Rajalakshmi Engineering College

Name: RAGHAVAN M.K
Email: 240701408@rajalakshmi.edu.in
Roll no: 240701408
Phone: 7397247776
Branch: REC
Department: CSE - Section 3
Batch: 2028
Degree: B.E - CSE

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

Refer to the sample input and output for format specifications.

Sample Test Case

Input: sample@gmail.com

Output: Valid email address

Answer

```
// You are using Java
import java.util.*;
class DotException extends Exception{
    public DotException(String message){
        super(message);
    }
}
class AtTheRateException extends Exception{
    public AtTheRateException(String message){
        super(message);
    }
}
class DomainException extends Exception{
    public DomainException(String message){
        super(message);
    }
}

class Main{
    public static void main(String[] args){
        Scanner myobj = new Scanner(System.in);
        String email = myobj.nextLine();

        try{
            validateEmail(email);
            System.out.println("Valid email address");
        }
        catch(DotException e){
            System.out.println("DotException: "+ e.getMessage());
            System.out.println("Invalid Email address");
        }
        catch(AtTheRateException e){
```

```
            System.out.println("AtTheRateException: "+ e.getMessage());
            System.out.println("Invalid Email address");
        }
        catch(DomainException e){
            System.out.println("DomainException: "+ e.getMessage());
            System.out.println("Invalid Email address");
        }
    }
    public static void validateEmail(String email)
        throws DotException, AtTheRateException, DomainException{
            int Cnt = email.length() - email.replace("@","").length();
            if(Cnt!=1){
                throw new AtTheRateException("Invalid @ usage");
            }
            if(email.startsWith(".") || email.endsWith(".")||
            email.startsWith("@") || email.endsWith("@")){
                throw new DotException("Invalid Dot usage");
            }
            String[] parts = email.split("@");
            if(parts.length!=2){
                throw new AtTheRateException("Invalid @ usage");
            }
            String domainPart = parts[1];
            if(!domainPart.contains(".")){
                throw new DotException("Invalid email address");
            }
            String domain = domainPart.substring(domainPart.lastIndexOf(".")+1);
            List<String> validDomains = Arrays.asList("in", "com", "net", "biz");
            if(!validDomains.contains(domain)){
                throw new DomainException("Invalid Domain");
            }
        }
}
```

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