



Denne guide er oprindeligt udgivet på Eksperten.dk

## Send email fra Java

**Denne artikel fortæller hvad man skal bruge og hvordan man skal kode for at sende email fra Java.**

**Simple eksempler ikke noget vildt avanceret.**

**Den forudsætter kendskab til Java.**

Skrevet den **16. Feb 2010** af **arne\_v** | kategorien **Programmering / Java** | ★★★★★

Historie:

V1.0 - 16/01/2004 - original

V1.1 - 31/01/2004 - forbedret formatering

V1.2 - 12/11/2004 - fix trunckeret linie og tilføj note til sidst om j2ee.jar

V1.3 - 16/02/2010 - smårettelser

### Indledning

Det hænder at man har brug for at ens applikation kan sende emails ud.

Og selvfølgelig kan det også gøres i Java.

Man kan sende email fra java på 2 måder:

- \* bruge JavaMail
- \* bruge en almindelig Socket connection til port 25

Jeg vil kun beskrive den første. Normalt vil man anbefale JavaMail, fordi der er indbygget en masse funktionalitet.

I det efterfølgende vil jeg antage at:

- ens SMTP server er 192.168.1.10
- ens egen email adresse er mig@etsted.dk
- modtagers email adresse er enanden@etandetsted.dk

SMTP server er den samme som man angiver i browser/email program (og for de fleste privat brugere er det en man får udleveret i forbindelse med sit internet abonnement).

For at bruge JavaMail skal man hente:

JavaMail - <http://java.sun.com/products/javamail/>

JAF (Java Activation Framework) - <http://java.sun.com/products/javabeans/glasgow/jaf.html>

Du downloader zip filerne, udpakker dem og putter jar filerne i din classpath.

Hvis du har J2EE/Java EE SDK installeret kan du også bare putte j2ee.jar/javaee.jar i classpath, fordi den har det hele.

## Eksempler

Lad os starte med et simpelt eksempel som sender en text i ASCII.

```
import java.util.*;

import javax.mail.*;
import javax.mail.internet.*;

public class Email1 {
    private final static String SMTP_SERVER = "192.168.1.10";
    private final static String FROM = "mig@etsted.dk";
    public static void send(String to, String subj, String body) {
        try {
            Properties props = System.getProperties();
            props.put("mail.smtp.host", SMTP_SERVER);
            Session session = Session.getDefaultInstance(props, null);
            MimeMessage message = new MimeMessage(session);
            message.setFrom(new InternetAddress(FROM));
            message.addRecipient(Message.RecipientType.TO, new
InternetAddress(to));
            message.setSubject(subj);
            message.setText(body);
            Transport.send(message);
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
    public static void main(String[] args) {
        send("enanden@etandetsted.dk", "Test 1", "Linie 1\nLinie 2\nLinie
3\n");
    }
}
```

Nu modificerer vi det til at sende danske bogstaver i ISO-8859-1 karakter sæt:

```
import java.util.*;

import javax.mail.*;
import javax.mail.internet.*;

public class Email2 {
    private final static String SMTP_SERVER = "192.168.1.10";
    private final static String FROM = "mig@etsted.dk";
    public static void send(String to, String subj, String body) {
        try {
            Properties props = System.getProperties();
            props.put("mail.smtp.host", SMTP_SERVER);
            Session session = Session.getDefaultInstance(props, null);
```

```

        MimeMessage message = new MimeMessage(session);
        message.setFrom(new InternetAddress(FROM));
        message.addRecipient(Message.RecipientType.TO, new
InternetAddress(to));
        message.setSubject(subj, "ISO-8859-1");
        message.setText(body, "ISO-8859-1");
        Transport.send(message);
    } catch (Exception e) {
        e.printStackTrace();
    }
}
public static void main(String[] args) {
    send("enanden@etandetsted.dk", "Test 2 med ÆØÅ", "ABC\n123\nÆØÅ\n");
}
}

```

Nu prøver vi at sende HTML:

```

import java.util.*;

import javax.mail.*;
import javax.mail.internet.*;

public class Email3 {
    private final static String SMTP_SERVER = "192.168.1.10";
    private final static String FROM = "mig@etsted.dk";
    public static void send(String to, String subj, String body) {
        try {
            Properties props = System.getProperties();
            props.put("mail.smtp.host", SMTP_SERVER);
            Session session = Session.getDefaultInstance(props, null);
            MimeMessage message = new MimeMessage(session);
            message.setFrom(new InternetAddress(FROM));
            message.addRecipient(Message.RecipientType.TO, new
InternetAddress(to));
            message.setSubject(subj, "ISO-8859-1");
            message.setContent(body, "text/html; charset=ISO-8859-1");
            Transport.send(message);
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
    public static void main(String[] args) {
        send("enanden@etandetsted.dk", "Test 3 med ÆØÅ", "<P>Linie
1</P><P>Linie 2</P><P>Linie 3 med ÆØÅ</P>");
    }
}

