**SDM College of Engineering and Technology**

Dhavalagiri , Dharwad-580002. Karnataka State. India.

Email: [principal@sdmcet.ac.in](mailto:principal@sdmcet.ac.in), [cse.sdmcet@gmail.com](mailto:cse.sdmcet@gmail.com)

Ph: 0836-2447465/ 2448327 Fax: 0836-2464638 Website: sdmcet.ac.in

**Department**

**of**

**COMPUTER SCIENCE AND ENGINEERING**

**ASSIGNMENT-1**

**[18UCSE508- ADVANCED OBJECT ORIENTED PROGRAMMING]**

**Course Teacher: Prof. Indira R Umarji**

****

**2022- 2023**

Submitted

By

**Vinyas LS**

**2SD20CS125**

**5th Semester B division**

Q1. Write a Java program to generate and handle any three built-in exceptions and display appropriate error messages.

**//** Java program togenerate and handle any three built-in exceptions

import java.util.Scanner;

class BuiltInFunctions {

public static void main(String args[])

{

int choice;

Scanner sc=new Scanner(System.in);

while(1>0) {

System.out.println("enter 1 --> To demonstrate Arithmetic Exceptions ");

System.out.println("enter 2 --> To demonstrate Array index out of bounds ");

System.out.println("enter 3 --> To demonstrate Null Pointer Exception\nenter 0 --> To stop\n");

choice=sc.nextInt();

switch(choice) {

case 1: try {

System.out.println("enter any number for Numerator , enter 0 number for denomenator to demonstrate an exception ");

System.out.printf("Numerator :-> ");

int a=sc.nextInt();

System.out.printf("Denominator :-> ");

int b=sc.nextInt();

int c = a / b;

System.out.println("Result = " + c);

}

catch (ArithmeticException e) {

System.out.println("Can't divide a number by 0");

} break;

case 2: try {

int a[] = new int[5];

System.out.println("Actually Array size is 5, enter some greater value to demonstrate an exception ");

int i=sc.nextInt();

a[i] = 9;

}

catch (ArrayIndexOutOfBoundsException e) {

System.out.println("Array Index is Out Of Bounds");

}break;

case 3: try {

String a = null;

System.out.println(a.charAt(0));

}

catch (NullPointerException e) {

System.out.println("NullPointerException..");

}break;

case 0: System.exit(0);

default : System.out.println("Invalid choice");

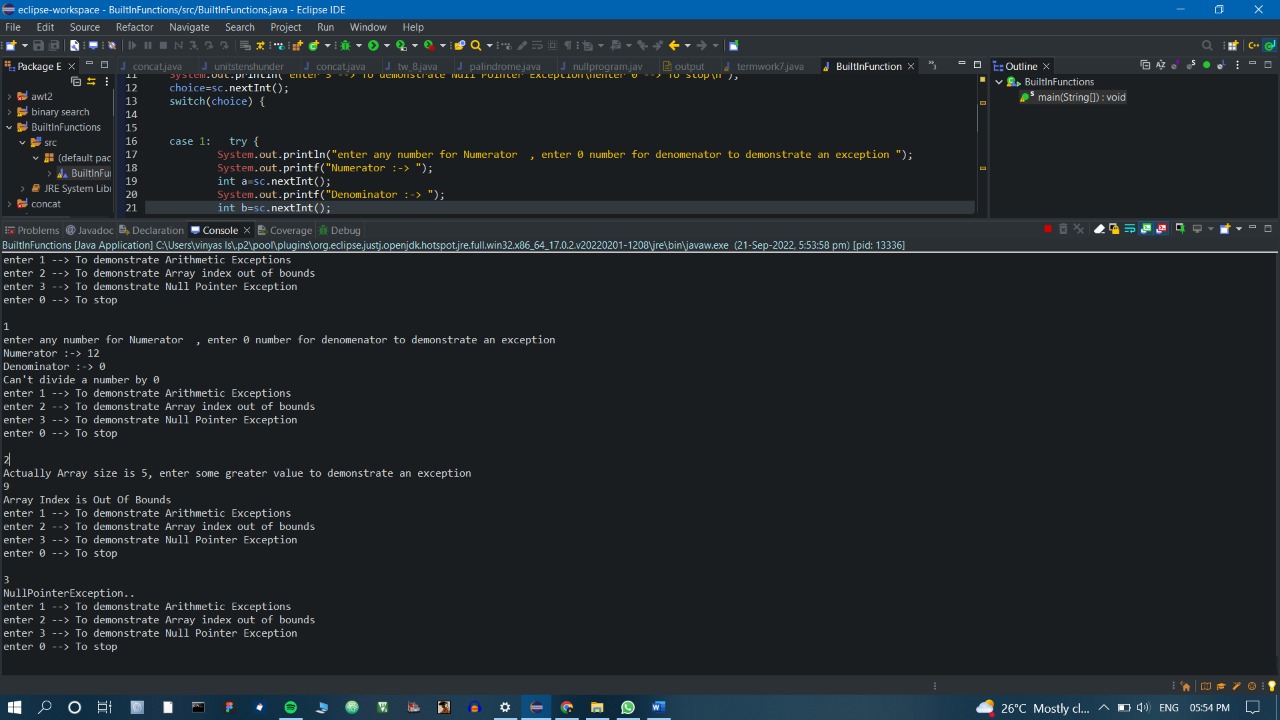
}

}

}

}

OUTPUT:



Q2. Write a Java program to read an integer and check whether the number is prime or not. If negative number is entered, throw an exception NegativeNumberNotAllowedException and if entered number is not prime, then throw NumberNotPrimeException.

import java.util.Scanner;

class NegativeNumberNotAllowedException extends Exception{

public String ToString() {

return "Negative number is not allowed";

}

}

class NotaPrimeNumber extends Exception{

public String ToString() {

return "it is not a prime";

}

}

public class ExceptionsAssignment {

public static void main(String[] args) throws Exception{

int flag1=0;

int i,m=0,flag=0;

Scanner sc=new Scanner(System.in);

int n;

System.out.println("enter the number");

n=sc.nextInt();

if(n<0) {

throw new NegativeNumberNotAllowedException() ;

}

m=n/2;

if(n==0||n==1){

flag1=1;

}else{

for(i=2;i<=m;i++){

if(n%i==0){

flag=1;

break;

}

}

if(flag==0) {

System.out.println(n+" is prime number");

}

}

if(flag1==1) {

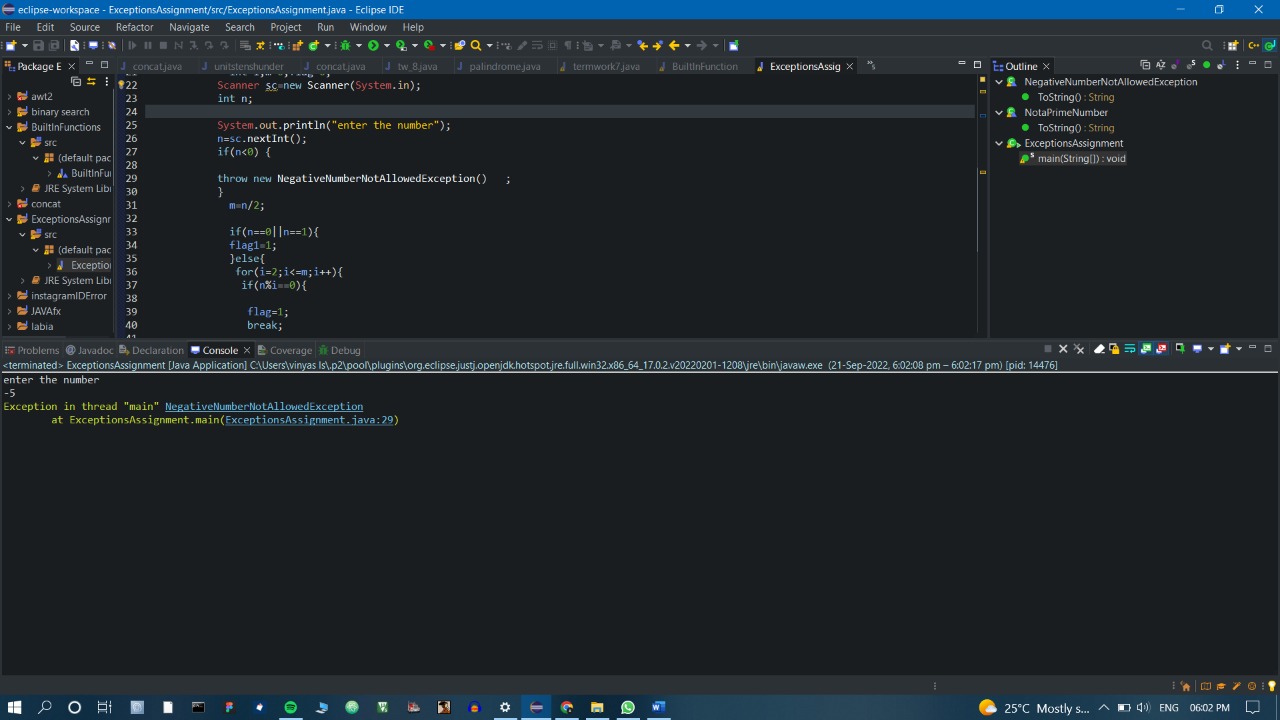
throw new NotaPrimeNumber();

}

}

}

OUTPUT:



Q3. 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

//Exception class

**public** **class** subStringNotFoundException **extends** Exception{

**public** String toString(){

**return** "sub-String Not Found";

}

}

//main class

**import** java.util.Scanner;

**class** Q3 {

**public** **static** **void** main(String[] args)**throws** Exception {

Scanner sc=**new** Scanner(System.***in***);

System.***out***.println("Enter the text");

String text=sc.next();

System.***out***.println("Enter pattern");

String pattern=sc.next();

