

# Rajalakshmi Engineering College

Name: vignesh rajaram

Email: 241001505@rajalakshmi.edu.in

Roll no: 241001505

Phone: 7904972586

Branch: REC

Department: IT - Section 3

Batch: 2028

Degree: B.E - IT

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**

```
import java.util.Scanner;

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);
    }
}

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

        try {
            int atCount = email.length() - email.replace("@", "").length();
            if (atCount != 1 || email.startsWith "@" || email.endsWith "@" || email.contains "@@") {
                throw new AtTheRateException("Invalid @usage");
            }
        }
    }
}
```

```

        }

        int atIndex = email.indexOf('@');
        if (atIndex == -1 || atIndex == email.length() - 1) {
            throw new DotException("Invalid Dot usage");
        }
        String afterAt = email.substring(atIndex + 1);
        int dotCount = afterAt.length() - afterAt.replace(".", "").length();
        if (dotCount != 1 || email.endsWith(".") || email.startsWith(".") || email.contains("..")) {
            throw new DotException("Invalid Dot usage");
        }

        int lastDotIndex = email.lastIndexOf('.');
        String domain = email.substring(lastDotIndex + 1);
        if (!(domain.equals("in") || domain.equals("com") || domain.equals("net") || domain.equals("biz")))) {
            throw new DomainException("Invalid Domain");
        }

        System.out.println("Valid email address");

    } catch (AtTheRateException e) {
        System.out.println("AtTheRateException: " + e.getMessage());
        System.out.println("Invalid email address");
    } catch (DotException e) {
        System.out.println("DotException: " + e.getMessage());
        System.out.println("Invalid email address");
    } catch (DomainException e) {
        System.out.println("DomainException: " + e.getMessage());
        System.out.println("Invalid email address");
    }
}
}

```

**Status :** Correct

**Marks :** 10/10