

POP3 Component

Post Office Protocol v3

Java™ edition



COPYRIGHT © 2004-2005 WiseSoft Ltd. ALL RIGHTS RESERVED

1. Overview

1.1 Introduction

The POP3 (Post Office Protocol v3) component provides a simple interface for communicating with a POP3 server.

1.2 Target audience

This document is intended to serve as a technical guide to using the POP3 component. Readers should understand the Java™ programming language and be proficient in writing Java™ applications. A basic understanding of POP3 and networking protocols is also helpful.

1.3 General features

- * Ability to communicate with POP3 server
- * Ability to retrieve messages from POP3 server
- * Ability to delete messages from POP3 server
- * Ability to easily store messages to disk

2. Connecting to a POP3 server

2.1. Establishing a connection

To establish a connection to a POP3 server, create an PopClient instance and provide the POP3 server hostname, username and password.

Example

```
PopClient pop=new PopClient();
```

2.2. Sets whether to debug or not

Example

```
pop.debug=true;
```

2.3. Connects to the POP3 server

```
pop.connect(String pop3Server,String username, String password);
```

2.4. Releasing a connection

To disconnect from a POP3 server, invoke the disconnect method.

Example

```
pop.disconnect();
```

3. Retrieving POP3 email

3.1. Getting a message count

Upon establishing a connection to a POP3 server you can retrieve a count of messages in the POP3 account mailbox by getting the value of the `numberMessage`

Example

```
pop.connect(String pop3Server,String username, String password);  
int count = pop.numberMessage;
```

3.2. Getting a message header

Example

```
String header=pop.getHeader();
```

3.3. Retrieving all email messages

Upon establishing a connection to a POP3 server you can retrieve all messages in the POP3 account mailbox

Example

```
//Gets all messages  
pop.connect("pop.aba.com","username","password");  
// Iterate on number of messages  
for (int i=1;i<=pop.numberMessages;i++) {  
    // Gets a message  
    EmailMessage emailMessage=pop.retrieveMsg(i);  
}
```

3.4. Retrieving a single email message

Upon establishing a connection to a POP3 server you can retrieve a message in the POP3 account mailbox by invoking the `retrieveMsg` method

Example

```
EmailMessage em=pop.retrieveMsg(i);
```

3.5. Deleting all email messages

Upon disconnecting from a POP3 server you can optionally delete any messages retrieved from the POP3 account mailbox during your session by invoking the deleteMsg method

Example

```
//Deletes all messages
pop.connect("pop.aba.com","username","password");
// Iterate on number of messages
for (int i=1;i<=pop.numberMessages;i++) {
    pop.deleteMsg(i);
}
```

3.6. Getting the email message subject

Upon retrieving an email message from a POP3 mailbox you can get the email subject by getting the value of the subject

Example

```
EmailMessage em=pop.retrieveMsg(i);
String emailSubject=em.subject;
```

3.7. Getting the email message body

```
EmailMessage em=pop.retrieveMsg(i);
MsgPart mp=em.getPart(1);
String body=mp.getData();
```

3.8. Setting the received message stored file

Example

```
pop. msgFile="msg.txt";
```