

Le pagine .jsp



Gli elementi fondamentali che si presentano spesso all'interno dello scripting JSP:

- Le **Dichiarazioni**.
- Le **Scriptlet**.
- Le **Espressioni**.
- Le Actions o **Standard Actions**



Scriptlet → `<% codice %>`

Dichiarazioni → `<%! codice %>`

Espressioni → `<%= codice %>`

Esempi:

- Dichiarazione → `<%! int x = 0; %>`
- Espressione → `<%= x++; %>`
- Scriptlet (dichiarazione + assegnamento) →
`<% java.util.Date data_di_oggi = new java.util.Date();
out.println(String.valueOf(data_di_oggi)); %>`

Oggetti impliciti

L'uso delle Espressioni da parte dello sviluppatore viene facilitato dall'esistenza di oggetti predefiniti che possono essere immediatamente utilizzati per produrre il comportamento per cui sono deputati.



request:

Istanza della classe **HttpServletRequest** associata alla richiesta di risorse generata in input dall'utente; grazie ad essa abbiamo la possibilità di rilevare parametri fondamentali (getParameter) come:

- metodi utilizzati per la richiesta (Post, Get..),
- intestazioni (headers..) ...

Esempio:

```
<%= request.getRemoteHost() %>
```

sfrutta l'oggetto request per richiedere il nome di Host del server (se testiamo lo script su Tomcat in locale a meno di nostre modifiche otterremo il classico 127.0.0.1).



response:

Istanza della classe HttpServletResponse associata alla risposta generata in output dal Servlet Engine; attraverso questa variabile possiamo notificare errori, spedire intestazioni, settare cookies,

application:

il ServletContext generato sulla base dell'azione operata sui parametri passati a `getServletConfig().getContext()`; permette di accedere al contesto in cui viene eseguita la Web application, sarà quindi possibile ottenere informazioni sull'Engine JSP e reperire parametri inizializzati.

config:

la ServletConfig riferita alla pagina;



pageContext:

istanza della classe PageContext. L'oggetto pageContext è in grado di rappresentare ogni elemento riconducibile al contesto in cui viene eseguita la pagina.

out:

il cosiddetto PrintWriter, a cui spetta la funzione fondamentale di inviare l'output al client da cui è stata spedita la richiesta attraverso il protocollo di comunicazione.

session:

HttpSession associato alla richiesta; ci consente d'identificare l'utente rendendo la sua sessione valida per tutto l'ambito di un'applicazione



Le actions

Le Actions o Standard Actions sono tags specifici che agiscono sul comportamento dei file ".jsp" e sugli output destinati a soddisfare le richieste

`<jsp:include page="index.jsp"/>`

→ fornisce un modo per includere risorse aggiuntive, di tipo statico o dinamico, nella Pagina JSP corrente. Gli attributi più importanti legati a questa azione sono:

- **page** : rappresenta la URL relativa della risorsa da includere;
- **flush** : questo attributo contiene un valore boolean che indica se il buffer debba o meno essere svuotato.

`<jsp:forward page=" index.jsp "/>`

→ consente al motore Jsp l'inoltro, all'atto dell'esecuzione, della risorsa corrente a una risorsa statica, a un Servlet o ad una Pagina JSP. L'unico attributo possibile per questa azione è:

- **page** : rappresenta la URL relativa all'oggetto da inoltrare.



```
<jsp:useBean id="myBean" scope="application"  
class="mypackage.MyBean" />
```

→ associa un'istanza di un JavaBean a un ambito e ad un ID tramite una variabile di script dichiarata con lo stesso nome; se non trova un'istanza "in vita" cerca di crearla.

Gli attributi più importanti legati a questa azione sono:

- **id** : rappresenta l'identità dell'istanza dell'oggetto all'interno dell'ambito specificato;
- **scope** : rappresenta l'ambito dell'oggetto. Possibili scope: page, request, session, application;
- **class** : rappresenta il nome di classe che definisce l'implementazione dell'oggetto.

```
<jsp:setProperty name="myBean" property="age" /> <jsp:include  
page="filename" />
```

→ questa azione principale derivano le seguenti sottoazioni:

- **<jsp:setProperty>**: imposta il valore di una proprietà di un bean;
- **<jsp:getProperty>**: prende il valore della proprietà del bean referenziato, lo converte in un `java.lang.String` e lo inserisce nell'oggetto out implicito;

Un JavaBeans è un componente Java al 100% che opera all'interno di una qualsiasi macchina virtuale; è una classe Java che implementa l'interfaccia `java.io.Serializable` e utilizza metodi pubblici `get/set` per esporre le proprietà.

`<jsp:param name="paramName" value="paramValue" />`

→ Viene impiegata per fornire informazioni a coppie di tag/valore, includendoli come sottoattributi delle azioni **<jsp:forward>**, **<jsp:useBean>**, **<jsp:include>** all'atto dell'esecuzione, della risorsa corrente a una risorsa statica, a un Servlet o ad una pagina JSP.

I suoi attributi sono:

- **name** : rappresenta il nome del parametro referenziato;
- **value** : rappresenta il valore del parametro referenziato.