**Java Mail**

* Internet Mail requires clients & servers. Clients are responsible for end user interaction (like preparing, sending, replying, forwarding & receiving mails) & server takes care of the actual mail delivery
* Java mail lets our J2EE / Java application to interact with mail servers
* Java Mail offers 2 types of services

1. Transport

It takes care of sending the email message to its recipients

1. Receive & Store

It receives the incoming message and stores it in mail boxes.

Java Mail

**SendMail**

**public** **class** SendMail

{

**public** **static** **void** main(String[] args)

{

String from = "praveen4students@gmail.com";

String[] sendTo = {"praveen.dyamappa@gmail.com", "ultimatepravee@gmail.com"};

String subject = "Your Account Got Created";

String mailContent = "<font color=\"blue\"> Your Account 1234 Got Created </font>";

Properties props = **new** Properties();

//TLS Configuration

props.put("mail.smtp.auth", "true");

props.put("mail.smtp.starttls.enable", "true");

props.put("mail.smtp.host", "smtp.gmail.com");

props.put("mail.smtp.port", "587");

//SSL Configuration

/\*props.put("mail.smtp.host", "smtp.gmail.com");

props.put("mail.smtp.socketFactory.port", "465");

props.put("mail.smtp.socketFactory.class", "javax.net.ssl.SSLSocketFactory");

props.put("mail.smtp.auth", "true");

props.put("mail.smtp.port", "465");\*/

/\*Session session = Session.getDefaultInstance(props,

new javax.mail.Authenticator() {

protected PasswordAuthentication getPasswordAuthentication() {

return new PasswordAuthentication("praveen4students@gmail.com","Sharadha#4636#");

}

});\*/

/\*Authenticator authenticate = new javax.mail.Authenticator()

{

protected PasswordAuthentication getPasswordAuthentication() {

return new PasswordAuthentication("praveen4students@gmail.com","Sharadha#4636#");

}

};

Session session = Session.getDefaultInstance(props, authenticate);\*/

Session session = Session.*getDefaultInstance*(props, **new** MyGmailAuthenticator());

**try**

{

Message message = **new** MimeMessage(session);

InternetAddress addressFrom = **new** InternetAddress(from);

InternetAddress[] addressTo = **new** InternetAddress[sendTo.length];

**for** (**int** i = 0; i < sendTo.length; i++) {

addressTo[i] = **new** InternetAddress(sendTo[i]);

}

message.setFrom(addressFrom);

message.setRecipients(Message.RecipientType.*TO*, addressTo);

//message.setRecipients(Message.RecipientType.CC, addressTo);

//message.setRecipients(Message.RecipientType.BCC, addressTo);

message.setSubject(subject);

//To Send the HTMl Mail

message.setContent(mailContent,"text/html");

//To Send the Text Mail

//message.setText(mailContent);

Transport.*send*(message);

System.*out*.println("Done");

} **catch** (MessagingException e) {

**throw** **new** RuntimeException(e);

}

}

}

**MyGmailAuthenticator**

**public** **class** MyGmailAuthenticator **extends** Authenticator

{

**protected** PasswordAuthentication getPasswordAuthentication()

{

**return** **new** PasswordAuthentication("praveen4students@gmail.com","Sharadha#4636#");

}

}

**SendMailWithAttachment**

**public** **class** SendMailWithAttachment

{

**public** **static** **void** main(String[] args)

{

String from = "praveen4students@gmail.com";

String[] sendTo = {"sj.vini@gmail.com", "rekha.ajjar@gmail.com","vinu6991@gmail.com"};

String subject = "Your Account Got Created";

String mailContent = "<font color=\"blue\"> Your Account 1234 Got Created </font>";

String fileName = "C:\\Praveen\\Baby.txt";

Properties props = **new** Properties();

//TLS Configuration

props.put("mail.smtp.auth", "true");

props.put("mail.smtp.starttls.enable", "true");

props.put("mail.smtp.host", "smtp.gmail.com");

props.put("mail.smtp.port", "587");

Session session = Session.*getDefaultInstance*(props, **new** MyGmailAuthenticator());

**try**

{

Message message = **new** MimeMessage(session);

InternetAddress addressFrom = **new** InternetAddress(from);

InternetAddress[] addressTo = **new** InternetAddress[sendTo.length];

**for** (**int** i = 0; i < sendTo.length; i++) {

addressTo[i] = **new** InternetAddress(sendTo[i]);

}

message.setFrom(addressFrom);

message.setRecipients(Message.RecipientType.*TO*, addressTo);

message.setSubject(subject);

// Set the email message text.

