

Wykład Java Server Pages (JSP)

Zaawansowane programowanie w Javie

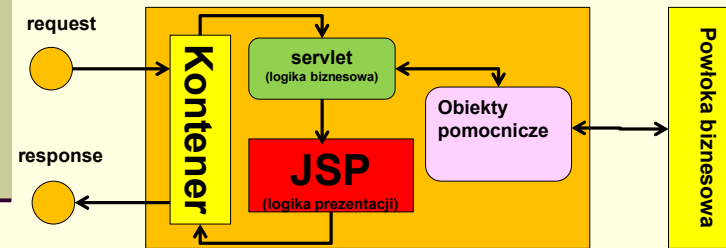
Dr inż. Damian Raczyński

Wprowadzenie – czym jest JSP



Widok dynamicznych elementów

Obsługuje logikę prezentacji w modelach MVC



3

Czego można się spodziewać na wykładzie

1. Wprowadzenie,
2. Elementy JSP,
3. Obiekty bezpośrednie (Implicit Objects),
4. Biblioteki znaczników,
5. Akcje JSP,
6. Obsługa błędów.

2

Do czego można używać serwletów?



- uwierzytelnianie użytkowników,
- odwołania do bazy danych,
- filtrowanie przychodzących żądań od użytkowników,
- logika w aplikacji,
- przekierowanie do JSP.

4

Wady servletów

Tworząc odpowiedź (stronę Internetową) z wykorzystaniem servletów konieczne jest umieszczanie kodu tworzonej strony w instrukcjach języka Java – przykładowo:

```
out.print("<h1> nagłówek </h1> ...");
```

Tworzony servlet stanowi zatem kod Javy z „wklejonymi” znacznikami HTML.

Odwrotność do servletów stanowi technologia JSP – wykorzystywany jest HTML z „wklejonym” kodem Java

5

JSP

Servlety

Obie technol
loginy użytko
Przeznaczeni

Metody cyklu życia:
init() – może zostać nadpisana
service() – może zostać nadpisana
destroy() – może zostać nadpisana

język HTML wewnątrz JAVY

Generalnie można przyjąć zasadę, że metody, których nazwa rozpoczyna się od podkreślenia nie mogą zostać nadpisane.

Metody cyklu życia:
jspinit() – może zostać nadpisana
_jspService() – nie może zostać nadpisana
jspDestroy() – może zostać nadpisana

JAVA wewnątrz HTML

Obie technologie działają wyłącznie w obrębie kontenera Web

JSP

Java Server Pages (JSP) jest wykorzystywane do prezentacji informacji. JSP stanowi także servlet.

JSP umożliwia zamieszczanie zwyczajnego kodu HTML oraz informacji generowanych przez serwlety.

Dokumenty JSP są automatycznie przekształcane do postaci serwletów.

Oprócz kodu HTML w JSP mogą pojawić się:
elementy skryptowe,
dyrektywy,
akcje.

6

JSP

JSP tak na prawdę jest zwykłym servletem!

Transformacja

Tłumaczenie

Po załadowaniu przez web kontener

JSP
np.
str.jsp

servlet
str_jsp.java

kod bajtowy
00101010100
0110
str_jsp.class

servlet
str_jsp
Servlet

8

JSP



Jakie elementy strony JSP zapewniają dynamiczną zawartość ????

Deklaracje JSP – kod, który trafia na zewnątrz metody `service`

Skryptlety JSP - elementy stanowią kod Javy realizujący pewną funkcjonalność w metodzie `service`.

Wyrażenia JSP – kod wyrażen jest wyznaczany

Dyrektwy JSP - pozwalają na określenie struktury generowanego serwletu.

Akcje JSP - rozszerzenie funkcjonalności.

9

Elementy skryptowe



Deklaracje:

`<%! KOD_JAVA %>` - znacznik deklaracji

Wszystko, co zostanie zamieszczone w znaczniku zostaje umieszczone na zewnątrz metody `service`

Co tu zamieścić?

Deklarację metod i zmiennych

Po co używać znacznika?

Przykłady – zmienne współdzielone pomiędzy żądaniami, powtarzający się kod w JSP można zamieścić w metodzie

11

Elementy skryptowe



Jakie elementy strony JSP zapewniają dynamiczną zawartość ??? - cd

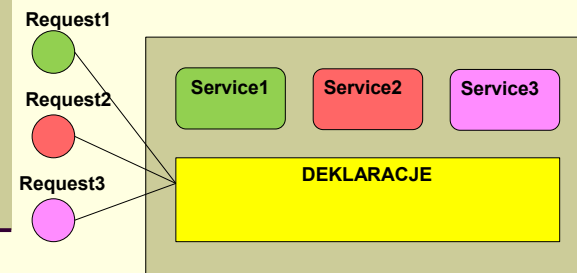
Wyrażenia: `<%= wyrażenie %>` - wartość jest obliczana i wstawiana do kodu

skryptlety: `<% kod %>` - kod wstawiany do metody `_jspService` serwletu

deklaracja: `<%! kod %>` - kod wstawiany wewnątrz klasy serwletu (poza metodami).

