

SDM College of Engineering and Technology
Dhavalagiri , Dharwad-580002. Karnataka State. India.
Email: principal@sdmcet.ac.in, 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



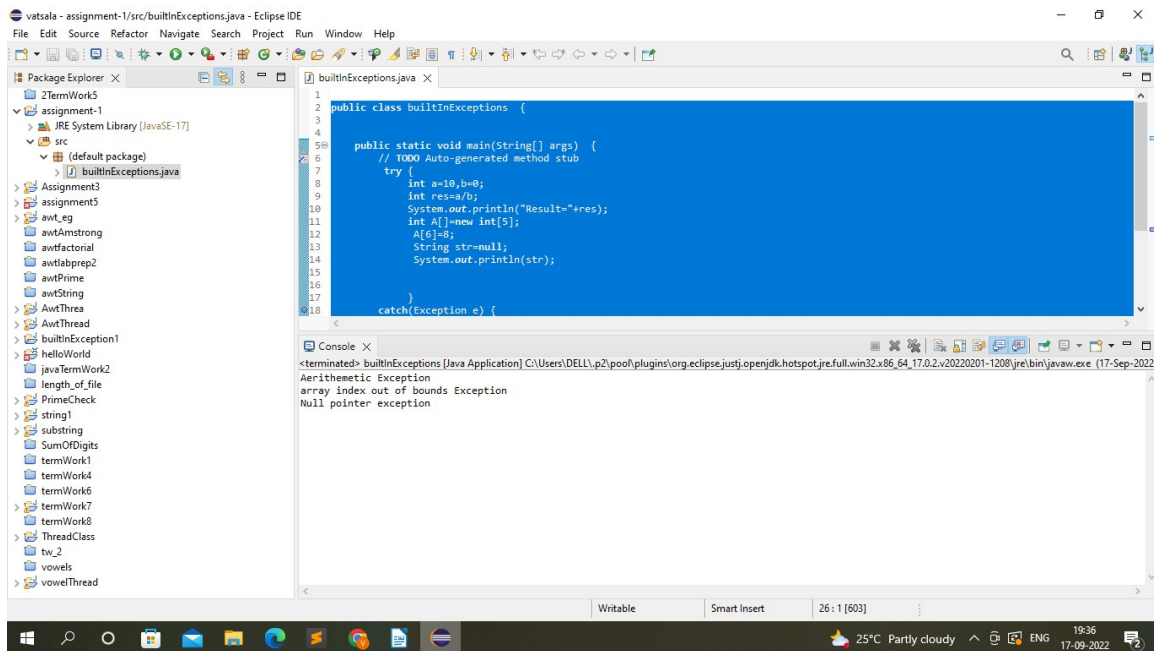
2022- 2023

Submitted
By
Ms. Vatsala P Kaashyap
2SD20CS120
5th Semester B division

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

```
public class builtInExceptions {  
  
    public static void main(String[] args) {  
        // TODO Auto-generated method stub  
        try {  
            int a=10,b=0;  
            int res=a/b;  
            System.out.println("Result="+res);  
            int A[]=new int[5];  
            A[6]=8;  
            String str=null;  
            System.out.println(str);  
  
        }  
        catch(Exception e) {  
            System.out.println("Arithmetic Exception");  
            System.out.println("array index out of bounds Exception");  
            System.out.println("Null pointer exception");  
        }  
    }  
}
```

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;
public class CheckPrime {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        System.out.println("Please enter an integer");
        Scanner sc=new Scanner(System.in);
        int num=sc.nextInt();
        try {
            if(num<0)
            {
                throw new negativeNumberException();
            }
            for(int i=2;i<num;i++)
            {
                if(num%i==0)
                {
                    throw new notPrimeException();
                }
            }

            System.out.println("Its a prime");
        }
    }
}
```

```

    }
    catch(Exception e) {
        System.out.println(e);
    }
}

class negativeNumberException extends Exception{
    public String toString() {
        return "NegativeNumberNotAllowed";
    }
}

class notPrimeException extends Exception{
    public String toString() {
        return "Not a prime number";
    }
}

```

The screenshot shows the Eclipse IDE with a project named 'PrimeCheck'. The 'src' folder contains a file 'CheckPrime.java'. The code in the editor is as follows:

```

1 import java.util.Scanner;
2 public class CheckPrime {
3
4     public static void main(String[] args) {
5         // TODO Auto-generated method stub
6         System.out.println("Please enter an integer");
7         Scanner sc=new Scanner(System.in);
8         int num=sc.nextInt();
9         try {
10             if(num<0)
11             {
12                 throw new negativeNumberException();
13             }
14             for(int i=2;i<num;i++)
15             {
16                 if(num%i==0)
17                 {
18                     throw new notPrimeException();
19                 }
20             }
21         }
22     }
23 }

```

The console output shows the following sequence of events:

```

Please enter an integer
4
Not a prime number

```

output:

Q3. Write a Java program to perform the following operations:

- Read a line of text
- Search for a sub-string SDMCET (case insensitive search)
- If found, then print success message
- Otherwise throw an exception SubStringNotFoundException with appropriate message

```

package substring;
import java.util.Scanner;
public class SDMCET {

    public static void main(String[] args) {
        // TODO Auto-generated method stub

        System.out.println("Enter the string");
        Scanner sc=new Scanner(System.in);
        String str=sc.next();
        try {
            if(str.contentEquals("sdmcet")||str.contentEquals("SDMCET"))
            {
                System.out.println("Its a valid string");
            }
            else {
                throw new NotaSubStringException();
            }
        }
        catch(Exception e){
            System.out.println(e);
        }
    }
}

class NotaSubStringException extends Exception{
    public String toString() {
        return"The string does not contain sdmcet";
    }
}

```

output:

The screenshot shows the Eclipse IDE interface. The Package Explorer on the left displays the project structure, including the 'substring' package and the 'SDMCET.java' file. The main editor window shows the source code of 'SDMCET.java', which is the same code as provided in the previous block. The Console window at the bottom shows the output of the program. It displays the prompt 'Enter the string', followed by the user input 'sdmcet is a good college', and the program's response 'Its a valid string'.

```

-terminated> SDMCET [Java Application] C:\Users\DELL\p2\pool\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_17.0.2.v20220201-1208\jre\bin\javaw.exe (17-Sep-2022, 7:44:20 p
Enter the string
sdmcet is a good college
Its a valid string

```

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.*;
import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.io.FileOutputStream;
import java.io.IOException;
public class Streams {

    public static void main(String[] args) throws Exception {
        // TODO Auto-generated method sub

        FileInputStream is=new FileInputStream("C:\\Users\\DELL\\
Desktop\\vatsala\\Assignment3\\src\\Alphabets.txt");
        FileOutputStream os=new FileOutputStream("C:\\Users\\DELL\\
Desktop\\vatsala\\Assignment3\\src\\Cnsonants.txt");

        int c;

        while((c=is.read())!=-1) {
            try {

if(c=='a' || c=='e' || c=='i' || c=='o' || c=='u' || c=='A' || c=='I' || c=='E' || c=='O' ||
c=='U')
            {
                throw new vowelNotAllowedException();
            }
                os.write(c);

            }
catch(Exception e) {
    System.out.println(e);
    }

        }

    }

    class vowelNotAllowedException extends Exception{
        public String toString() {
            return "Vowels not allowed";
        }
    }
}

```

Output:
 Alphabets.txt
 Hello
 This is vatsala

Consonants.txt

Hll

Ths s vtsl

