Department of Computer Science & Engineering, SDMCET, Dharwad-2

AOOP Assignment Submission Report

[Submitted as part of CTA Assignment No-1]

Course:	Advanced Object-Oriented Programming	Course Code:	18UCSE508
Semester:	V	Division:	B

Submitted by:

USN:	2SD20CS113	Name:	Sushmita Patil

1. Problem Definition:

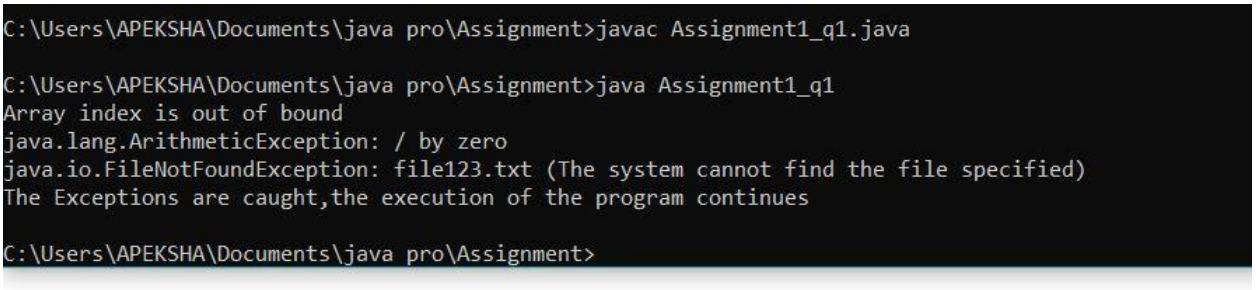
Write a Java program to generate and handle any three built-in exceptions and display appropriate error messages.

2. Java Program:

```
/*Sushmita Patil
  USN :2SD20CS113
  */
/* Exception :It is an unwanted event that occurs during the execution of the
program .The program halts if an excetion occurs .This exception can be handeled*/
import java.lang.*;
import java.io.FileInputStream;
import java.util.Scanner;
class Assignment1_q1{

        public static void main(String args[]){

                int[] number ;
int number2= 0;
number = new int[1];
number[0] = 3;

          try{
                int number4 = number[0]+number[1];
             }
              //The catch block has exception handler code
```

```
catch(ArrayIndexOutOfBoundsException excep_obj){
                System.out.println("Array index is out of bound");
            }
try{
                int number3 = number[0]/number2;
           }
             catch(ArithmeticException excep_obj){
                  System.out.println(excep_obj.toString());
             }
try{
             FileInputStream f1 = new FileInputStream("file123.txt");
             Scanner sc = new Scanner;
            }catch(FileNotFoundException ob1){
                 System.out.println(ob1);
                }
                     System.out.println("The Exceptions are caught,the execution of the program continues");
 }
}
```

3. Screen Shots of Execution:



1. Problem Definition:

Write a Java program to read an integer and check whether the number is prime or not.If negativenumber is entered, throw an exception NegativeNumberNotAllowedException and if entered number is not prime, then throw NumberNotPrimeException.

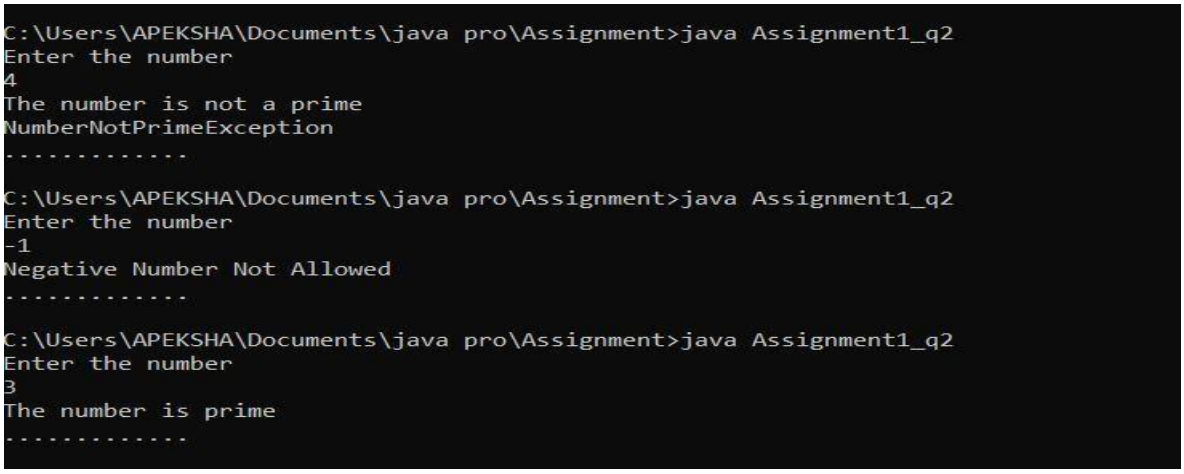
2. Java Program:

```
import java.util.Scanner;
import java.lang.*; import
java.io.*;
//creating a class of customized exception by extending exception class class
NegativeNumberNotAllowedException extends Exception{
        String str1;
         NegativeNumberNotAllowedException(String str1){
super(str1);
         }
 }
 class NumberNotPrimeException extends Exception{

       NumberNotPrimeException(){
           System.out.println("The number is not a prime");
       }
 }
class Assignment1_q2{

     public static void check(int num) throws NegativeNumberNotAllowedException{
if(num<0){
```

```
throw new NegativeNumberNotAllowedException("Negative Number Not Allowed");

            }
        }
    public static void main(String[] args){
        //Asking for the input
        System.out.println("Enter the number");

        Scanner sc = new Scanner(System.in);
int num = sc.nextInt();

        try{
            check(num);
int remainder,flag=0;
for(int i=2;i<=num/2;i++){
remainder=num%i;
if(remainder==0)
                    {
                        flag=1;
break;
                     }
                }
if(flag==1){
                    throw new NumberNotPrimeException();
            }
else {
                System.out.println("The number is prime");
            }
```

```
    }catch(NegativeNumberNotAllowedException e){
                    System.out.println(e.getMessage());
        }
      catch(NumberNotPrimeException e){
                    System.out.println(e.toString());
}
   System.out.println(".............");
   }
}
```