MimeBodyPart messagePart = **new** MimeBodyPart();

//messagePart.setContent(mailContent, "text/html");

messagePart.setText(mailContent);

// Set the email attachment file

MimeBodyPart attachmentPart = **new** MimeBodyPart();

FileDataSource fileDataSource = **new** FileDataSource(fileName) {

@Override

**public** String getContentType() {

**return** "application/octet-stream";

}

};

attachmentPart.setDataHandler(**new** DataHandler(fileDataSource));

attachmentPart.setFileName(fileDataSource.getName());

Multipart multipart = **new** MimeMultipart();

multipart.addBodyPart(messagePart);

multipart.addBodyPart(attachmentPart);

message.setContent(multipart);

Transport.*send*(message);

System.*out*.println("Done");

}**catch**(Exception ex){

ex.printStackTrace();

}//End of Try Catch Block

}//End of Main Method

}//End of Class

**SendMailUtil**

**public** **class** SendMailUtil

{

**public** **static** **boolean** sendMail(String fromAddr, String[] toAddr, String subject, String mailContent,

**boolean** isHTML, String filePathAndName)

{

Properties props = **new** Properties();

//TLS Configuration

props.put("mail.smtp.auth", "true");

props.put("mail.smtp.starttls.enable", "true");

props.put("mail.smtp.host", "smtp.gmail.com");

props.put("mail.smtp.port", "587");

Session session = Session.*getDefaultInstance*(props, **new** MyGmailAuthenticator());

**try**

{

Message message = **new** MimeMessage(session);

InternetAddress addressFrom = **new** InternetAddress(fromAddr);

InternetAddress[] addressTo = **new** InternetAddress[toAddr.length];

**for** (**int** i = 0; i < toAddr.length; i++) {

addressTo[i] = **new** InternetAddress(toAddr[i]);

}

message.setFrom(addressFrom);

message.setRecipients(Message.RecipientType.*TO*, addressTo);

message.setSubject(subject);

**if**(filePathAndName == **null**)

{

**if**(isHTML){

message.setContent(mailContent,"text/html");

}**else**{

message.setText(mailContent);

}

}**else**{

// Set the email message text.

MimeBodyPart messagePart = **new** MimeBodyPart();

**if**(isHTML){

messagePart.setContent(mailContent, "text/html");

}**else**{

messagePart.setText(mailContent);

}

// Set the email attachment file

MimeBodyPart attachmentPart = **new** MimeBodyPart();

FileDataSource fileDataSource = **new** FileDataSource(filePathAndName) {

@Override

**public** String getContentType() {

**return** "application/octet-stream";

}

};

attachmentPart.setDataHandler(**new** DataHandler(fileDataSource));

attachmentPart.setFileName(fileDataSource.getName());

Multipart multipart = **new** MimeMultipart();

multipart.addBodyPart(messagePart);

multipart.addBodyPart(attachmentPart);

message.setContent(multipart);

}

Transport.*send*(message);

**return** **true**;

}**catch**(Exception ex){

ex.printStackTrace();

**return** **false**;

}//End of Try-Catch

}//End of Send Mail

**public** **static** **void** main(String[] args)

{

String from = "praveen4students@gmail.com";

String[] sendTo = {"praveen.dyamappa@gmail.com", "ultimatepravee@gmail.com"};

String subject = "Your Account Got Created";

String mailContent = "<font color=\"yellow\"> Your Account 1234 Got Created !!!! </font>";

String fileName = "D:\\Praveen\\Apprisal.txt";

**boolean** mailSent = **false**;

//Text Email Message - Without Attachment

mailSent = *sendMail*(from, sendTo, subject, mailContent, **false**, **null**);

//HTML Email Message - Without Attachment

mailSent = *sendMail*(from, sendTo, subject, mailContent, **true**, **null**);

//Text Email Message - With Attachment

mailSent = *sendMail*(from, sendTo, subject, mailContent, **false**, fileName);

//HTML Email Message - With Attachment

mailSent = *sendMail*(from, sendTo, subject, mailContent, **true**, fileName);

**if**(mailSent){

System.*out*.println("Mail Sent !!!!");

}**else**{

System.*out*.println("Mail Didn't Sent .... Pls check the error logs");

}

}//End of Main Method

}//End of Class