**Second-Project:** Email Validation

DESCRIPTION

**Objective:**

Write a program to search a string entered by a user from the array of strings.

**Background of the problem statement:**

Writing a program in Java code to search the email ID entered by a user from an array of email IDs of employees.

**You must use the following:**

* Eclipse/IntelliJ: An IDE to code for the application.
* Java: A programming language.
* Git: To connect and push files from the local system to GitHub.
* GitHub: To store the application code and track its versions.

**Steps needed to write a program to search the email ID entered by a user:**

1. Creating a new project in Eclipse

2. Writing a program in Java code to search the email ID entered by a user from an array of email IDs of employees.

3. Executing the program and verifying the result according to the particular emailid.

4. Print the result.

5. Pushing the code to GitHub repositories.

**Step1:**

**Creating a new project in Eclipse:**

* Open Eclipse
* Go to File -> New -> Project -> Java Project -> Next.
* Type in project name as **Second-Project** and click on “Finish.”
* Select your project and go to File -> New -> Package.
* Enter **com.simplilearn.test** as Package name.
* Select your project and go to File -> New -> Class.
* Enter **EmailValidation** as class name, check the checkbox “public static void main(String[] args)”., and click on “Finish.”

A valid email address consists of an email prefix and an email domain, both in acceptable formats.

The prefix appears to the left of the @ symbol. The domain appears to the right of the @ symbol.

For example, in the address example@mail.com, "example" is the email prefix, and "mail.com" is the email domain.

In this program we create an array of email IDs of employees using Arraylist.

ArrayList<String> emailid=**new** ArrayList<String>();

We add the emailid to the Arraylist using

emailid.add("sindhu.s216@jmail.com");

By using Iterator interface we can traverse Arraylist elements

Iterator<String> retrive= emailid.iterator();

**while**(retrive.hasNext())

{

System.***out***.println(retrive.next());

}

To enter the emailid to search we use scanner. The below code is used to print the emailid to be searched.

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

System.***out***.println("Enter Emailid to search:");

String SearchElement = input.nextLine();

input.close();

The below code is used to check whether the entered emailid is in the valid format or not.

String regex = "^(.+)@(.+)$";

Matcher matcher = Pattern.*compile*(regex).matcher(SearchElement);

**if** (matcher.matches() && emailid.stream().anyMatch(mail -> mail.equals(SearchElement)))

{

System.***out***.println(" Emailid " + SearchElement + " Is Valid ");

}

**else**

{

System.***out***.println(" Emailid " + SearchElement + " Is Not Valid");

}

If the entered emailid is valid then if condition is performed.

If the entered emailid is not valid then if else is performed.

Contains() method is used to search whether the entered emailid is matching or not matching

**if**(emailid.contains(SearchElement))

{

System.***out***.println(" Emailid " + SearchElement + " Is Matching ");

}

**else**

{

System.***out***.println(" Emailid " + SearchElement +" Is Not Matching");

}

**2. Writing a program in Java code to search the email ID entered by a user from an array of email IDs of employees:**

**package** com.simplilearn.test;

**import** java.util.ArrayList;

**import** java.util.Iterator;

**import** java.util.Scanner;

**import** java.util.regex.Matcher;

**import** java.util.regex.Pattern;

**public** **class** EmailValidation {

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

System.***out***.println("ArrayList of email ID's of employees");

ArrayList<String> emailid=**new** ArrayList<String>();

emailid.add("sindhu.s216@jmail.com");

emailid.add("lekha.q354@jmail.com");

emailid.add("vishnu.h568@jmail.com");

emailid.add("varun.n389@jmail.com");

emailid.add("harika.f987@jmail.com");

emailid.add("lavanya.d456@jmail.com");

emailid.add("sagar.w765@jmail.com");

emailid.add("shekha.yahoo.com");

emailid.add("jaya#main.com");

emailid.add("krish#brand.com");

Iterator<String> retrive= emailid.iterator();

**while**(retrive.hasNext())

{

System.***out***.println(retrive.next());

}

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

System.***out***.println("Enter Emailid to search:");

String SearchElement = input.nextLine();

input.close();

String regex = "^(.+)@(.+)$";

Matcher matcher = Pattern.*compile*(regex).matcher(SearchElement);

**if** (matcher.matches() && emailid.stream().anyMatch(mail -> mail.equals(SearchElement)))

{

System.***out***.println(" Emailid " + SearchElement + " Is Valid ");

}

**else**

{

System.***out***.println(" Emailid " + SearchElement + " Is Not Valid");

}

**if**(emailid.contains(SearchElement))

{

System.***out***.println(" Emailid " + SearchElement + " Is Matching ");

}

**else**

{

System.***out***.println(" Emailid " + SearchElement +" Is Not Matching");

}

}

}

**Step 3:**

**Executing the program and verifying the result according to the particular operation:**

Before executing the program, check for syntactical corrections. If no errors are found, follow the steps mentioned below:

* [*Right click*] in the program space.
* Select *Run As Java Application.*

**Step4:**

**Print the result:**

Case1:

ArrayList of email ID's of employees

sindhu.s216@jmail.com

lekha.q354@jmail.com

vishnu.h568@jmail.com

varun.n389@jmail.com

harika.f987@jmail.com

lavanya.d456@jmail.com

sagar.w765@jmail.com

shekha.yahoo.com

jaya#main.com

krish#brand.com

Enter Emailid to search:

lekha.q354@jmail.com

Emailid lekha.q354@jmail.com Is Valid

Emailid lekha.q354@jmail.com Is Matching

Case 2:

ArrayList of email ID's of employees

sindhu.s216@jmail.com

lekha.q354@jmail.com

vishnu.h568@jmail.com

varun.n389@jmail.com

harika.f987@jmail.com

lavanya.d456@jmail.com

sagar.w765@jmail.com

shekha.yahoo.com

jaya#main.com

krish#brand.com

Enter Emailid to search:

jaya#main.com

Emailid jaya#main.com Is Not Valid

Emailid jaya#main.com Is Matching

Case 3:

ArrayList of email ID's of employees

sindhu.s216@jmail.com

lekha.q354@jmail.com

vishnu.h568@jmail.com

varun.n389@jmail.com

harika.f987@jmail.com

lavanya.d456@jmail.com

sagar.w765@jmail.com

shekha.yahoo.com

jaya#main.com

krish#brand.com

Enter Emailid to search:

krish.jmail.com

Emailid krish.jmail.com Is Not Valid

Emailid krish.jmail.com Is Not Matching

Therefore, the above code is able to execute the required solution by using Strings,Methods,Collections.

**Step5:**

**Pushing the code to your GitHub repositories:**

* Open your command prompt and navigate to the folder where you have created your files.

**cd <folder path>**

* Initialize your repository using the following command:

**git init**

* Add all the files to your git repository using the following command:

**git add .**

* Commit the changes using the following command:

**git commit . -m “Changes have been committed.”**

* Push the files to the folder you initially created using the following command:

**git push -u origin master**