```

Nu prøver vi så at sende en email med både text og HTML:

```

import java.util.*;

import javax.mail.*;
import javax.mail.internet.*;

public class Email4 {
    private final static String SMTP_SERVER = "192.168.1.10";
    private final static String FROM = "mig@etsted.dk";
    public static void send(String to, String subj, String body1, String
body2) {
        try {
            Properties props = System.getProperties();
            props.put("mail.smtp.host", SMTP_SERVER);
            Session session = Session.getDefaultInstance(props, null);
            MimeMessage message = new MimeMessage(session);
            message.setFrom(new InternetAddress(FROM));
            message.addRecipient(Message.RecipientType.TO, new
InternetAddress(to));
            message.setSubject(subj, "ISO-8859-1");
            MimeBodyPart mpart1 = new MimeBodyPart();
            mpart1.setText(body1, "ISO-8859-1");
            MimeBodyPart mpart2 = new MimeBodyPart();
            mpart2.setContent(body2, "text/html; charset=ISO-8859-1");
            MimeMultipart mpart = new MimeMultipart();
            mpart.addBodyPart(mpart1);
            mpart.addBodyPart(mpart2);
            message.setContent(mpart);
            Transport.send(message);
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
    public static void main(String[] args) {
        send("enanden@etandetsted.dk", "Test 4 med ÆØÅ",
            "Linie 1\nLinie 2\nLinie 3 med ÆØÅ\n",
            "<P>Linie 1</P><P>Linie 2</P><P>Linie 3 med ÆØÅ</P>");
    }
}

```

Og til sidst vil vi sende en email med attachment:

```

import java.io.*;
import java.util.*;

import javax.activation.*;
import javax.mail.*;
import javax.mail.internet.*;

public class Email5 {
    private final static String SMTP_SERVER = "192.168.1.10";

```

```

private final static String FROM = "mig@etsted.dk";
public static void send(String to, String subj, String body, String
filename) {
    try {
        Properties props = System.getProperties();
        props.put("mail.smtp.host", SMTP_SERVER);
        Session session = Session.getDefaultInstance(props, null);
        MimeMessage message = new MimeMessage(session);
        message.setFrom(new InternetAddress(FROM));
        message.addRecipient(Message.RecipientType.TO, new
InternetAddress(to));
        message.setSubject(subj, "ISO-8859-1");
        MimeBodyPart mpart1 = new MimeBodyPart();
        mpart1.setText(body, "ISO-8859-1");
        MimeBodyPart mpart2 = new MimeBodyPart();
        mpart2.setDataHandler(new DataHandler(new
FileDataSource(filename)));
        mpart2.setFileName((new File(filename)).getName());
        MimeMultipart mpart = new MimeMultipart();
        mpart.addBodyPart(mpart1);
        mpart.addBodyPart(mpart2);
        message.setContent(mpart);
        Transport.send(message);
    } catch (Exception e) {
        e.printStackTrace();
    }
}
public static void main(String[] args) {
    send("enanden@etandetsted.dk", "Test 5",
        "Vedhæftet Z.ZIP\n", "C:\\Z.ZIP");
}
}

```

Det er faktisk ikke svært at sende email fra Java.

Bemærk at det ikke spiller nogen rolle om man sender fra en J2SE/Java SE eller J2EE/Java EE applikation. Koden er den samme.

Men en vigtig pointe er: hvis det er en J2EE app, så må man godt compile sin kode mod j2ee.jar/javaee.jar, men man må ikke bruge j2ee.jar/javaee.jar runtime med andre servere end SUN's reference implementation. Hvis javax.mail klasserne mangler så skal man deploye mail.jar og activation.jar.

**Kommentar af evj d. 21. Aug 2005 | 1**

**Kommentar af duckman d. 11. Sep 2004 | 2**

Ganske fin artikel med masser af kode eksempler som er lige til at bruge. Dog ville det være rart med lidt mere dokumentation i selv koden, og evt. en mindre forklaring om de forskellige ISO charsets

### **Kommentar af simonvalter d. 27. Jan 2004 | 3**

God artikel. Det her kan helt sikkert få dig igang.

### **Kommentar af skovborg d. 17. Feb 2004 | 4**

skide godt..  
Med venlig hilsen  
Morten skovborg

### **Kommentar af flar74 d. 23. Jul 2007 | 5**

Simpelthen god stil :-)

### **Kommentar af maximus d. 11. Sep 2004 | 6**

Der mangler dog lidt flere kommentar på hvad der sker, men omvendt, er man lidt rutineret i Java så er det ikke det store problem at gennemskue hvad der sker.  
Så alt i alt er man godt kørende med denne artikel  
Der mangler dog et lidt forklaring hvad man gør når den mail-server man benytter kræver brugernavn og adgangskode

### **Kommentar af mercutio d. 22. Jul 2004 | 7**

Kan kun sige tak for hjælpen =)

### **Kommentar af digitalsoul d. 06. May 2004 | 8**

Meget god artikel, mangler måske lidt kommentare i kode eksemplerne.

### **Kommentar af nfrank d. 15. Feb 2005 | 9**

Meget informativ og inspirerende.

### **Kommentar af kurtpedersen d. 27. Jul 2004 | 10**

Bliver ikke det fjerneste klogere.

### **Kommentar af vedikkebedre d. 20. Oct 2004 | 11**

God lille artikel der giver en god grundviden til at komme videre med email i java.

Det er korrekt at flere kommentarer vil give en hurtigere indlæring, men man lærer også ved at tænke selv.

Fortsæt endelig med sådanne artikeler der giver en basis-viden om forskellige emner.

### **Kommentar af visualdeveloper d. 01. Nov 2005 | 12**

super artikel. flot skrevet arne\_v god og informativ