



Интернет програмирање

предавање 16

Проф. др Мирослав Лутовац
mlutovac@viser.edu.rs

PROGRAMIRANJE INTERNET APLIKACIJA

udžbenik sa zbirkom zadataka

rti.etf.bg.ac.rs PIA_knjiga_2017_v0.9_draft.pdf

ELEKTROTEHNIČKI FAKULTET UNIVERZITETA U BEOGRADU
maj 2017. godine

Prof. dr Boško Nikolić, Dražen Drašković

PROGRAMIRANJE INTERNET APLIKACIJA - udžbenik sa zbirkom zadataka

Recenzenti: Doc. dr Zaharije Radivojević, Doc. dr Miloš Cvetanović

Izdavač: Elektrotehnički fakultet Univerziteta u Beogradu

Akadska misao, Beograd

Na osnovu člana 42. Statuta Elektrotehničkog fakulteta i člana 14. Pravilnika o udžbenicima i drugoj nastavnoj literature, kao i pozitivne recenzije dr Zaharija Radivojevića i dr Miloša Cvetanovića, Nastavno-naučno veće Elektrotehničkog fakulteta je na svojoj sednici održanoj dana 22. marta 2016. godine donelo Odluku o odobravanju nastavnog materijala za štampu pod nazivom "Programiranje internet aplikacija - udžbenik sa zbirkom rešenih zadataka" (broj 2154/2).

Copyright © 2017, Elektrotehnički fakultet Univerziteta u Beogradu



Рад са више сервлета

**PROGRAMIRANJE
INTERNET
APLIKACIJA**
udžbenik sa zbirkom zadataka

- Сервлети са прослеђивањем параметара
- Решен задатак 3 из збирке корак-по-корак
- Покретање и заустављање сервера
- Провера исправности рада сервлета

Сервлети са прослеђивањем параметара, задатак 3

- Коришћењем Јава сервлет технологије написати html страну која има форму са текстуалним пољем и дугме за потврду.
- ✓ Страна html треба да позове први сервлет, који ће унети параметар (текст унет кроз форму) проследити другом сервлету коришћењем:
 1. `sendRedirect` методе
 2. објекта класе `RequestDispatcher` и `forward` методе
- У првом сервлету осим параметра који уноси клијент, треба поставити још два параметра,
 - ✓ један који се прослеђује другом сервлету преко захтева (`request`) и
 - ✓ један који се прослеђује другом сервлету преко сесије (односно објекта класе `HttpSession`).

Сервлети са прослеђивањем параметара, задатак 3

- За први део задатка
 - Произвољно се бира име фолдера у коме су сви сервлет фолдери и фајлови, "servlet03" и који реализују.
- За други део задатка
 - Направи се фолдер "servlet03b".
 - Радни директоријум "D:\myProject\tomcat"
 - Сервлети у фолдеру "webapps", где су и *html* фајлови

Избор назива фолдера и фајлова, структура на серверу

d:\myProject\



tomcat

javac



bin

server.xml
web.xml
context.xml
tomcat-users.xml



conf



webapps

У фолдеру **webapps**
праве се нови фолдери:
servlet03 који реализују
први део задатка,
servlet03b за други део
задатка



servlet03b



servlet03



WEB-INF



classes

Сваки сервлет има свој
WEB-INF и *classes*

index.html који позва сервлет

- Сервлет је у фолдеру "servlet03", и путања је "D:\myProject\tomcat\webapps\servlet03";
- фолдер "servlet03b", има путању "D:\myProject\tomcat\webapps\servlet03b"

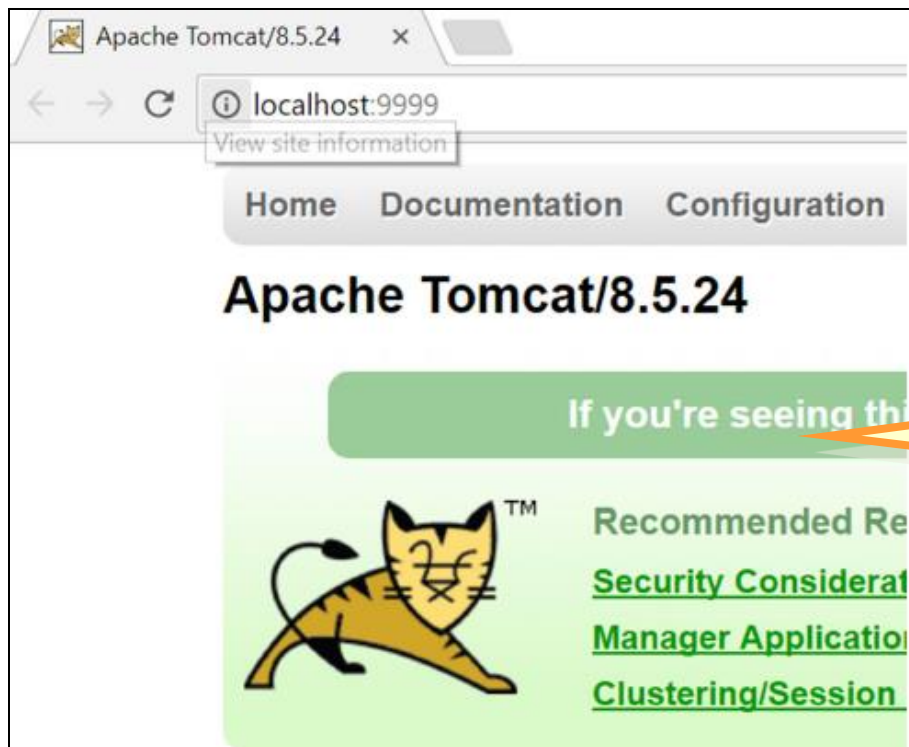
```
<!DOCTYPE html>
<html>
  <head>
    <title></title>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
  </head>
  <body>
    <form action="Servlet1" method="get">
      Unesite tekst i kliknite posalji:<br>
      <input type="text" name="mojparametar"/><br/>
      <input type="submit" value="Posalji"/>
    </form>
  </body>
</html>
```

