



Project Explorer

- > wipro_crm [boot]
- > zoho_crm [boot]
 - > src/main/java
 - > com.example.zoho_crm
 - > com.example.zoho_crm.controller
 - > LeadController.java
 - > com.example.zoho_crm.entity
 - > com.example.zoho_crm.repository
 - > com.example.zoho_crm.restcontroller
 - > com.example.zoho_crm.services
 - > com.example.zoho_crm.util
 - > ZohoEmail.java
 - > ZohoEmailImpl.java
 - > src/main/resources
 - > src/test/java
 - > JRE System Library [JavaSE-14]
 - > Maven Dependencies
 - > src
 - > target
 - HELP.md
 - mvnw
 - mvnw.cmd
 - pom.xml
- > zoho_crm1 [boot]
- > zoho_crm2 [boot]

Boot Dashboard



Turn tags, projects, or working set names to match

ZohoEmail.java ZohoEmailImpl.java application.prop... zoho_crm/pom.xml LeadController.java

```

1 package com.example.zoho_crm.util;
2
3 public interface ZohoEmail {
4     public void zohoEmailSend(String to,String sub,String text);
5 }
6
    
```

Outline

- com.example.zoho_crm.util
 - > ZohoEmail
 - zohoEmailSend(String, String, Strir



Project Explorer

- wipro_crm [boot]
- zoho_crm [boot]
 - src/main/java
 - com.example.zoho_crm
 - com.example.zoho_crm.controller
 - LeadController.java
 - com.example.zoho_crm.entity
 - com.example.zoho_crm.repository
 - com.example.zoho_crm.restcontroller
 - com.example.zoho_crm.services
 - com.example.zoho_crm.util
 - ZohoEmail.java
 - ZohoEmailImpl.java
 - src/main/resources
 - src/test/java
 - JRE System Library [JavaSE-14]
 - Maven Dependencies
 - src
 - target
 - HELP.md
 - mvnw
 - mvnw.cmd
 - pom.xml
 - zoho_crm1 [boot]
 - zoho_crm2 [boot]

Boot Dashboard



Tune tags, projects, or working set names to match

ZohoEmail.java

ZohoEmailImpl.java

application.prop...

zoho_crm/pom.xml

LeadController.java

```
1 package com.example.zoho_crm.util;
2
3 import org.springframework.beans.factory.annotation.Autowired;
4 import org.springframework.mail.SimpleMailMessage;
5 import org.springframework.mail.javamail.JavaMailSender;
6 import org.springframework.stereotype.Component;
7
8 @Component
9 public class ZohoEmailImpl implements ZohoEmail {
10
11     @Autowired
12     private JavaMailSender sender;
13
14     @Override
15     public void zohoEmailSend(String to, String sub, String text) {
16         SimpleMailMessage sim=new SimpleMailMessage();
17         sim.setTo(to);
18         sim.setSubject(sub);
19         sim.setText(text);
20
21         sender.send(sim);
22     }
23
24 }
25
```

Outline

- com.example.zoho_crm.util
 - ZohoEmailImpl
 - sender : JavaMailSender
 - zohoEmailSend(String, String, Stri

Send Email in Spring Boot using Gmail SMTP - YouTube

About featured snippets Feedback



GeeksforGeeks

<https://www.geeksforgeeks.org/spring-boot-sending-e...>

Spring Boot - Sending Email via SMTP

28 Mar 2022 — **Spring Boot** provides the ability to **send emails** via SMTP using the JavaMail Library. Here we will be illustrating step-by-step guidelines to ...



Mailtrap

<https://mailtrap.io/blog/spring-send-email>

Sending Emails with Spring Boot

26 Jan 2024 — How to **send email** in Java **Spring Boot** and SMTP? · Using the @Autowired annotation, inject the JavaMailSender interface into the EmailService ...

People also ask

How do I send an email notification using Spring Boot?

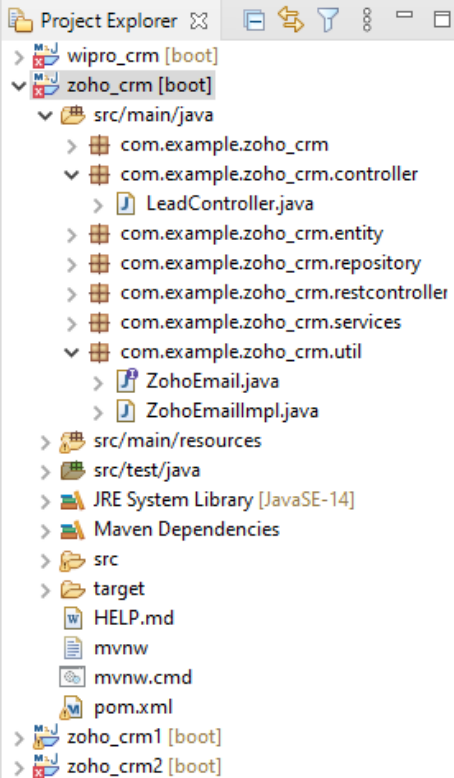


How to create email web api using Spring Boot?

