Rajalakshmi Engineering College

Name: Ishwarya L
Email: 241801098@rajalakshmi.edu.in
Roll no: 241801098
Phone: 7094493654
Branch: REC
Department: AI & DS - Section 5
Batch: 2028
Degree: B.E - AI & DS

Scan to verify results

2024_28_III_OOPS Using Java Lab

2028_REC_OOPS using Java_Week 8_Q1

Attempt : 1
Total Mark : 10
Marks Obtained : 10

Section 1 : Coding

1. Problem Statement

Write a program to validate the email address and display suitable exceptions if there is any mistake.

Create 3 custom exception classes as below

DotExceptionAtTheRateExceptionDomainException

A typical email address should have a " . " character, and a "@" character, and also the domain name should be valid. Valid domain names for practice be 'in', 'com', 'net', or 'biz'.

Display Invalid Dot usage, Invalid @ usage, or Invalid Domain message based on email id.

Get the email address from the user, validate the email by checking the

above-mentioned criteria, and print the validity status of the input email address.

Input Format

The first line of input contains the email to be validated.

Output Format

The output prints a Valid email address or an Invalid email address along with the suitable exception

If email ends with . or contains not exactly one . after @, it throws:

DotException: Invalid Dot usage

Invalid email address

If @ appears not exactly once, it throws:

AtTheRateException: Invalid @ usage

Invalid email address

If the part after the last dot is not among accepted domains:

DomainException: Invalid Domain

Invalid email address

If all conditions satisfied then print:

Valid email address

241801098

Refer to the sample input and output for format specifications.

Sample Test Case

Input: sample@gmail.com
Output: Valid email address

Answer

```
import java.util.*;
class DotException extends Exception
{
   DotException(String s)
   {
      super(s);
   }
}
class AtTheRateException extends Exception
{
   AtTheRateException(String s)
   {
      super(s);
   }
}
class DomainException extends Exception
{
   DomainException(String s)
   {
      super(s);
   }
}
class Main
{
   public static void checkAt(String str) throws AtTheRateException
   {
      int c=0;
      for(int i=0;i<str.length();i++)
      {
         if(str.charAt(i)=='@')
         {
```

```
            c+=1;
        }
      }
      if(c!=1)
      {
         throw new AtTheRateException("Invalid @ usage");
      }
   }
   public static void checkDot(String str) throws DotException
   {
      int d=0;
      for(int i=0;i<str.length();i++)
      {
         if(str.charAt(i)=='.')
         {
            d+=1;
         }
      }
      if(d>1 || str.charAt(str.length()-1)=='.')
      {
         throw new DotException("Invalid Dot usage");
      }
   }
   public static void checkExtension(String s) throws DomainException
   {
      int l=s.length();
      if(!((s.contains("in") || s.contains("com") || s.contains("net")||
s.contains("biz"))))
      {
         throw new DomainException("Invalid Domain");
      }
   }
   public static void main(String[] args)
   {
      Scanner sc=new Scanner(System.in);
      String s=sc.nextLine();
      int f=0;
      try
      {
            checkAt(s);
            checkDot(s);
            checkExtension(s);
```

```
            //System.out.println("Valid email address");
        }
        catch(DotException e)
        {
            f=1;
            System.out.println(e);
            System.out.println("Invalid email address");
        }
        catch(AtTheRateException e)
        {
            f=1;
            System.out.println(e);
            System.out.println("Invalid email address");
        }
        catch(DomainException e)
        {
            f=1;
            System.out.println(e);
            System.out.println("Invalid email address");
        }
        if(f==0)
        {
            System.out.println("Valid email address");
        }
    }
}
```

Status : Correct **Marks : 10/10**

241801098