

"인텔리제이로" Gmail 보내기

산대특 2023 4차 박연지



자바로 메일 보내기



1단계

라이브러리 설치

2단계

네이버, 구글
SMTP/IMAP환경설정

2단계

코드 작성
발신자의 메일계정과
비밀번호 (2차 인증 및
앱 비밀번호) 설정

4단계

코드 작성
발신자의 메일계정과
비밀번호 (2차 인증 및
앱 비밀번호) 설정

라이브러리 설치

Project Structure

Project Settings
Project
Modules
Libraries
Facets
Artifacts
Platform Settings
SDKs
Global Libraries
Problems

activation-1.1.1
gson-2.10.1
jackson-core-2.16.1
jakarta.activation-api
javax.mail-api-1.6.2
json-simple-1.1.1
ojdbc11-23.3.0.23.05

Name: javax.mail-api-1.6.2

+ + + -

Classes

C:\Users\admin\Downloads\javax.mail-api-1.6.2.jar

activation-1.1.1.jar
javax.mail-api-1.6.2.jar

2024-04-20 오후 7:40

JAR 파일

68KB

2024-04-20 오후 7:38

JAR 파일

215KB



환경설정 (Gmail)

설정

기본설정 라벨 받은편지함 계정 및 가져오기 필터 및 차단된 주소 전달 및 POP/IMAP 부가기능 채팅 및 Meet 고급 오프라인 테마

전달:

[자세히 알아보기](#)

전달 주소 추가

도움말: 필터를 만들면 메일 중 일부분 전달할 수도 있습니다.

POP 다운로드:

[자세히 알아보기](#)

1. 상태: 모든 메일에 대해 POP가 사용 설정되어 있습니다.

- ☐ 이미 다운로드 된 메일을 포함하여 모든 메일에 POP를 활성화 하기
- ☐ 지금부터 수신되는 메일에만 POP를 사용하기
- ☐ POP 사용 안함

2. POP로 메시지를 여는 경우 Gmail 사본을 받은편지함에 보관하기

3. 이메일 클라이언트 구성 (예: Outlook, Eudora, Netscape Mail)

[설정 방법](#)

IMAP 액세스:

(IMAP을 사용하여 다른 클라이언트에서 Gmail에 액세스)

[자세히 알아보기](#)

상태: IMAP를 사용할 수 있습니다.

- ☒ IMAP 사용
- ☐ IMAP 사용 안함

IMAP에서 메일을 삭제된 것으로 표시하는 경우:

- ☒ 자동 삭제 사용 - 서버를 즉시 업데이트(기본값)
- ☐ 자동 삭제 사용 안함 - 클라이언트가 서버를 업데이트할 때까지 대기

메일이 삭제된 것으로 표시되고 마지막으로 표시된 IMAP 폴더에서 삭제된 경우:

- ☒ 메일 보관(기본값)
- ☐ 메일을 휴지통으로 이동
- ☐ 메일을 즉시 완전삭제

폴더 크기 제한

- ☒ IMAP 폴더에서 메일 수를 제한하지 않습니다(기본값).
- ☐ 이만큼의 메일만 포함하도록 IMAP 폴더를 제한합니다. 1,000

이메일 클라이언트 구성(예: Outlook, Thunderbird, iPhone)

[설정 방법](#)

변경사항 저장

취소

POP IMAP 접근 허용




환경설정 (Gmail)

앱 비밀번호 만들고 사용하기

중요: 앱 비밀번호를 만들려면 Google 계정에 2단계 인증이 필요합니다.

2단계 인증을 사용 중이며 로그인할 때 '잘못된 비밀번호' 오류가 표시된다면 앱 비밀번호를 사용해 볼 수 있습니다.

1. Google 계정  으로 이동합니다.
2. 보안을 선택합니다.
3. 'Google에 로그인'에서 2단계 인증을 선택합니다.
4. 페이지 하단에서 앱 비밀번호를 선택합니다.
5. 앱 비밀번호를 사용할 위치를 기억하는 데 도움이 되는 이름을 입력합니다.
6. 생성을 선택합니다.
7. 앱 비밀번호를 입력하려면 화면에 표시된 안내를 따릅니다. 앱 비밀번호는 기기에서 생성되는 16자 코드입니다.
8. 완료를 선택합니다.

2단계 인증을 설정했는데 앱 비밀번호를 추가하는 옵션을 찾을 수 없다면 다음과 같은 이유 때문일 수 있습니다.

- Google 계정에 **보안 키**에만 2단계 인증이 설정되어 있습니다.
- 직장, 학교 또는 다른 조직 계정에 로그인한 상태입니다.
- Google 계정에 **고급 보호**가 설정되어 있습니다.

도움말: 일반적으로 앤 또는 기기당 한 번씩 앤 비밀번호를 입력해야 합니다.

구글 2차 인증 및 앱 비밀번호 발급

← 앱 비밀번호

앱 비밀번호를 사용하면 최신 보안 표준을 지원하지 않는 오래된 앱 및 서비스에서 Google 계정에 로그인할 수 있습니다.

앱 비밀번호는 최신 보안 표준을 사용하는 최신 앱 및 서비스보다 보안 수준이 낮습니다.
앱 비밀번호를 만들기 전에 앱에 로그인하려면 비밀번호가 필요하지 확인해야 합니다.

자세히 알아보기

앱 비밀번호

매일

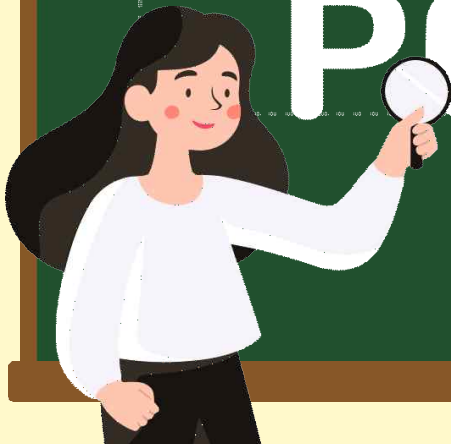
생성일: 오후 7:56

🗑

앱 전용 비밀번호를 새로 만들려면 아래에 앱 이름을 입력하세요...

앱 이름

SMTP POP/IMAP



SMTP, POP, IMAP

SMTP

Simple Mail Transfer Protocol

인터넷을 통해 이메일 메시지를 보내고 받는 데 사용되는 통신 프로토콜

POP

Post Office Protocol

응용 계층 인터넷 프로토콜 중 하나로,
원격서버로부터 TCP/IP 연결을 통해 이메일을 가져 오는데 사용된다.

IMAP

인터넷 메시지 액세스 프로토콜(IMAP)은 이메일을 받기 위한 프로토콜.
프로토콜은 동일한 하드웨어 또는 소프트웨어를 사용하는지 여부와 관계없이 서로 연결할 수 있도록 한다.

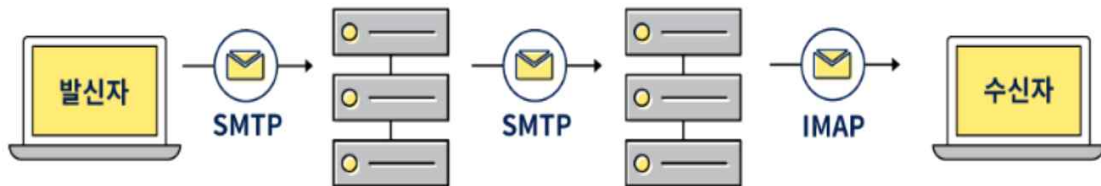


SMTP와 IMAP의 조합

SMTP



IMAP



1) 아웃룩 등의 클라이언트가
SMTP를 이용해 이메일을 서버로 보냄

2) 이메일 서버가 SMTP를 이용해
송신자 이메일 서버에 이메일을 발송

3) SMTP 통신을 통해 메일을 수신하면
수신자의 이메일 클라이언트는 IMAP을 활용해
메시지를 사서함으로 가져옴



IMAP와 POP의 차이

imap	POP3
사용자는 모든 장치에서 이메일에 액세스할 수 있습니다.	기본적으로 이메일은 다운로드된 장치에서만 액세스할 수 있습니다.
서버에 이메일이 저장됩니다. IMAP는 서버와 클라이언트 간의 중개자 역할을 합니다.	다운로드되면 달리 구성되지 않는 한 이메일이 서버에서 삭제됩니다.
이메일을 오프라인에서 액세스할 수 없습니다.	이메일을 오프라인에서 액세스할 수 있지만, 다운로드한 장치에서만 액세스할 수 있습니다.
이메일 본문은 사용자가 클릭할 때까지 다운로드되지 않지만, 제목 줄과 발신자 이름은 이메일 클라이언트에 빠르게 채워집니다.	이메일이 기본적으로 장치에 다운로드되므로 메시지를 로드하는 데 시간이 더 오래 걸릴 수 있습니다.
이메일이 서버에서 자동으로 삭제되지 않으므로 IMAP는 더 많은 서버 공간이 필요합니다.	POP3는 이메일이 서버에서 자동으로 삭제되므로 이메일 서버 저장소 공간이 절약됩니다.



코드 작성

```
public class Mail {  
    new *  
    public static void main(String[] args) {  
        String user = "pyj10370160@gmail.com"; // Gmail 계정  
        String password = [REDACTED] // 보안 앱 비밀번호 또는 Gmail 비밀번호  
  
        // SMTP 서버 정보를 설정한다.  
        Properties prop = new Properties();  
        prop.put("mail.smtp.host", "smtp.gmail.com");  
        prop.put("mail.smtp.port", "587");  
        prop.put("mail.smtp.auth", "true");  
        prop.put("mail.smtp.starttls.enable", "true");  
        prop.put("mail.smtp.ssl.protocols", "TLSv1.2");  
        prop.put("mail.smtp.ssl.ciphers", "TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256");  
    }  
}
```

발신자의 메일계정 및 비밀번호 설정

Property 객체 생성 및
SMTP 서버 정보 설정



코드 작성

```
new *  
Session session = Session.getDefaultInstance(prop, new javax.mail.Authenticator() {  
    no usages new *  
    protected PasswordAuthentication getPasswordAuthentication() {  
        return new PasswordAuthentication(user, password);  
    }  
});
```

Session 객체 생성 및
서버 정보와 사용자 정보 넣기



코드 작성

```
try {  
    MimeMessage message = new MimeMessage(session);  
    message.setFrom(new InternetAddress(user)); // 발신자 이메일 주소  
    message.addRecipient(Message.RecipientType.TO, new InternetAddress("pyj10370160@gmail.com"));  
    message.setSubject("제목을 입력하세요"); // 메일 제목  
    message.setText("내용을 입력하세요"); // 메일 내용  
  
    Transport.send(message); // 전송  
    System.out.println("메시지 전송 완료...");  
} catch (MessagingException e) {  
    e.printStackTrace();  
}
```

Message 객체 생성 및
수신자, 제목, 내용 넣기

발신자에게 보낼 정보를 담은
message를 Transport 클래스
의 send 메소드를 사용하여
메일 보내기



```
C:\Users\admin\.jdk\corretto-17.0.9\bin\java.exe *-javaagent:C:\Program Files\JetBrains\IntelliJ ID
javax.mail.MessagingException: Could not convert socket to TLS;
  nested exception is:
    javax.net.ssl.SSLHandshakeException: No appropriate protocol (protocol is disabled or cipher suit
    at com.sun.mail.smtp.SMTPTransport.startTLS(SMTPTransport.java:1907)
    at com.sun.mail.smtp.SMTPTransport.protocolConnect(SMTPTransport.java:666)
    at javax.mail.Service.connect(Service.java:317)
    at javax.mail.Service.connect(Service.java:176)
    at javax.mail.Service.connect(Service.java:125)
    at javax.mail.Transport.send0(Transport.java:194)
    at javax.mail.Transport.send(Transport.java:124)
    at yeonji.jsp_submit.MailSender.main(MailSender.java:38)
```

SSL/TLS 프로토콜이나 암호화 알고리즘이 SMTP 서버와의 호환되지 않아 발생한 오류

해결법

1. TLS 프로토콜 명시적 설정
2. Cipher Suites 설정
3. Debug 모드 활성화

```
public class MailSender {
    public static void main(String[] args) {
        String user = "pyj10370160@gmail.com"; // Gmail 계정
        String password = [REDACTED] // 보안 앱 비밀번호 또는 Gmail 비밀번호

        // SMTP 서버 정보를 설정한다.
        Properties prop = new Properties();
        prop.put("mail.smtp.host", "smtp.gmail.com");
        prop.put("mail.smtp.port", "587");
        prop.put("mail.smtp.auth", "true");
        prop.put("mail.smtp.starttls.enable", "true");

        Session session = Session.getDefaultInstance(prop, new javax.mail.Authenticator() {
            no usages
            protected PasswordAuthentication getPasswordAuthentication() {
                return new PasswordAuthentication(user, password);
            }
        });
    }
}
```

