {
```

```
                System.out.println(e.getMessage());
    }
    catch (NegativeValuesException e) {
                System.out.println(e.getMessage());
    }
    catch (ValuesOutOfRangeException e) {
                System.out.println(e.getMessage());
            }
            System.out.println("Name: " + name);
            System.out.println("Marks of subject A: " + subA);
            System.out.println("Marks of subject B: " + subB);
            System.out.println("Marks of subject C: " + subC);
        }
     }
}
```

Assignment-4

/*A student portal provides user to register their profile. During registration the system needs to validate the user should be located in India. If not the system should throw an exception.

Step 1: Create a user defined exception class named “InvalidCountryException”.

Step 2: Overload the respective constructors.

Step 3: Create a main class “UserRegistration”, add the following method,

void registerUser(String username,String userCountry) with the below implementation

• if userCountry is not equal to “India” throw a InvalidCountryException with the message “User Outside India cannot be registered”

• if userCountry is equal to “India”, print the message “User registration done successfully”

Invoke the method registerUser from the main method with the data specified and see how the program behaves.

Example1)

i/p:Mickey,US

o/p:InvalidCountryException should be thrown.

The message should be “User Outside India cannot be registered”

Example2)

i/p:Mini,India

o/p:User registration done successfully

***/**

```java
import java.util.Scanner;

class InvalidCountryException extends Exception {

        public InvalidCountryException() {

                System.out.println("InvalidCountryException occured");

                System.out.println("User Outside India cannot be registered");
        }
}

public class UserRegistration {

        public void registerUser(String username, String userCountry) throws InvalidCountryException {
                if (!userCountry.equals("India"))
                        throw new InvalidCountryException();
                else
                        System.out.println("User registration done successfully");
        }

        public static void main(String[] args) {
                Scanner scan = new Scanner(System.in);
                String name = "", countryName = "";
                System.out.print("Enter the name of user :");
                name = scan.nextLine();
                System.out.print("Enter country name :");
                countryName = scan.nextLine();
                UserRegistration registration = new UserRegistration();
                try {
                        registration.registerUser(name, countryName);
                }
                catch (InvalidCountryException e) {
                        System.out.println(e.getMessage());
                }
        }
```

Assignment-5

```
/*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 InvalidAgeException extends Exception {
    public InvalidAgeException() {
        System.out.println("Invalid age");
    }
}

public class Solution {
    public static void main(String[] args) throws InvalidAgeException {
        // TODO Auto-generated method stub
        String name = args[0];
        int age = Integer.parseInt(args[1]);
        if (age < 18 || age >= 60)
            throw new InvalidAgeException();
        System.out.println("Name: " + name + " Age: " + age);
    }

}
```