**int** len\_t = text.length();

**int** len\_p = pattern.length();

**int** k = 0, i = 0, j = 0;

**for** (i = 0; i <= (len\_t - len\_p); i++) {

**for** (j = 0; j < len\_p; j++)

{

**if** (text.charAt(i + j) != pattern.charAt(j))

**break**;

}

**if** (j == len\_p)

{

k++;

System.***out***.println("Pattern Found");

}

}

**if** (k == 0) {

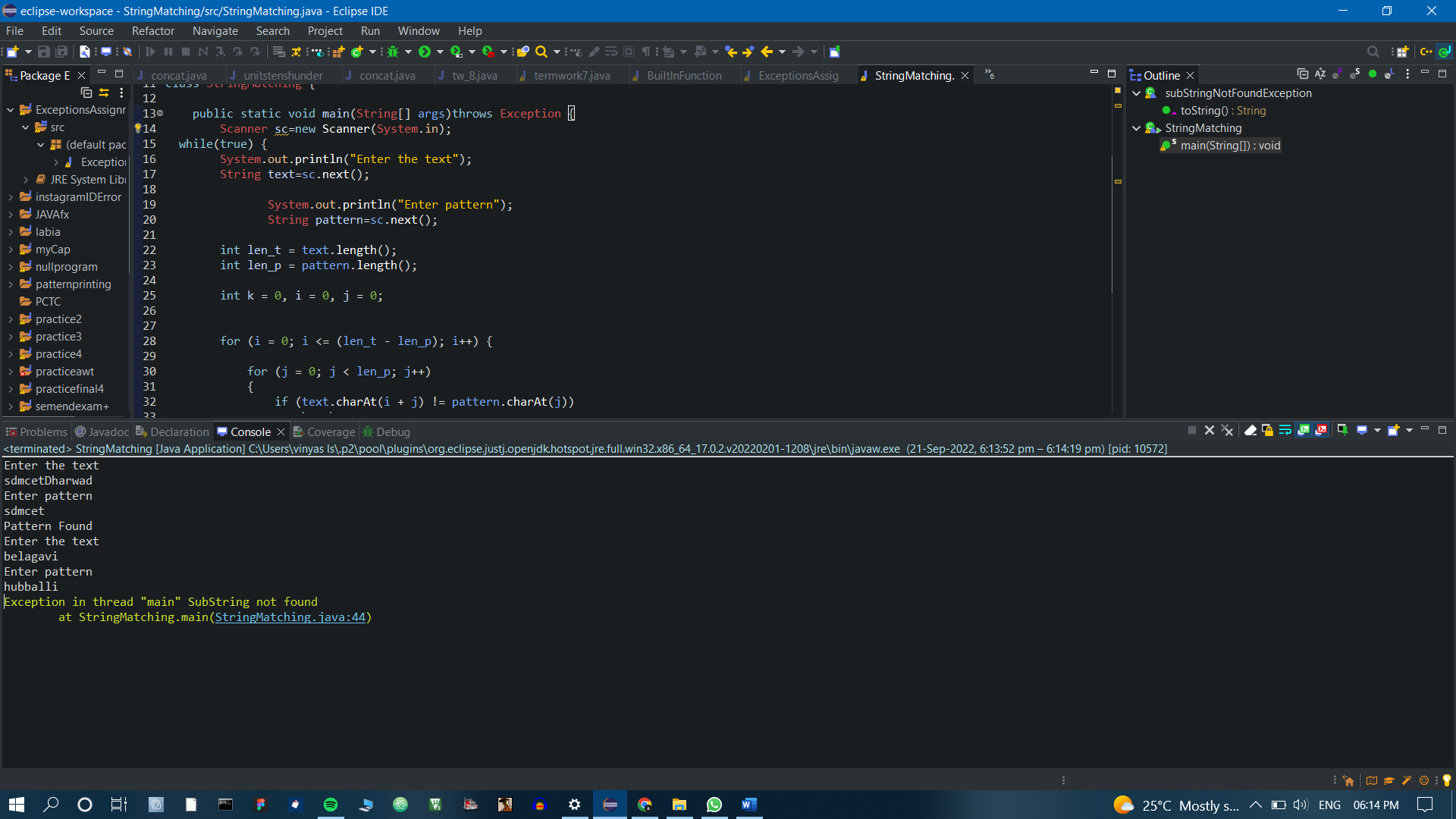
**throw** **new** subStringNotFoundException();

}

}

}

OUTPUT:



Q4. 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

import java.io.FileInputStream;

import java.io.FileOutputStream;

class VowelNotAllowed extends Exception{

public String ToString() {

return "Vowels not allowed";

}

}

public class termwork7 {

public static void main(String[] args) throws Exception{

FileInputStream fis=new FileInputStream("C:\\Users\\vinyas ls\\Desktop\\Alphabets.txt");

FileOutputStream fos=new FileOutputStream("C:\\Users\\vinyas ls\\Desktop\\Consonants.txt");

char c;

int flag=0;

while((c=(char)fis.read())!=' ') {

if(c=='a'||c=='e'||c=='i'||c=='o'||c=='u') {

throw new VowelNotAllowed();

}

else {

fos.write(c);

}}}}

Output: