

# Come inviare email con Java

Inviare email programmaticamente è una funzionalità comune in molte applicazioni. Con Java, questo può essere fatto facilmente utilizzando la libreria JavaMail. In questo articolo, ti guiderò attraverso i passi necessari per inviare un'email con Java.

## Prerequisiti

- Un ambiente di sviluppo Java (come IntelliJ IDEA, Eclipse, o NetBeans).
- La libreria JavaMail, che può essere aggiunta al progetto tramite un gestore di dipendenze come Maven o Gradle.

## Aggiungere JavaMail al Progetto

Per aggiungere JavaMail al tuo progetto, includi la seguente dipendenza nel tuo file `pom.xml` (se usi Maven):

```
<dependency>
  <groupId>com.sun.mail</groupId>
  <artifactId>javax.mail</artifactId>
  <version>1.6.2</version>
</dependency>
```

Se utilizzi Gradle, aggiungi questa riga al file `build.gradle`:

```
implementation 'com.sun.mail:javax.mail:1.6.2'
```

## Configurare il Codice per Inviare Email

Ecco un esempio completo di codice Java per inviare un'email tramite SMTP:

```
import java.util.Properties;
import javax.mail.*;
import javax.mail.internet.*;

public class EmailSender {
    public static void main(String[] args) {
        // Configurazione delle proprietà SMTP
        Properties properties = new Properties();
        properties.put("mail.smtp.host",
"smtp.example.com"); // Sostituisci con il tuo server
SMTP

        properties.put("mail.smtp.port", "587");
        properties.put("mail.smtp.auth", "true");
        properties.put("mail.smtp.starttls.enable",
"true");

        // Credenziali dell'account email
        String username = "tuo_email@example.com";
        String password = "tua_password";

        // Creare una sessione con autenticazione
        Session session =
Session.getInstance(properties, new Authenticator() {
            @Override
            protected PasswordAuthentication
getPasswordAuthentication() {
                return new
PasswordAuthentication(username, password);
            }
        });
    }
}
```

```
try {
    // Comporre l'email
    Message message = new
MimeMessage(session);
    message.setFrom(new
InternetAddress("tuo_email@example.com"));

message.setRecipients(Message.RecipientType.TO,
InternetAddress.parse("destinatario@example.com"));
    message.setSubject("Oggetto della tua
email");
    message.setText("Questo è il contenuto
della tua email.");

    // Inviare l'email
    Transport.send(message);

    System.out.println("Email inviata con
successo!");
} catch (MessagingException e) {
    e.printStackTrace();
}
}
```

## Come Funziona il Codice

Il codice utilizza la libreria JavaMail per configurare le proprietà SMTP, autenticarsi con l'account email e inviare il messaggio. Ecco i principali passaggi:

- Configurazione del server SMTP (host, porta, autenticazione, e TLS).
- Creazione di una sessione con `Session.getInstance()` usando un'implementazione di `Authenticator`.

- Composizione del messaggio con la classe `MimeMessage`.
- Invio del messaggio con `Transport.send()`.

## **Conclusione**

Inviare email con Java è un processo semplice una volta configurata correttamente la libreria `JavaMail`. Puoi personalizzare ulteriormente il codice per aggiungere allegati, immagini o HTML al contenuto dell'email.