

# Server Side Java web applications



# Hello World in JavaEE

Vedremo ora il primo esempio completo di una web app scritta in java sfruttando le tecnologie già introdotte

Obiettivi:

1. Stampare a video la scritta “Prova”
2. Effettuare il deploy del progetto e fare l'upload su Tomcat
3. Appena avviata l'applicazione web deve aprire la pagina index.jsp

Da Eclipse creiamo un Nuovo Progetto:

File → New → Dynamic Web Project



# Creazione delle pagine

## Project Explorer

- ▼ EsercizioJSP01
    - ▶ JAX-WS Web Services
    - ▶ Deployment Descriptor: EsercizioJSP01
    - ▼ Java Resources
      - src
      - ▶ Libraries
    - ▶ JavaScript Resources
    - ▶ build
    - ▼ WebContent
      - ▶ META-INF
      - ▼ WEB-INF
        - lib
- web.xml

## Project Explorer

- ▲ EsercizioJSP1
  - ▲ JAX-WS Web Services
    - Service Endpoint Interfaces
    - Web Services
  - ▲ Deployment Descriptor: EsercizioJSP1
    - Context Parameters
    - Error Pages
    - Filter Mappings
    - Filters
    - Listeners
    - References
    - Servlet Mappings
    - Servlets
    - Welcome Pages
  - ▲ Java Resources
    - src
    - ▲ Libraries
      - Apache Tomcat v7.0 [Apache Tomcat v7.0]
      - EAR Libraries
      - ▶ JRE System Library [jre7]
      - Web App Libraries
  - ▶ JavaScript Resources
  - ▶ Deployed Resources
  - ▶ build
  - ▲ WebContent
    - ▲ build
    - classes
    - index.jsp
    - ▶ META-INF
    - ▲ WEB-INF

# web.xml

Il file web.xml è chiamato **deployment descriptor**, istruisce il servlet container (es: Tomcat) su tutte le informazioni relative alla configurazione della web app come quali classi caricare e quali pagine web la compongano

- Servlet e pattern URL per invocarle.
- Filtri e le loro mappature.
- Listeners – classi invocate al verificarsi di particolari eventi
- Parametri di configurazione
- Pagine di errore



\*web.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://www.w3.org/2001/XMLSchema-instance"
  <display-name>Esercizio1</display-name>
  <welcome-file-list>
    <welcome-file>index.html</welcome-file>
    <welcome-file>index.htm</welcome-file>
    <welcome-file>index.jsp</welcome-file>
    <welcome-file>default.html</welcome-file>
    <welcome-file>default.htm</welcome-file>
    <welcome-file>default.jsp</welcome-file>
  </welcome-file-list>
</web-app>
```



# index.jsp

Il codice Java della pagina "index" è stato isolato tra i delimitatori "<%" e "%>" in modo da poter essere riconosciuto come tale.

L'istruzione `out.println` ha il compito di stampare a video il parametro che le viene passato

A screenshot of a code editor window titled 'index.jsp'. The code is a JSP page that sets the page language to 'java', content type to 'text/html', and page encoding to 'ISO-8859-1'. It includes a DOCTYPE declaration for HTML 4.01 Transitional. The page structure includes a head section with a title 'Pagina JSP' and a body section containing a Java out.println statement that prints 'Prova' wrapped in an h1 tag.

```
index.jsp
<%@ page language="java" contentType="text/html; charset=
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional,
<html>
<head>
<title>Pagina JSP</title>
</head >
<body>
<% out.println("<h1>Prova</h1>"); %>
</body>
</html>
```

## Esempio n.2 → Colleghiamo due pagine jsp



```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<div style="text-align:center">
<form name="Login" action="news.jsp">
  Categoria: <input name="categoria" type="text" /><br />
  <input type="submit" value="Show" />
</form>
</div>
</body>
</html>
```

**news.jsp** → La seconda jsp stampa il risultato inserito nel form della prima

```
<body>
<% if(request.getParameter("categoria").equals("politica")){ %>
    <table>
        <tr><td style="text-align:center">Prima sezione= politica</td></tr>
    </table>
<% }else if(request.getParameter("categoria").equals("finanza")){ %>
    <table>
        <tr><td style="text-align:center">Seconda sezione= finanza</td></tr>
    </table>
<% }else if(request.getParameter("categoria").equals("sport")){ %>
    <table>
        <tr><td style="text-align:center">Terza sezione= sport</td></tr>
    </table>
<% }else{ %>
    <h4>Errore</h4>
<%
}
%>
</body>
```