"D:\myProject\tomcat\webapps\servlet03\index.html"



Тестирање

- Стартује се сервер,
- Покрене прегледач
- У прегледач се унесе адреса **http://localhost:9999/**



Ако је све у реду,
добива се
поздравни екран сервера

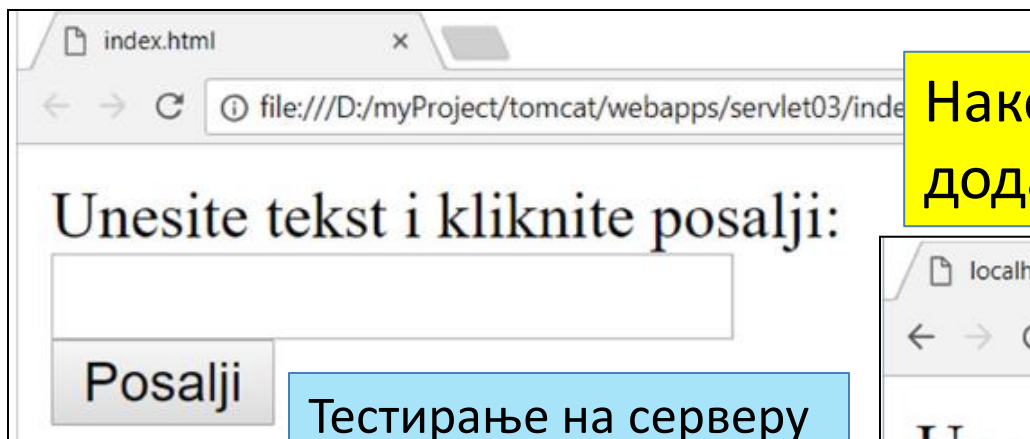
Приказ *index.html* са клијентовог рачунара

"http://localhost:9999/servlet03/"

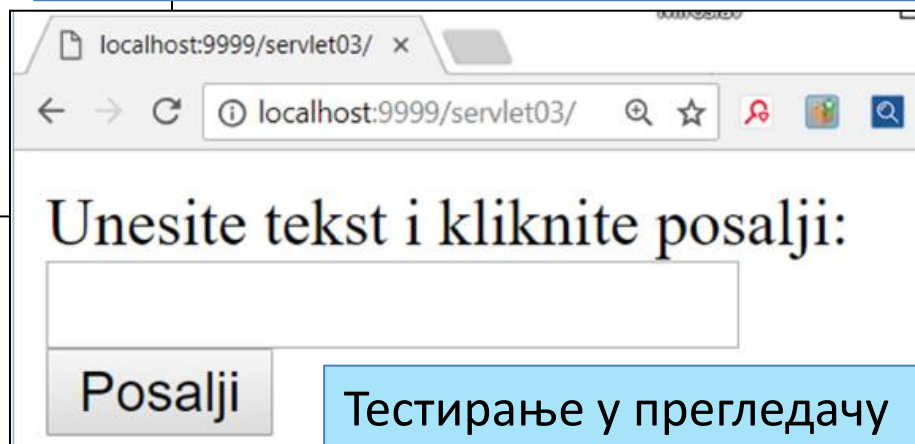
"http://localhost:9999/servlet03/index.html"

Унесе се адреса где треба да буде сервлет

Прегледач интерпретира и приказује на клијентовој страни



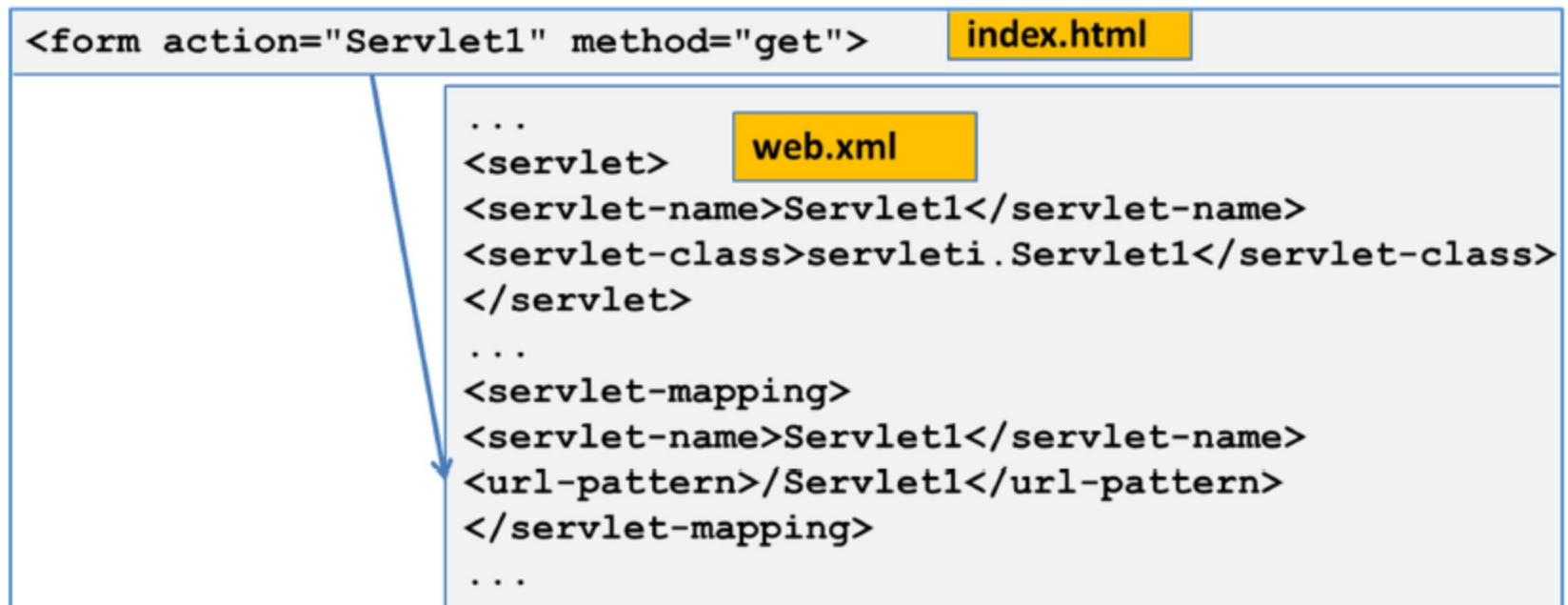
Након клика, методом "Get"
додаје се адреси текст "Servlet1"