에러노트

```
public class MailSender {  
    public static void main(String[] args) {  
        String user = "pyj10370160@gmail.com"; // Gmail 계정  
        String password = [REDACTED] // 보안 앱 비밀번호 또는 Gmail 비밀번호  
  
        // SMTP 서버 정보를 설정한다.  
        Properties prop = new Properties();  
        prop.put("mail.smtp.host", "smtp.gmail.com");  
        prop.put("mail.smtp.port", "587");  
        prop.put("mail.smtp.auth", "true");  
        prop.put("mail.smtp.starttls.enable", "true");  
    }  
}
```

1. TLS 버전을 명시적으로 설정

-> TLSv1.2를 사용하도록 설정하여 코드 수정

2. 암호화 알고리즘의 문제 수정

-> 사용 가능한 Cipher Suites를 설정하여 코드 수정.

```
public class Mail {  
    new *  
    public static void main(String[] args) {  
        String user = "pyj10370160@gmail.com"; // Gmail 계정  
        String password = [REDACTED] // 보안 앱 비밀번호 또는 Gmail 비밀번호  
  
        // SMTP 서버 정보를 설정한다.  
        Properties prop = new Properties();  
        prop.put("mail.smtp.host", "smtp.gmail.com");  
        prop.put("mail.smtp.port", "587");  
        prop.put("mail.smtp.auth", "true");  
        prop.put("mail.smtp.starttls.enable", "true");  
        prop.put("mail.smtp.ssl.protocols", "TLSv1.2");  
        prop.put("mail.smtp.ssl.ciphers", "TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256")  
    }  
}
```

3.Debug 모드 활성화

JavaMail의 디버그 모드를 활성화하여
상세한 로그를 확인하여
문제가 발생하는 지점을 좀 더 정확하게 파악

```
Session session = Session.getDefaultInstance(prop, new javax.mail.Authenticator() {  
    no usages new *  
    protected PasswordAuthentication getPasswordAuthentication() {  
        return new PasswordAuthentication(user, password);  
    }  
});  
  
session.setDebug(true); // Debug 모드 활성화  
  
try {  
    MimeMessage message = new MimeMessage(session);  
    message.setFrom(new InternetAddress(user)); // 발신자 이메일 주소  
    message.addRecipient(Message.RecipientType.TO, new InternetAddress("pyj10370160@gmail.com"));  
    message.setSubject("제목을 입력하세요"); // 메일 제목  
    message.setText("내용을 입력하세요"); // 메일 내용
```

코드 실행 및 결과

□ ☆ 나

제목을 입력하세요 - 내용을 입력하세요

제목을 입력하세요 받은편지함 x



pyj10370160@gmail.com

나에게 ▾

내용을 입력하세요

오후 3:09 (0분 전) ☆ 😊 ↶ ⋮

↶ 답장

↷ 전달



```
250 2.0.0 OK 1713679779 y16-20020aa79e1000000b006e04ca18c2bsm5669845pfq.196 - smtp
QUIT
221 2.0.0 closing connection y16-20020aa79e1000000b006e04ca18c2bsm5669845pfq.196 - smtp
메시지 전송 완료...
```

Process finished with exit code 0

레퍼런스

코드 참고

- <https://ktko.tistory.com/entry/JAVA-SMTP%EC%99%80-Mail-%EB%B0%9C%EC%86%A1%ED%95%98%EA%B8%B0Google-Naver>

- <https://heodolf.tistory.com/99?category=887835>

- <https://velog.io/@terria1020/%EC%9E%90%EB%B0%94-%EB%A9%94%EC%9D%BC-%EB%B3%B4%EB%82%B4%EA%B8%B0>

- <https://timulyslog.tistory.com/6>

개념 참고

<https://totlelab.com/faq/q=YToxOntzOjEyOiJrZXI3b3JkX3R5cGUiO3M6MzoiYWxsIjt9&bmode=view&idx=9461749&t=board&category=6o7p46838>



이상 발표를
마치겠습니다!