

Documentation Home > The Java EE 5 Tutorial > Part III Web Services > Chapter 19 SOAP with Attachments API for Java > Code Examples > Attachments Example

## The Java EE 5 Tutorial

**Previous:** DOM and DOMSource Examples

**Next:** SOAP Fault Example

### Attachments Example

The example `Attachments.java`, based on the code fragments in the sections [Creating an `AttachmentPart` Object](#) and [Adding Content and Accessing an `AttachmentPart` Object](#), creates a message that has a text attachment and an image attachment. It then retrieves the contents of the attachments and prints the contents of the text attachment. You will find the code for the `Attachments` class in the following directory:

```
tut-install/javaee5tutorial/examples/saaj/attachments/src/
```

`Attachments` first creates a message in the usual way. It then creates an `AttachmentPart` for the text attachment:

```
AttachmentPart attachment1 = message.createAttachmentPart();
```

After it reads input from a file into a string named `stringContent`, it sets the content of the attachment to the value of the string and the type to `text/plain` and also sets a content ID.

```
attachment1.setContent(stringContent, "text/plain");
attachment1.setContentId("attached_text");
```

It then adds the attachment to the message:

```
message.addAttachmentPart(attachment1);
```

The example uses a `javax.activation.DataHandler` object to hold a reference to the graphic that constitutes the second attachment. It creates this attachment using the form of the `createAttachmentPart` method that takes a `DataHandler` argument.

```
// Create attachment part for image
URL url = new URL("file:///../xml-pic.jpg");
DataHandler dataHandler = new DataHandler(url);
AttachmentPart attachment2 = message.createAttachmentPart(dataHandler);
attachment2.setContentId("attached_image");
```

```
message.addAttachmentPart(attachment2);
```

The example then retrieves the attachments from the message. It displays the `contentId` and `contentType` attributes of each attachment and the contents of the text attachment.

### Building and Running the Attachments Example

The `Attachments` class takes a text file as an argument. You can specify any text file. The `attachments` directory contains a file named `addr.txt` that you can use.

To build the program using NetBeans IDE, follow these steps:

1. In NetBeans IDE, choose Open Project from the File menu.
  2. In the Open Project dialog, navigate to `tut-install/javaee5tutorial/examples/saaj/`.
  3. Select the `attachments` folder.
  4. Select the Open as Main Project check box.
  5. Click Open Project.
- A Reference Problems dialog appears. Click Close.
6. Right-click the `attachments` project and choose Resolve Reference Problems.
  7. In the Resolve Reference Problems dialog, select the first of the missing JAR files and click Resolve.

The missing files are `activation.jar`, `javaee.jar`, and `appserv-ws.jar`.

8. Navigate to the `as-install/lib/` directory.
9. Select the missing JAR file ( `activation.jar` , for example) and click Open.

In the Resolve Reference Problems dialog, all the files have green check marks to the left of their names.

10. Click Close.
11. Right-click the project and choose Build.

To run the program using NetBeans IDE, follow these steps:

1. Right-click the `attachments` project and choose Properties.
2. Select Run from the Categories tree.
3. In the Arguments field, type the name of a text file:

```
addr.txt
```

4. Click OK.
5. Right-click the project and choose Run.

To run `Attachments` using Ant, go to the directory `tut-install/javaee5tutorial/examples/saaj/attachments/` . Use the following command:

```
ant run-att -Dfile=path-name
```

Specify a text file as the *path-name* argument:

```
ant run-att -Dfile=addr.txt
```

When you run `Attachments` using this file, you will see output like the following:

```
Running Attachments.
Attachment attached_text has content type text/plain
Attachment contains:
Update address for Sunny Skies, Inc., to
10 Upbeat Street
Pleasant Grove, CA 95439
USA

Attachment attached_image has content type image/jpeg
```

**Previous:** DOM and DOMSource Examples

**Next:** SOAP Fault Example

© 2010, Oracle Corporation and/or its affiliates