"http://localhost:9999/servlet03/Servlet1"

Направити *web.xml*

Сервер у *web.xml* треба да пронађе назив "Servlet1" из *index.html* који користи за мапирање



Мапирања за два сервлета

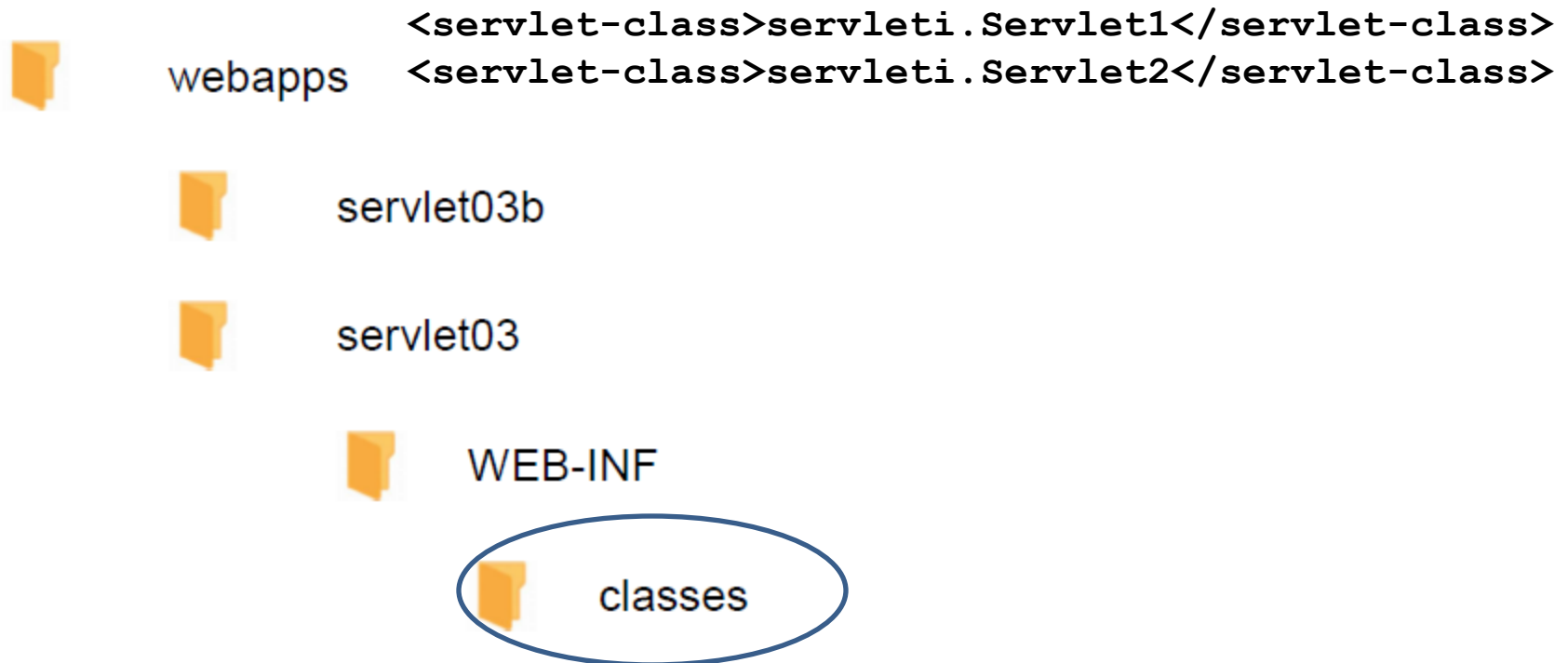
треба да се направе два сервлета, за оба да се одреде називи који ће обезбедити коректно мапирање у различите јава класе

web.xml

```
...  
<servlet>  
<servlet-name>Servlet1</servlet-name>  
<servlet-class>servleti.Servlet1</servlet-class>  
</servlet>  
<servlet>  
<servlet-name>Servlet2</servlet-name>  
<servlet-class>servleti.Servlet2</servlet-class>  
</servlet>  
<servlet-mapping>  
<servlet-name>Servlet1</servlet-name>  
<url-pattern>/Servlet1</url-pattern>  
</servlet-mapping>  
<servlet-mapping>  
<servlet-name>Servlet2</servlet-name>  
<url-pattern>/Servlet2</url-pattern>  
</servlet-mapping>  
...
```

не зна се ни ко ће,
ни када,
позвати сервлете

За две Јава класе потребна су два јава фајла.
Уместо једног Јава фајла који је био у фолдеру classes,
у фолдеру classes сада ће бити поддиректоријум servleti,
а у фолдеру servleti ће бити изворни Јава кодови и
њихове компајлиране верзије.
Овај фолдер се назива пакет, package servleti.