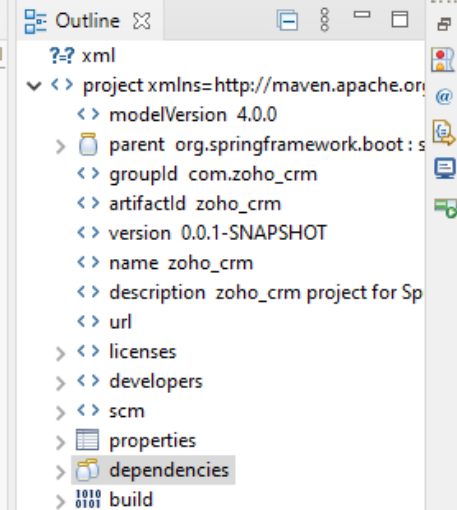


How to add mail dependency in Spring Boot?





```
55         <scope>provided</scope>
56     </dependency>
57     <!-- https://mvnrepository.com/artifact/javax.servlet/jstl -->
58 <dependency>
59     <groupId>javax.servlet</groupId>
60     <artifactId>jstl</artifactId>
61     <version>1.2</version>
62 </dependency>
63 <dependency>
64     <groupId>org.springframework.boot</groupId>
65     <artifactId>spring-boot-starter-mail</artifactId>
66 </dependency>
67
68 </dependencies>
69
70 <build>
71     <plugins>
72     <plugin>
73         <groupId>org.springframework.boot</groupId>
74         <artifactId>spring-boot-maven-plugin</artifactId>
75     </plugin>
76     </plugins>
77 </build>
78
79 </project>
80
```





Project Explorer

- wipro_crm [boot]
- zoho_crm [boot]
 - src/main/java
 - com.example.zoho_crm
 - com.example.zoho_crm.controller
 - LeadController.java
 - com.example.zoho_crm.entity
 - com.example.zoho_crm.repository
 - com.example.zoho_crm.restcontroller
 - com.example.zoho_crm.services
 - com.example.zoho_crm.util
 - ZohoEmail.java
 - ZohoEmailImpl.java
 - src/main/resources
 - src/test/java
 - JRE System Library [JavaSE-14]
 - Maven Dependencies
 - src
 - target
 - HELP.md
 - mvnw
 - mvnw.cmd
 - pom.xml
 - zoho_crm1 [boot]
 - zoho_crm2 [boot]

ZohoEmail.java

ZohoEmailImpl.java

*application.pro...

zoho_crm/pom.xml

LeadController.java

Outline

```
1 spring.application.name=zoho_crm
2 spring.datasource.url=jdbc:mysql://localhost:3306/zoho_crm
3 spring.datasource.username=root
4 spring.datasource.password=test
5
6 spring.jpa.hibernate.ddl-auto=update
7 spring.jpa.show-sql=true
8
9 spring.mvc.view.prefix=/WEB-INF/jsp/
10 spring.mvc.view.suffix=.jsp
11
12 spring.mail.host=smtp.gmail.com
13 spring.mail.port=587
14 spring.mail.username=s.indiran2003@gmail.com
15 spring.mail.password=*****
16 spring.mail.properties.mail.smtp.auth=true
17 spring.mail.properties.mail.smtp.starttls.enable=true
```

Boot Dashboard



Type tags, projects, or working set names to match

← App passwords

App passwords are less secure than using up-to-date apps and services that use modern security standards. Before you create an app password, you should check to see if your app needs this in order to sign in.
[Learn more](#)

Your app passwords

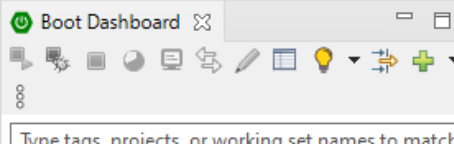
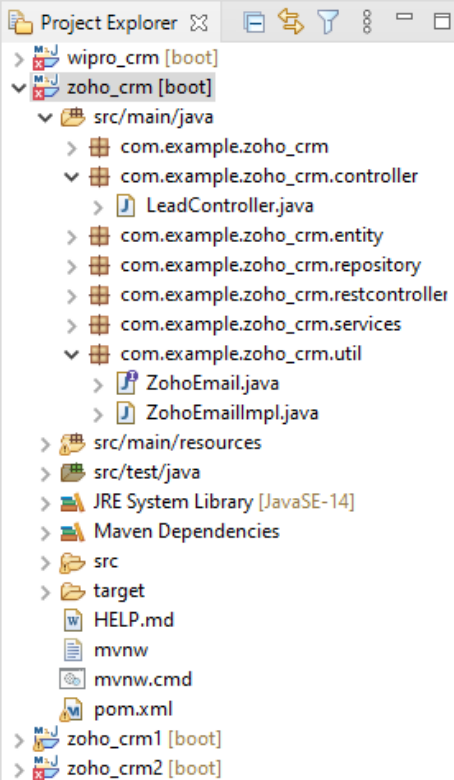
demoEmail

