Sending email from java program.

[jag.javaprogram@gmail.com](mailto:jag.javaprogram@gmail.com)

before June 2022, sending email from gmail was enabled simply by going to “Manage Account”

* “Less secure app access” to be tick (selected)

And then use your gmail address and gmail password in our java program.

After June 2022, gmail demand you to create a password for app access. And use this password in our program.

Demo:

Steps to enable app access in gmail (generate an app password)

1. Lets go to gmail, login using [jag.javaprogram@gmail.com](mailto:jag.javaprogram@gmail.com)
2. Manage Your google account
3. Security in the left side
4. Search for App Passwords
5. We generated password as <<<password>>

Steps in spring boot:

1. start.spring.io
2. create a rest api
3. dependencies:

email

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-mail</artifactId>

</dependency>

1. in application.properites

spring.mail.host=smtp.gmail.com

spring.mail.port=587

spring.mail.username=jag.javaprogram@gmail.com

spring.mail.password=<<your password>>

spring.mail.properties.mail.smtp.auth=true

spring.mail.properties.mail.smtp.starttls.enable=true

server.port=5000

1. create a service class:

package com.ust.email.service;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.mail.SimpleMailMessage;

import org.springframework.mail.javamail.JavaMailSender;

import org.springframework.stereotype.Service;

@Service

public class EmailService {

@Autowired

private JavaMailSender jms;

public void send(String to, String subject, String body)

{

SimpleMailMessage msg=new SimpleMailMessage();

msg.setTo(to);

msg.setSubject(subject);

msg.setText(body);

jms.send(msg);

}

}

1. Controller

@Autowired

private EmailService es;

@GetMapping

public String sendEmail()

{

es.send("rjagadeeswaran@yahoo.com", "hi", "THis is sent from java progtram,,,,");

return "Hello world";

}