Servlet1 има функцију да направи html форму за унос текстуелног садржаја, који треба да проследи следећем сервлету, укључујући и параметре из јава класе

package servleti

```
package servleti;
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
public class Servlet1 extends HttpServlet {
protected void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
String ime="VISER";
request.setAttribute("param2", ime);
HttpSession sesija = request.getSession();
sesija.setAttribute("param3", ime);
//zadatak pod a
response.sendRedirect("Servlet2");
//zadatak pod b
//RequestDispatcher rd=request.getRequestDispatcher("Servlet2");
//rd.forward(request, response);
}
}
```

Servlet2 генерише одговарајући *html* фајл за прегледач као завршетак сесије

```
package servleti;  
import java.io.*;  
import javax.servlet.*;  
import javax.servlet.http.*;  
public class Servlet2 extends HttpServlet {  
    protected void doGet(HttpServletRequest request, HttpServletResponse response)  
        throws ServletException, IOException {  
        response.setContentType("text/html; charset=UTF-8");  
        PrintWriter out = response.getWriter();  
        try {  
            HttpSession sesija = request.getSession();  
            String parametar=request.getParameter("mojparametar");  
            String param2 = request.getParameter("param2");  
            String param3 = (String)sesija.getAttribute("param3");  
            out.println("<html>");  
            out.println("<head>");  
            out.println("<title>Servlet servlet2</title>");  
            out.println("</head>");  
            out.println("<body>");  
            out.println("<h1>Servlet servlet2: <br/> parametar=" + parametar +  
                "<br/> param2 (iz request-a)=" + param2 +  
                "<br/> param3 (iz sesije)=" + param3 + "</h1>");  
            out.println("</body>");  
            out.println("</html>");  
        } finally {  
            out.close();  
        }  
    }  
}
```

package servleti

Компајлирање

- ✓ Из програма у јава формату генеришу се `class` фајлови компајлирањем јава фајлова
- ✓ Може се користити развојно окружење као што је Еклипс, с тим да се сви `.class` фајлови поставе у фолдер `classes`
- ✓ За стартовање сервера могу да се користе
 - `command prompt`
 - програма сервер менаџер или
 - развојно окружење Еклипс


```
Command Prompt
Microsoft Windows [Version 10.0.16299.371]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Users\MIROSLAV>startServer.bat

C:\Users\MIROSLAV>d:

D:\>cd \myProject\tomcat\bin

D:\myProject\tomcat\bin>startup
Using CATALINA_BASE:  "D:\myProject\tomcat"
Using CATALINA_HOME:  "D:\myProject\tomcat"
Using CATALINA_TMPDIR: "D:\myProject\tomcat\temp"
Using JRE_HOME:       "c:\Program Files\Java\jdk1.8.0_121"
Using CLASSPATH:      "D:\myProject\tomcat\bin\bootstrap.jar;D:\myProject\tomcat\bin\tomcat-juli.jar"
D:\myProject\tomcat\bin>
```

```
Tomcat
09-May-2018 08:16:05.290 INFO [localhost-startStop-1] org.apache.catalina.startup
p.HostConfig.deployDirectory Deploying web application directory [D:\myProject\t
omcat\webapps\servlet03]
09-May-2018 08:16:05.308 INFO [localhost-startStop-1] org.apache.catalina.startup
p.HostConfig.deployDirectory Deployment of web application directory [D:\myProje
ct\tomcat\webapps\servlet03] has finished in [18] ms
09-May-2018 08:16:05.308 INFO [localhost-startStop-1] org.apache.catalina.startup
p.HostConfig.deployDirectory Deploying web application directory [D:\myProject\t
omcat\webapps\servlet03b]
09-May-2018 08:16:05.448 INFO [localhost-startStop-1] org.apache.catalina.startup
p.HostConfig.deployDirectory Deployment of web application directory [D:\myProje
ct\tomcat\webapps\servlet03b] has finished in [140] ms
09-May-2018 08:16:05.448 INFO [localhost-startStop-1] org.apache.catalina.startup
p.HostConfig.deployDirectory Deploying web application directory [D:\myProject\t
omcat\webapps\statickaWebStrana]
09-May-2018 08:16:05.464 INFO [localhost-startStop-1] org.apache.catalina.startup
p.HostConfig.deployDirectory Deployment of web application directory [D:\myProje
ct\tomcat\webapps\statickaWebStrana] has finished in [16] ms
09-May-2018 08:16:05.464 INFO [main] org.apache.coyote.AbstractProtocol.start St
arting ProtocolHandler ["http-nio-9999"]
09-May-2018 08:16:05.480 INFO [main] org.apache.coyote.AbstractProtocol.start St
arting ProtocolHandler ["ajp-nio-8009"]
09-May-2018 08:16:05.480 INFO [main] org.apache.catalina.startup.Catalina.start
Server startup in 1135 ms
```

```
Command Prompt
D:\myProject\tomcat\bin>dir
Volume in drive D is LENOVO
Volume Serial Number is 1AB4-84F3

Directory of D:\myProject\tomcat\bin

05/07/2018  11:10 PM    <DIR>          .
05/07/2018  11:10 PM    <DIR>          ..
01/21/2018  12:13 PM             34,738 bootstrap.jar
01/21/2018  12:13 PM             1,703 catalina-tasks.xml
01/21/2018  12:13 PM            15,815 catalina.bat
01/21/2018  12:13 PM            23,315 catalina.sh
01/21/2018  12:13 PM           207,125 commons-daemon-native.tar.gz
01/21/2018  12:13 PM            25,145 commons-daemon.jar
01/21/2018  12:13 PM             2,040 configtest.bat
01/21/2018  12:13 PM             1,922 configtest.sh
01/21/2018  12:13 PM             8,509 daemon.sh
01/21/2018  12:13 PM             2,091 digest.bat
01/21/2018  12:13 PM             1,965 digest.sh
02/15/2018  02:28 PM              450 javacServlet.bat
02/25/2018  01:47 AM              455 javacServlet01.bat
02/25/2018  04:41 PM              425 javacServlet02.bat
05/07/2018  10:42 PM              347 javacServlet03.bat
05/07/2018  11:11 PM              348 javacServlet03b.bat
02/19/2018  12:50 PM              458 javacServlet2.bat
01/21/2018  12:13 PM            3,574 setclasspath.bat
01/21/2018  12:13 PM            3,680 setclasspath.sh
01/21/2018  12:13 PM            2,020 shutdown.bat
01/21/2018  12:13 PM            1,902 shutdown.sh
01/21/2018  12:13 PM            2,022 startup.bat
01/21/2018  12:13 PM            1,904 startup.sh
01/21/2018  12:13 PM           48,595 tomcat-juli.jar
01/21/2018  12:13 PM          405,109 tomcat-native.tar.gz
01/21/2018  12:13 PM            4,574 tool-wrapper.bat
01/21/2018  12:13 PM            5,483 tool-wrapper.sh
01/21/2018  12:13 PM            2,026 version.bat
01/21/2018  12:13 PM            1,908 version.sh
                29 File(s)          809,648 bytes
                2 Dir(s)  21,534,752,768 bytes free

D:\myProject\tomcat\bin>
```