Created on 10:57 AM

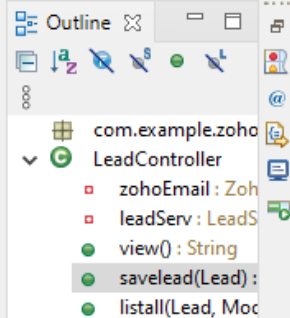


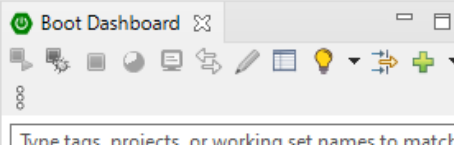
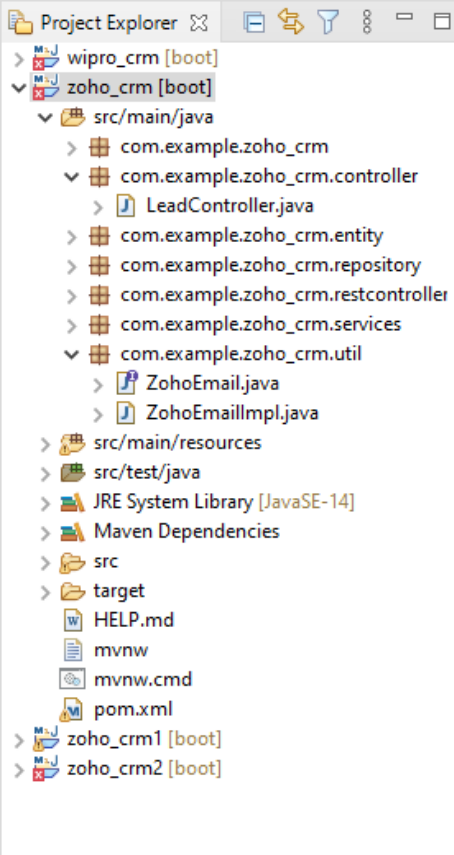
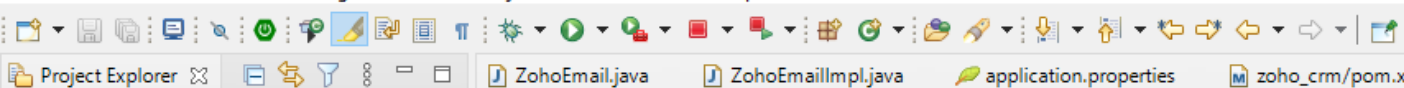
To create a new app specific password, type a name for it below...

Create

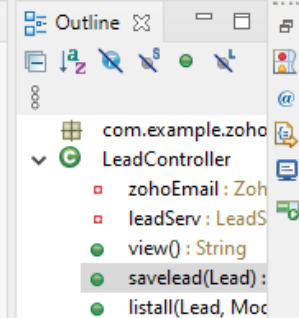


```
14 @Controller
15 public class LeadController {
16
17     @Autowired
18     private ZohoEmail zohoEmail;
19
20     @Autowired
21     private LeadService leadServ;
22
23     // http://localhost:8080/view
24     @RequestMapping("/view")
25     public String view() {
26         return "view";
27     }
28
29     //action = savelead like servlet
30     @RequestMapping("/savelead")
31     public String savelead(Lead lead) {
32         leadServ.saveLeadService(lead);
33         zohoEmail.zohoEmailSend(lead.getEmail(), "Welcome", "Thank you for car bok
34         return "view";
35     }
36
37     //http://localhost:8080/listall
38     @RequestMapping("/listall")
39     public String listall(Lead lead, Model model) {
40         List<Lead> leads = leadServ.listAllData(lead);
```





```
14 @Controller
15 public class LeadController {
16
17     @Autowired
18     private ZohoEmail zohoEmail;
19
20     @Autowired
21     private LeadService leadServ;
22
23     // http://localhost:8080/view
24     @RequestMapping("/view")
25     public String view() {
26         return "view";
27     }
28
29     //action = savelead like servlet
30     @RequestMapping("/savelead")
31     public String savelead(Lead lead) {
32         leadServ.saveLeadService(lead);
33         zohoEmail.zohoEmailSend(lead.getEmail(), "Welcome", "Thank you for car bok
34         return "view";
35     }
36
37     //http://localhost:8080/listall
38     @RequestMapping("/listall")
39     public String listall(Lead lead, Model model) {
40         List<Lead> leads = leadServ.listAllData(lead);
```



Compose

Inbox 94

Starred

Snoozed

Sent

Drafts

More

Labels +

Search mail

1 of 108

Welcome Inbox x

s.indiran2003@gmail.com
to me

Thank you for car bokking

Reply

Forward

12:10 (7 minutes ago)

31

Enable desktop notifications for Gmail. OK No, thanks X