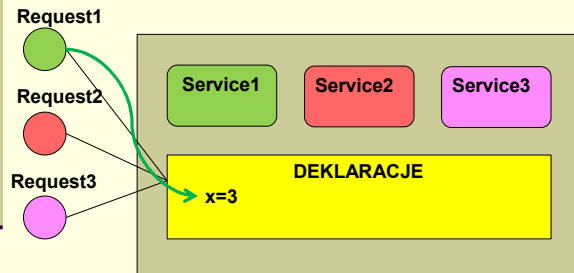
10

Elementy skryptowe



12

Elementy skryptowe



13

Elementy skryptowe



Skryptety:

Znacznik `<% KOD_JAVY %>` nazywany jest skryptetem.

Wszystko co zostanie zawarte w tym znaczniku wędruje do metody `service`

Wszystkie zmienne zadeklarowane w skrypcie są zmiennymi lokalnymi metody!

Co należy umieścić w skrypcie?

Logikę biznesową po stronie JSP

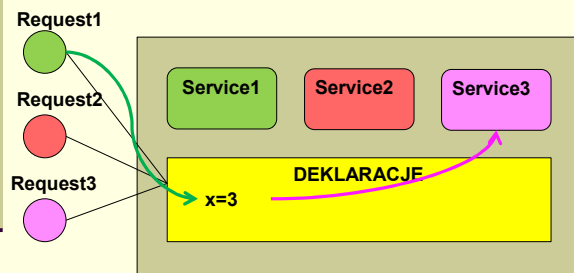
Po co skryptety są potrzebne?

Wykonanie niewielkich operacji związanych z logiką biznesową (jeżeli operacje są skomplikowane powinny zostać umieszczone w `servlecie`)

Wykonanie prostej walidacji danych

15

Elementy skryptowe



Zmienne zadeklarowane w blokach deklaracji są widoczne pomiędzy zadaniami!!!

14

Elementy skryptowe



Wyrażenia:

Znaczniki `<%= KOD_JAVY%>` nazywane są znacznikami wyrażień

Kod javy umieszczony w wyrażeniu jest wykonywany, natomiast zwrócone wartości wyświetlane są na stronie

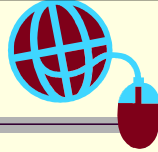
Cokolwiek jest umieszczane wewnątrz wyrażenia powinno zwracać wartość.

Co umieszczać wewnątrz wyrażień?

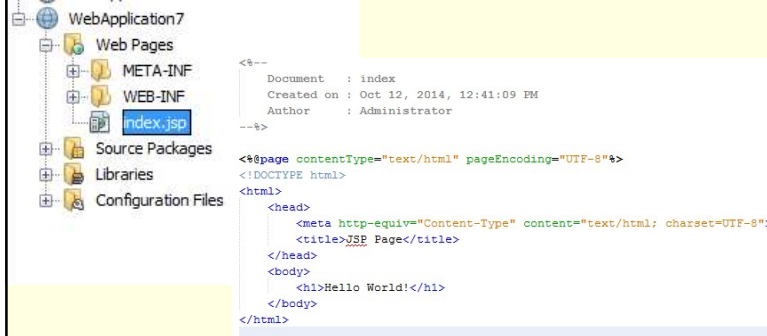
Zmienne lub metody zwracające wartość

16

JSP



Tworzenie strony JSP:



17

Elementy skryptowe



Predefiniowane zmienne:

| Zmienna | Znaczenie |
|-----------------|---|
| request | obiekt <code>HttpServletRequest</code> |
| response | obiekt <code>HttpServletResponse</code> |
| session | obiekt <code>HttpSession</code> |
| out | obiekt <code>JspWriter</code> |

19

Elementy skryptowe



Przykład:

```

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>JSP Page</title>
</head>
<body>
<h1>Hello World!</h1>
<p>
<%= new java.util.Date() %>
</p>
</body>
</html>
  
```

18

Elementy skryptowe



Przykład:

```

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>JSP Page</title>
</head>
<body>
<h1>Hello World!</h1>
<p>
<%= request.getRemoteHost() %>
</p>
</body>
</html>
  
```

20

Elementy skryptowe



Skryptlety pozwalają na wstawienie dowolnego kodu Java:
`<% kod java %>`.

Przykład: np. w jaki sposób ustawić kolor tła strony Internetowej na wartość przekazaną jako parametr URL?

Przykład w NetBeans

21

Elementy skryptowe



Skryptlety mogą zostać również wykorzystane do warunkowego przetwarzania strony Internetowej.

Konstrukcja typu:

```
<% if (warunek) { %>
    HTML
<% } else { %>
    HTML
<% } %>
```

Wykona wyłącznie jeden z bloków HTML.

23

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>JSP Page</title>
  </head>
  <%
    String kolor=request.getParameter("kolorTla");
    boolean kolorPodany;
    if(kolor!=null) kolorPodany=true;
    else { kolorPodany=false; kolor="WHITE"; }
  %>

  <body bgcolor="<%= kolor %>">
    <h1>Hello World!</h1>
    <p>
      <%= request.getRemoteHost() %>
    </p>
  </body>
</html>
```

Hello World!

```
<body>
  <h1>Rzut kostką gra v 1.0</h1>
  <%
    int losowa = (int) Math.floor(Math.random()*10000)%6+1;
    String strzal=request.getParameter("strzal");
    int podana=Integer.parseInt(strzal);
  %>
  <p>
    Podałeś: <%= podana %>
    Wylosowano: <%= losowa %>
    <%if(podana==losowa){ %>
    <b> BRAWO !!!!!!!!!!!!!!! </b>
    <% }else{ %>
    <b> nie zniechęcaj się ! </b>
    <% } %>
  </p>
</body>
```