*C:\Users\MIROSLAV\startServer.bat - Notepad++

File Edit Search View Encoding Language Settings To Window ?

startServer.bat x

```
1 d:
2 cd \myProject\tomcat\bin
3 startup
4
```

Ln:4 Col:1 Sel:0|0 Windows (CR LF) UTF-8 INS

D:\myProject\tomcat\bin\javacServlet03.bat - Notepad++

File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?

javacServlet03.bat

```
1 :: Assume that Tomcat is installed in d:\myProject\tomcat
2 :: Change directory to the source file
3 d:
4 cd \myProject\tomcat\webapps\servlet03\WEB-INF\classes\servleti
5 :: Compile
6 SET PATH=%PATH%;%JAVA_HOME%\bin
7 javac -cp .;d:\myProject\tomcat\lib\servlet-api.jar Servlet1.java
8 javac -cp .;d:\myProject\tomcat\lib\servlet-api.jar Servlet2.java
```

Batch file length: 347 lines: 9 Ln: 9 Col: 1 Sel: 0 | 0 Windows (CR LF) UTF-8 INS

```
C:\Program Files\NVIDIA Corporation\PhysX\Common;C:\WINDOWS\system32;C:\WINDOWS;C:\WINDOWS\System32\Wbem;C:\WINDOWS\System32\WindowsPowerShell\v1.0\;C:\Program Files\Intel\Intel(R) Management Engine Components\DAL;C:\Program Files\Intel\Intel(R) Management Engine Components\IPT;C:\Program Files (x86)\Intel\Intel(R) Management Engine Components\DAL;C:\Program Files (x86)\Intel\Intel(R) Management Engine Components\IPT;C:\Program Files\Intel\WiFi\bin\;C:\Program Files\Common Files\Intel\WirelessCommon\;C:\Program Files\Anaconda3;C:\Program Files\Anaconda3\Scripts;C:\Program Files\Anaconda3\Library\bin;C:\Users\MIROSLAV\Anaconda3;C:\Users\MIROSLAV\Anaconda3\Scripts;C:\Users\MIROSLAV\Anaconda3\Library\bin;C:\Users\MIROSLAV\AppData\Local\Microsoft\WindowsApps;c:\Program Files\Java\jdk1.8.0_121\bin

D:\myProject\tomcat\webapps\servlet03\WEB-INF\classes\servleti>javac -cp .;d:\myProject\tomcat\lib\servlet-api.jar Servlet1.java

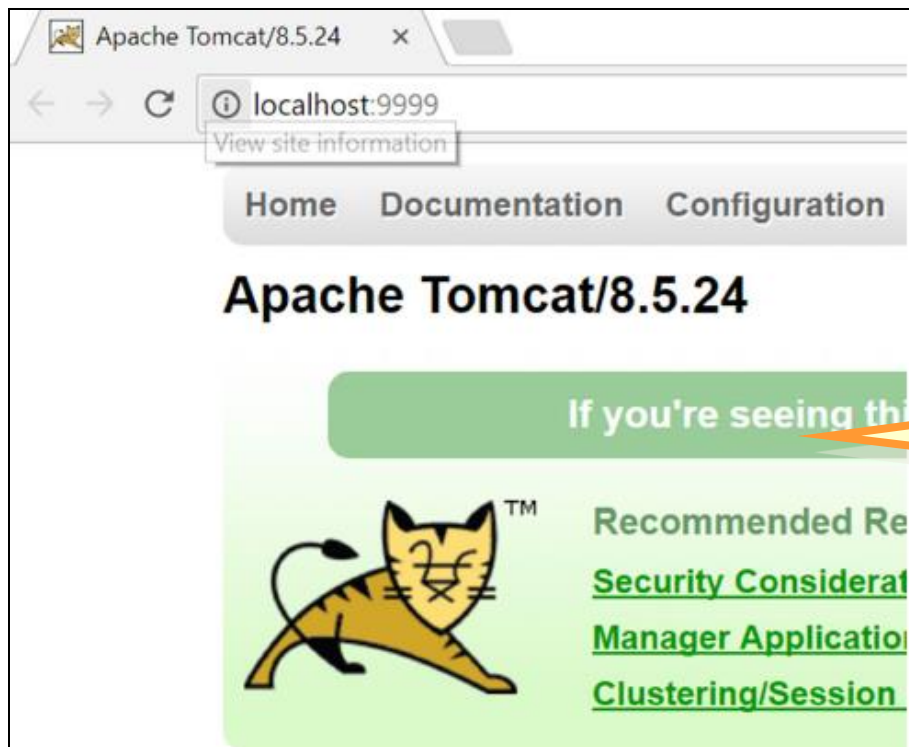
D:\myProject\tomcat\webapps\servlet03\WEB-INF\classes\servleti>javac -cp .;d:\myProject\tomcat\lib\servlet-api.jar Servlet2.java

D:\myProject\tomcat\webapps\servlet03\WEB-INF\classes\servleti>
```

19

Тестирање

- Стартује се сервер,
- Покрене прегледач
- У прегледач се унесе адреса **http://localhost:9999/**



Ако је све у реду,
добаја се
поздравни екран сервера

Тестирање

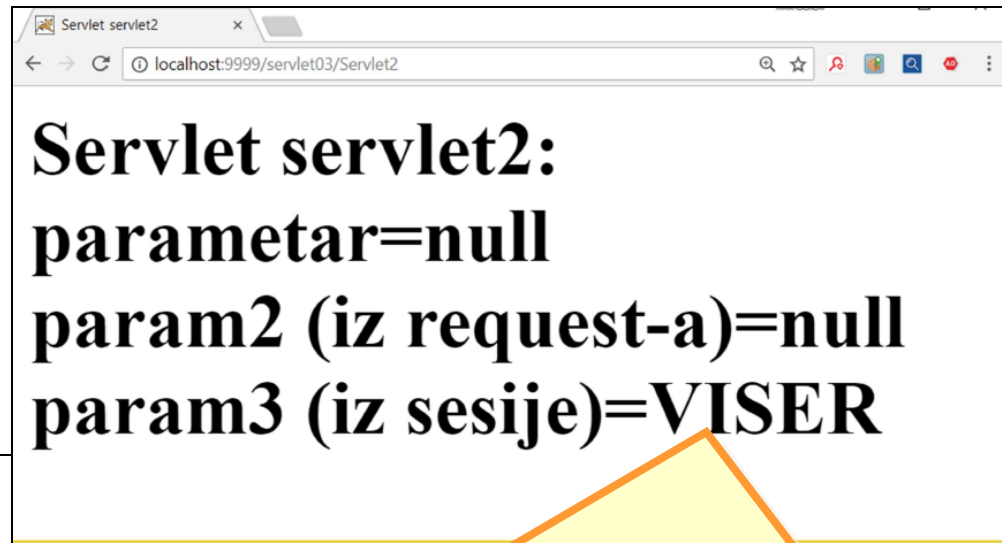
- адресе у прегледачу на клијентовој страни
"http://localhost:9999/**servlet03**",
сервер приказује садржај фајла "index.html"



подразумевани
фајл је *index.html*

Тестирање

Види се да је други сервлет приказао само вредност параметра из првог сервлета, али није приказан унети текст



```
<html>  
<head>  
<title>Servlet servlet2</title>  
</head>  
<body>  
<h1>Servlet servlet2: <br/> parametar=null<br/> param2 (iz request-a)=null<br/>  
param3 (iz sesije)=VISER</h1>  
</body>  
</html>
```

после уноса и клика на дугме првог сервлета

"http://localhost:9999/servlet03/servlet2/"

Тестирање

- адресе у прегледачу на клијентовој страни

"http://localhost:9999/**servlet03b/**",

сервер приказује садржај фајла "index.html"



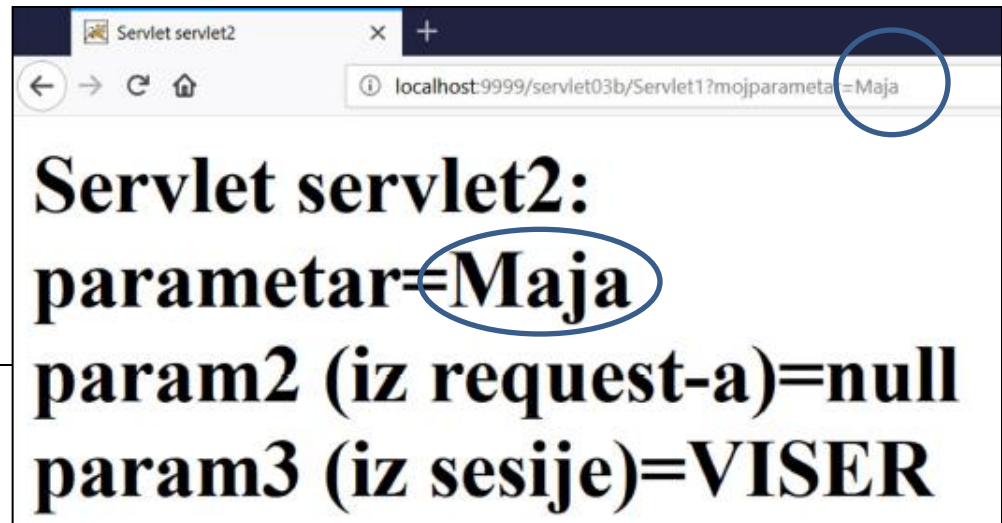
подразумевани
фајл је *index.html*

```
...
sesija.setAttribute("param3", ime);
//zadatak pod a
response.sendRedirect("Servlet2");
//zadatak pod b
//RequestDispatcher rd=request.getRequestDispatcher("Servlet2");
//rd.forward(request, response);
}
}
```


Тестирање

У адреси се појављује вредност параметра који је пренет са првог на други сервлет, и он се приказује у прегледачу као вредност за parametar

```
<html>
<head>
<title>Servlet servlet2</title>
</head>
<body>
<h1>Servlet servlet2: <br/> parametar=Maja<br/> param2 (iz request-a)=null<br/>
param3 (iz sesije)=VISER</h1>
</body>
</html>
```



"http://localhost:9999/servlet03b/Servlet1?mojparametar=Maja"

Задатак 1 из области JSP

- ip **11** PT&SCHE.pdf Java Server Pages (JSP),
решен **задатак 1 из области JSP** из збирке
корак-по-корак

Java Server Pages

- Java Server Pages (JSP) је програмска технологија на серверу која омогућава креирање динамичких веб апликација и која подржава динамичке садржаје, независно од платформе на којој се извршава
- JSP помаже програмерима да укључе Јава код на HTML странице користећи посебне JSP тагове, од којих већина почиње са `<%` и завршава са `%>`.
- JSP компонента је тип Јава сервлета који је дизајниран да испуни улогу корисничког интерфејса за Јава веб апликацију

Java Server Pages

- Веб програмери уписују JSP као текстуалне датотеке који комбинују HTML или XHTML код, XML елементе и уграђене JSP акције и команде
- Користећи JSP, могу се
 - прикупљати подаци од корисника путем *Webpage* форми
 - приказати записи из базе података или другог извора
 - динамички креирати веб странице
- JSP тагови могу се користити за:
 - преузимање података из базе података
 - регистрација корисничких преференци
 - приступање *JavaBeans* компонентама
 - преношење контроле између страница
 - размена информација између захтева и страница

Једностави JSP пример

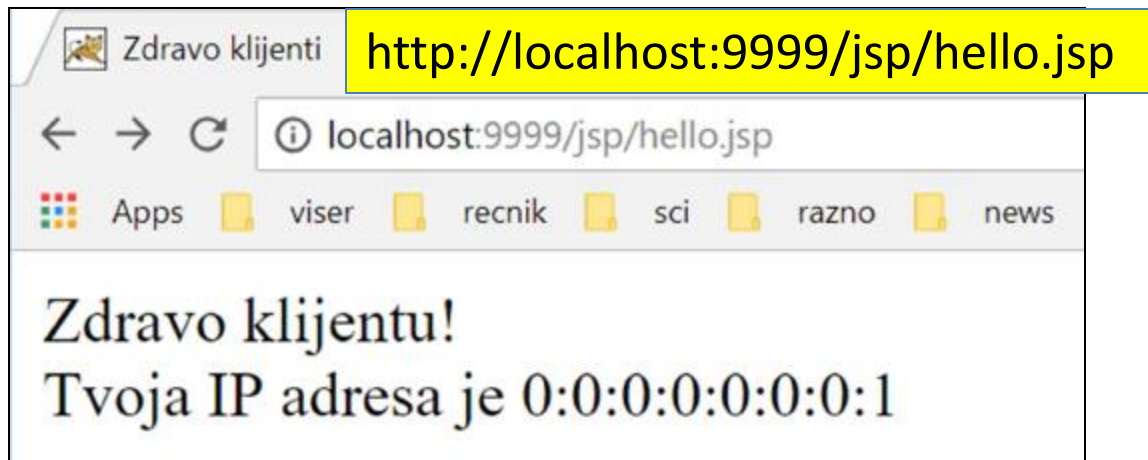
- Стартује се сервер, покрене се прегледач и унесе адреса "http://localhost:9999/" да се провери да ли сервер ради
- Унесе се адреса где се налази JSP; код за веб апликацију се налази у фолдеру "webapps"; за примере у JSP, направи се фолдер "jsp" испод "webapps"; У фолдеру "webapps/jsp", направи се је фајл ".jsp" код, на пример "hello.jsp"

```
<html>
<head><title>Zdravo klijenti</title></head>
  <body>
    Zdravo klijentu!<br/>
    <%
      out.println("Tvoja IP adresa je " + request.getRemoteAddr());
    %>
  </body>
</html>
```

Једностави JSP пример

- У прегледач се унесе адреса
"http://localhost:9999/jsp/hello.jsp"
- Пуна путања на диску рачунара је
"d:\myProject\tomcat\webapps\jsp\hello.jsp"

могу постојати
разлике у косим
цртама у
прегледачу (/)
и на диску у
зависности од
оперативног
система (\)