3. Screen Shots of Execution:



1. Problem Definition:

Write a Java program to perform the following operations: a)

Read a line of text

b) Search for a sub-string SDMCET (case insensitive search)

c) If found, then print success message

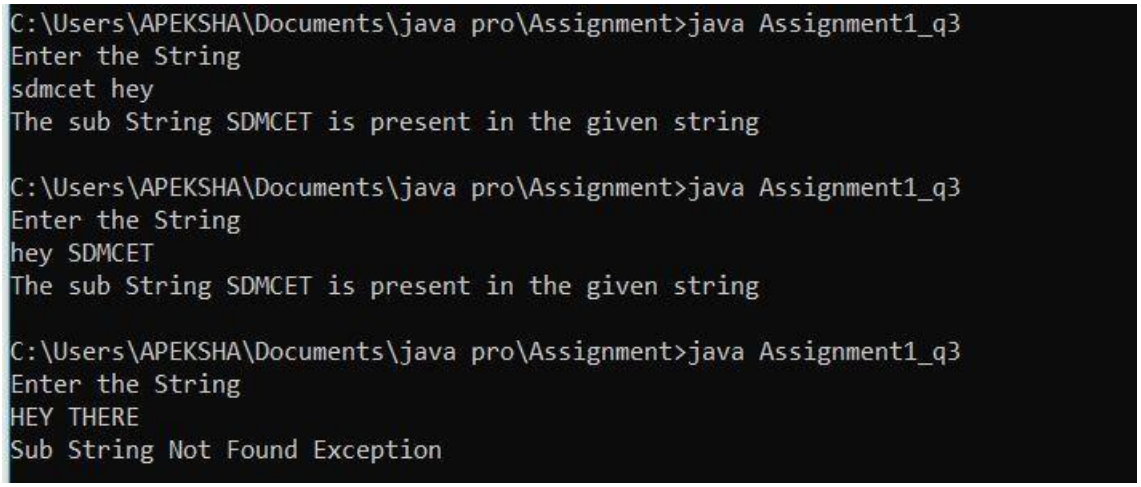
d) Otherwise throw an exception SubStringNotFoundException with appropriate message

2. Java Program:

```
import java.util.Scanner;
import java.lang.*; import
java.io.*;
class SubStringNotFoundException extends Exception{
        String str1;
         SubStringNotFoundException(String str1){
super(str1);
        }
 }
class Assignment1_q3{
 public static void main(String[] args){
          //Asking for the input
          System.out.println("Enter the String");
          Scanner sc = new Scanner(System.in);
          String str1 = sc.nextLine();
String str2 = "SDMCET";          str1
= str1.toUpperCase();          try{
                if(str1.contains(str2)){
```

```
                    System.out.println("The sub String SDMCET is present in the given string");
                }
else{
                    throw new SubStringNotFoundException("Sub String Not Found Exception");
                }
            }catch(SubStringNotFoundException e){
                    System.out.println(e.getMessage());
              }
   }
}
```

3. Screen Shots of Execution:



1. Problem Definition:

Write a Java program to perform the following operations:

a) Create a file named Alphabets.txt and insert appropriate data into it

b) Read the file and copy all the consonants into another file named Consonants.txt

c) If vowel is encountered, throw an exception VowelNotAllowedException and continue until end of file.

2. Java Program:

```
import java.io.*;



 VowelsNotAllowedException extends Exception{
      private String str;
      VowelsNotAllowedException(String str){
      this.str=str;
         }
}
 class Assignment1_q4 {
         public static void main(String[] args)throws IOException{

                  FileInputStream fin=new FileInputStream("C:\\Users\\
                  APEKSHA\\Documents\\java pro\\Alphabets.txt");
                  //reading text from alphabet.txt file
                  FileOutputStream fout=new
FileOutputStream("C:\\Users\\APEKSHA\Documents\\java pro\\Consonent.txt");
      //writing bytes to consonent.txt file


int extract;
                  while((extract=fin.read())!=-1) {
```

```
try {
        if(extract=='a'||extract=='A'||extract=='e'||extract=='E'||extract=='i'||extract=='I'||extract=='o
        '||extract=='O'||extract=='u'||extract=='U')    throw new
         VowelsNotAllowedException("Vowels Not Allowed"); //exception being
        thrown when vowels are found
                                    else
                                            fout.write(s);
                              }catch(VowelsNotAllowedException e) {
           e.printStackTrace();
                                 }
                  }
                              fin.close();
           fout.close();
         }
        }
```

3. Screen Shots of Execution: