

# Pengantar: Directive

**Chocolove Mic**

[chocolove\\_mic@yahoo.co.uk](mailto:chocolove_mic@yahoo.co.uk)

<http://www.mycgiserver.com/~chocolove2003>

## ***Lisensi Dokumen:***

*Copyright © 2003 IlmuKomputer.Com*

*Seluruh dokumen di IlmuKomputer.Com dapat digunakan, dimodifikasi dan disebarkan secara bebas untuk tujuan bukan komersial (nonprofit), dengan syarat tidak menghapus atau merubah atribut penulis dan pernyataan copyright yang disertakan dalam setiap dokumen. Tidak diperbolehkan melakukan penulisan ulang, kecuali mendapatkan ijin terlebih dahulu dari IlmuKomputer.Com.*

Directive adalah media yang digunakan JSP untuk mengirimkan “pesan” ke JSP container. Directive berguna untuk melakukan setting nilai global seperti deklarasi class atau method. Setting yang dilakukan oleh directive berlaku pada seluruh halaman (hanya halaman itu saja).

Sebagai gambaran, pada source code Java sering kita lihat bari berikut :

```
import java.io.*;
import java.sql.*;
```

Pada halaman JSP, programmer masih bisa melakukan hal tersebut tapi tentu saja dengan cara yang berbeda. Cara tersebut adalah dengan menggunakan directive.

## **Sintaks Directive**

Secara umum sintaks directive adalah sebagai berikut :

```
<%@ nama_directive atribut1="nilai1" atribut2="nilai2" . . . . %>
```

## **Jenis Directive**

Directive pada JSP terdiri atas tiga jenis tentu saja dengan fungsi yang berbeda-beda. Berikut adalah ketiga jenis directive tersebut :

- page directive
- include directive
- taglib directive

### **page directive**

Directive ini berfungsi untuk mendefinisikan atribut-atribut yang akan berlaku pada halaman tersebut. Sebagai contoh dengan menggunakan directive ini suatu halaman bisa diberikan

informasi mengenai apa, meng-import package-package yang akan digunakan, menyatakan halaman tersebut terlibat dalam HTTP session, mendefinisikan URL yang akan ditampilkan apabila terjadi error pada halaman JSP tersebut dan lain-lain. Pada sebuah halaman JSP dapat berisi atas banyak page directive.

Tanda yang digunakan untuk directive ini adalah :

```
<%@ page atribut1 atribut2 . . . %>
```

Atribut untuk tipe directive ini dapat dilihat pada contoh berikut :

### Atribut language

Atribut ini mendefinisikan bahasa pemrograman apa yang digunakan pada halaman tersebut. Atribut ini ada dikarenakan apabila dimasa yang akan datang JSP engine dapat men-support bahasa pemrograman lain. Berikut adalah contoh penggunaannya :

```
<%@ page language="java" %>
```

### Atribut import

Berikut adalah contoh penggunaannya :

```
<%@ page import="java.io.*, java.sql.*" %>
```

### Atribut info

Atribut ini hanya mendefinisikan informasi dari halaman. Dengan menggunakan atribut ini suatu aplikasi servlet dapat mengambil informasi tersebut dengan method

`Servlet.getServletInfo()`. Berikut adalah contohnya :

```
<%@ page info="Ini adalah halaman JSP-nya Choclove" %>
```

### Atribut errorPage

Atribut ini mendefinisikan URL yang akan ditampilkan apabila terjadi error pada halaman JSP tersebut.

```
<%@ page errorPage="error.jsp" %>
```

### Atribut contentType

Nilai default dari atribut ini adalah "text/html".

### Atribut session

Menyatakan halaman tersebut terlibat dalam HTTP session. Apabila halaman JSP menggunakan directive page dengan atribut ini artinya halaman tersebut nantinya akan digunakan untuk mengakses atau memberikan nilai pada variabel yang disimpan pada session.

```
<%@ page session="true" %>
```

Atribut lain yang dimiliki oleh directive ini adalah : `extends`, `buffer`, `autoFlush`, `isErrorPage` dan `isThreadSafe`.

Untuk penggunaan atribut tentu saja tidak harus dituliskan satu-satu seperti contoh di atas, tapi dapat disatukan sekaligus, seperti contoh berikut :

```
<%@ page language="java" import="java.sql.*, java.io.*, java.util.*"
session="true" buffer="24kb" autoFlush="true" info="Contoh penggunaan
Directive" errorPage="error.jsp" isErrorPage="false" isThreadSafe="false" %>
```

## include directive

Directive ini berfungsi untuk menyisipkan isi dari suatu file dengan tipe teks pada suatu halaman JSP. Sintaks yang digunakan oleh directive ini adalah :

```
<%@ include file="/namafile_yang_akan_disisipkan" %>
```

Bisa dilihat pada directive ini mempunyai satu atribut yaitu `file`. URL dari file yang akan disisipkan harus diawali dengan tanda `"/`. Apabila yang akan disisipkan adalah file `header.html` maka penulisannya adalah :

```
<%@ include file="/header.html" %>
```

Sedangkan apabila file `header.html` berada dalam direktori `"html-file"` maka penulisannya menjadi sebagai berikut :

```
<%@ include file="/html-file/header.html" %>
```

### Perhatian :

File yang disisipkan dengan directive ini sebaiknya adalah halaman statik, misalnya halaman HTML atau halaman JSP yang isinya tidak berubah-ubah. Karena “terkadang” apabila programmer mengubah isi dari file yang disisipkan (misalnya `header.html`) dan halaman JSP di-refresh maka perubahan itu tidak terjadi (pada Apache Tomcat 4.0). Pada Apache Tomcat 4.1 hal ini tidak terjadi.

## taglib directive

Directive ini berfungsi untuk penggunaan tag-tag yang dibuat sendiri oleh user pada halaman JSP. Tag-tag tersebut biasanya disimpan dalam “tag library” dalam bentuk file yang dikompres (ZIP atau JAR). Dalam file yang dikompres tersebut terdapat class-class dalam suatu paket. Dan untuk memanggil atau mengoperasikan method atau properti dalam class tersebut digunakan directive ini.

Sintaks dari directive ini adalah :

```
<%@ taglib uri="tag_library_URI" prefix="tag_prefix" %>
```

Atribut `uri` (Uniform Resource Identifier) berfungsi sebagai “tag library descriptor”. Dan atribut `prefix` berfungsi sebagai ID yang akan mempermudah “JSP Compiler” menentukan tag-tag dari “external library”. Tag-tag yang telah dikenali oleh “JSP Compiler” adalah `jsp`, `jspx`, `java`, `javax`, `servlet` dan `sunw`.

Berikut adalah contoh tag dari “external library” :

```
<%@ taglib uri="http://jakarta.apache.org/taglibs/application-1.0"
prefix="app" %>

<app:attribute name="test1"/>
<app:setAttribute name="test1">Isi Atribut</app:setAttribute>
```

Pada contoh di atas, secara default tidak dapat dijalankan dan akan menampilkan pesan error. Karena “tag library” ini belum diinstall pada sistem.

**Keterangan Artikel:**

Isi artikel ini hanya merupakan pengantar untuk sebagai materi pengenalan Directive pada JSP. Contoh-contoh yang lebih banyak akan dibahas pada bagian tersendiri. Selain itu juga akan dibahas bagaimana melakukan instalasi Tag Library yang baru dan penggunaannya.

**Catatan Penulis :**

Penulis adalah pemula dalam mempelajari Java, jadi mohon masukannya bagi pembaca yang menemukan kesalahan konsep atau asumsi yang digunakan penulis.