

[Home](#)[JBoss AS](#)[BPM](#)[Projects](#)[Web](#)[SOA-Cloud](#)[Eclipse](#)[Java EE](#)[Cool Stuff](#)

## WHAT IS JBOSS ?



Beginners can start learning in this section !

## DEVELOP APPS



Start developing Java EE applications with JBoss AS

## JBOSS BOOKS



Learn all about the application server

## CONTACT US



Want to write for us? You would like to advertise here?

## FOLLOW US



Follow us on Twitter! @mastertheboss

## Primefaces file upload example

**FuseSource**  
Integration everywhere

**\$0 Open Source Workshops On-Demand**

- ✓ Expert Instruction
- ✓ Attend Anytime
- ✓ Attend From Anywhere
- ✓ Did We Say **FREE?**

### Help W/ Java Programming?

[NewRelic.com/Java](http://NewRelic.com/Java)

Troubleshoot & Tune Java Apps W/ New Relic Pro. Download Free Trial!



AdChoices



The PrimeFaces **File upload** component can be used to perform server side file uploads. Let's see in this quick tutorial how to achieve it in some simple steps.

The first thing you need is adding some libraries to your application. As a matter of fact, PrimeFaces file upload relies on Apache **commons-file-upload** and **commons-io** libraries. So download them and add them to your WEB-INF/lib path:

[Here](#) you can download Apache Commons io

[Here](#) you can download Apache Commons File Upload

As far as **web.xml** configuration is concerned you need to declare the

PrimeFaces filter in order to enable file uploading

1. <filter>



Register Now!

search...

Find us on Facebook

Sign Up

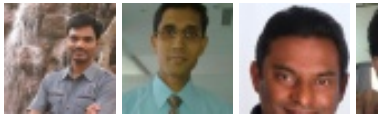
Create an account



Mastertheboss



409 people like Mastertheboss



Facebook social plugin



```
2. <filter-name>PrimeFaces FileUpload Filter</filter-name>
3. <filter-class>org.primefaces.webapp.filter.FileUploadFilter</filter-class>
4. </filter>
5. <filter-mapping>
6. <filter-name>PrimeFaces FileUpload Filter</filter-name>
7. <servlet-name>Faces Servlet</servlet-name>
8. </filter-mapping>
```

Good, your configuration is completed. Let's see a concrete example:

```
01. <html xmlns="http://www.w3.org/1999/xhtml"
02.     xmlns:h="http://java.sun.com/jsf/html"
03.     xmlns:f="http://java.sun.com/jsf/core"
04.     xmlns:p="http://primefaces.org/ui">
05.
06.     <h:head></h:head>
07.
08.     <h:body>
09.         <h:form id="form" >
10.
11.             <p:fileUpload fileUploadListener="#{fileUploadController.upload}"
12.                 6.allowTypes="/(\\.|\\/) (gif|jpe?g|png) $/" sizeLimit="100000" description="Select Images"/>
13.
14.         </h:form>
15.     </h:body>
16. </html>
```

The tag does the job of sending the multipart stream to the server. In the bean declared (fileUploadController) a listener will fire which will let you handle the file and do what you want with it.

Here's the bean implementation:

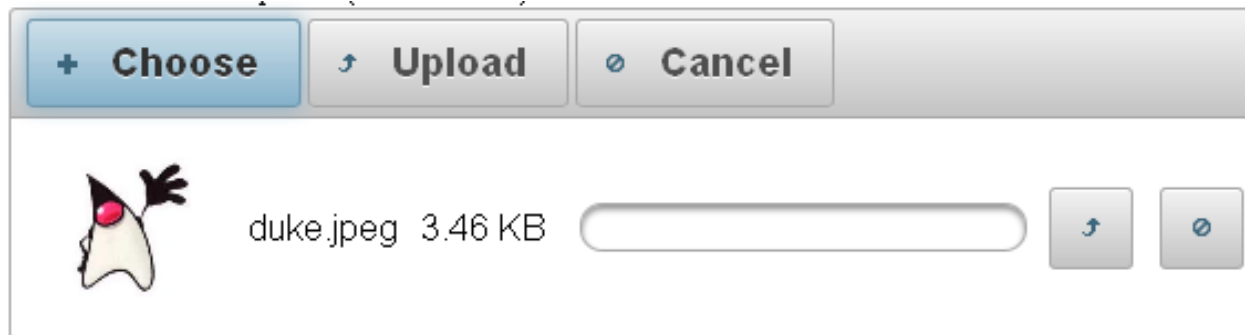
```
01. package com.sample.bean;
02.
03. import java.io.File;
04. import java.io.FileOutputStream;
05. import java.io.IOException;
06. import java.io.InputStream;
07. import java.io.OutputStream;
08.
09. import javax.faces.application.FacesMessage;
10. import javax.faces.bean.ManagedBean;
11. import javax.faces.context.FacesContext;
```

```

12.
13. import org.primefaces.event.FileUploadEvent;
14.
15. @ManagedBean(name="fileUploadController")
16. public class FileUploadController {
17.     private String destination="D:\\tmp\\";
18.
19.     public void upload(FileUploadEvent event) {
20.         FacesMessage msg = new FacesMessage("Success! ", event.getFile().getFileName() +
21.         " is uploaded.");
22.         FacesContext.getCurrentInstance().addMessage(null, msg);
23.         // Do what you want with the file
24.         try {
25.             copyFile(event.getFile().getFileName(), event.getFile().getInputStream());
26.         } catch (IOException e) {
27.             e.printStackTrace();
28.         }
29.     }
30.
31.     public void copyFile(String fileName, InputStream in) {
32.         try {
33.
34.
35.             // write the inputStream to a FileOutputStream
36.             OutputStream out = new FileOutputStream(new File(destination +
37.             fileName));
38.
39.             int read = 0;
40.             byte[] bytes = new byte[1024];
41.
42.             while ((read = in.read(bytes)) != -1) {
43.                 out.write(bytes, 0, read);
44.             }
45.
46.             in.close();
47.             out.flush();
48.             out.close();
49.
50.             System.out.println("New file created!");
51.         } catch (IOException e) {
52.             System.out.println(e.getMessage());
53.         }
54.     }

```

The upload method provides as parameter the `FileUploadEvent` from which you can retrieve the temporary file uploaded on the server. Once you have got it, use the stream to copy it or add it to a relational database.



As additional note, you can choose the path where the file is uploaded on the server by setting some additional parameters like `uploadDirectory` and `thresholdSize`

**thresholdSize** specifies the maximum file size in bytes to keep uploaded files in memory. If it exceeds this limit, it'll be temporarily written to disk.

**uploadDirectory** is the folder where to keep temporary files that exceed `thresholdSize`.

```
01. <filter>
02.   <filter-name>PrimeFaces FileUpload Filter</filter-name>
03.   <filter-class>org.primefaces.webapp.filter.FileUploadFilter</filter-class>
04.   <init-param>
05.     <param-name>thresholdSize</param-name>
06.     <param-value>51200</param-value>
07.   </init-param>
08.   <init-param>
09.     <param-name>uploadDirectory</param-name>
10.     <param-value>C:\etc</param-value>
11.   </init-param>
12. </filter>
```

You should however not rely on these attributes to define a stable storage of uploaded file. Always use IO Streams to collect the file and then store them in your favourite storage location<

**Weblogic J2ee**

[www.manageengine.com](http://www.manageengine.com)





Share



+1

4



Tweet

0

[Share](#)

Like

15



Send



Add a comment...

**Shikha Baid** · Works at APT Classes

where are the uploaded files stored?  
please help me.

[Reply](#) · [Like](#) · December 18, 2012 at 8:11am**Aulia Yayan** · 20 years old

can you help me. I'm already do it this step by step much time, but my pro  
tutorial about fileupload in jsf+ primefaces, Please help me... :'(

[Reply](#) · [Like](#) · December 10, 2012 at 9:28am**Rezano Poltak Simanjuntak** · University of Indonesia

wuidihhhhh, udah pake Jboss

[Reply](#) · [Like](#) · December 10, 2012 at 9:07pm**Anil Kurmi** · S.p.jain gurukul khurai

nice tuts

[Reply](#) · [Like](#) · November 20, 2012 at 9:14am**Vanderson Silva** · ★ Top Commenter · Programador at Giorgi & Santos

Pretty good article, congratulations! It helped me a lot!

Thank you!

[Reply](#) ·  2 · [Like](#) · November 10, 2012 at 11:26am**Akram Hamoudi** · Works at Dtmode

thank you

[Reply](#) · [Like](#) · December 25, 2012 at 4:45pm

Facebook social plugin

Mastertheboss.com Unless stated, all the contents of this site is licensed under Creative Commons License



Copyright © 2013 JBoss application server tutorials. All Rights Reserved. Designed by [JoomlaArt.com](#).  
[Joomla!](#) is Free Software released under the [GNU/GPL License](#).