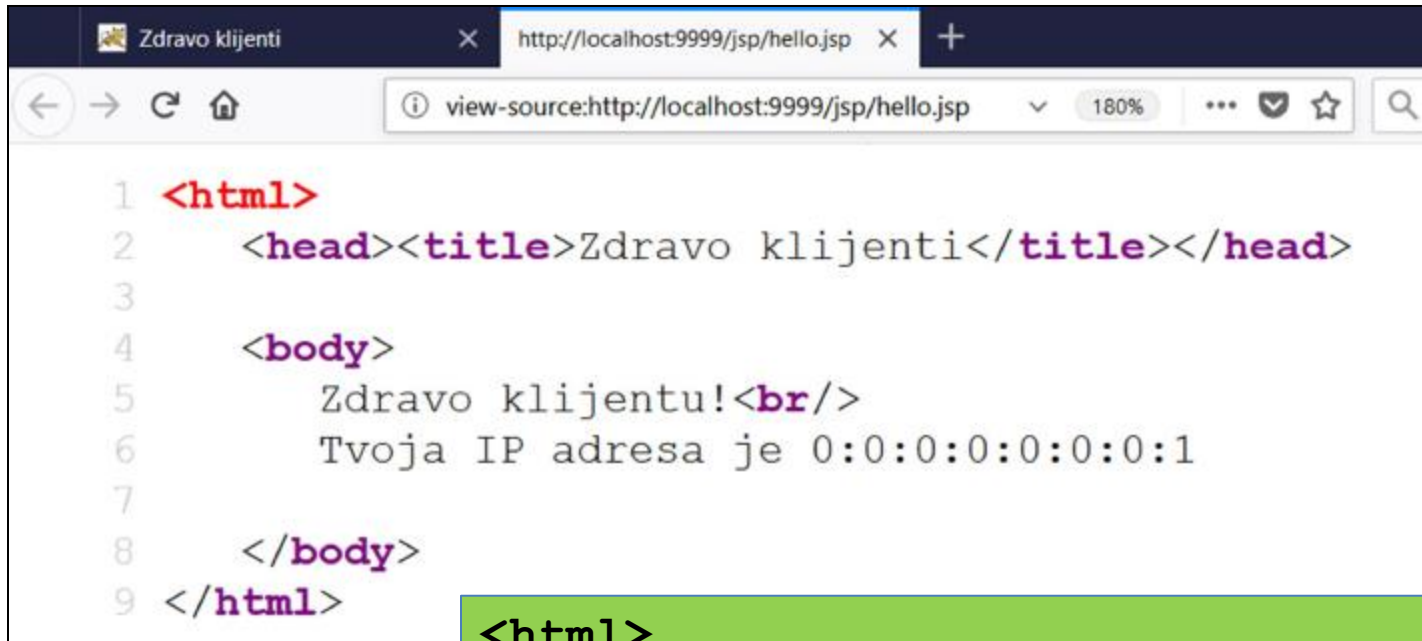
```
<%  
    out.println("Tvoja IP adresa je " + request.getRemoteAddr());  
%>
```

Разлике сервлет и Java Server Pages

- Разлике JSP у односу на сервлете:
 - није потребно правити класу
 - није потребно компајлирање
 - једноставно се убацује код између "<%>" и "<%>"
 - екстензија фајла је ".jsp" (није html)
- У прегледач се уноси адреса
"http://localhost:9999/" где
уместо праве IP адресе стоји локална адреса када је сервер
на истом рачунару на коме је и клијент, "localhost",
- ":9999" означава софтверски порт као и код сервлета
(подешено при инсталацији сервера на клијентски рачунар)

Једностави JSP пример

html у прегледача који је направио фајл на серверу "hello.jsp"



```
1 <html>
2   <head><title>Zdravo klijenti</title></head>
3
4   <body>
5     Zdravo klijentu!<br/>
6     Tvoja IP adresa je 0:0:0:0:0:0:0:1
7
8   </body>
9 </html>
```

```
<html>
  <head><title>Zdravo klijenti</title></head>
  <body>
    Zdravo klijentu!<br/>
    Tvoja IP adresa je 0:0:0:0:0:0:0:1
  </body>
</html>
```

Процесирање JSP

- Клијентски рачунар нема могућност да извршава или интерпретира JSP код, који има екстензију ".jsp":
 1. Као и са нормалном веб страницом, клијентов прегледач шаље *HTTP* упит на веб сервер.
 2. Веб сервер препознаје да је *HTTP* захтев за *JSP* страницу и прослеђује га софтверу који зна шта треба да се ради са страницама које имају екстензију ".jsp" а не екстензију ".html" (тај део софтвера се на енглеском назива *JSP engine*).
Сервер зна шта да ради са фајловима типа ".jsp", а клијент не зна шта да ради са фајловима ".jsp"
 3. *JSP engine* учитава *JSP* страницу са диска сервера и претвара га у садржај сервлета. Ова конверзија је врло једноставна; сви текстови се претварају у `println()` наредбе, а сви *JSP* елементи се конвертују у Јава код. Овај код имплементира одговарајуће динамичко понашање странице.

Процесирање JSP

4. JSP engine компајлира сервлет у извршну Јава класу и прослеђује оригинални захтев софтверу који извршава сервлете (све ово се дешава на серверу).
5. Софтвер, који извршава сервлете на серверу, учитава класе сервлета и извршава их. Током извршења, сервлет производи поруку у HTML формату. Ова порука се даље прослеђује веб серверу који је уноси у HTTP одговор.
6. Веб сервер прослеђује HTTP одговор клијентовом прегледачу као да се ради о статичком HTML садржају.
7. Динамички се генерише садржај на страни сервера, и тај садржај прослеђује клијенту. Одговор од сервера који стигне у клијентов прегледач, који се обрађује на клијентској страни и приказује као да је садржај статичке веб стране.

Процесирање JSP

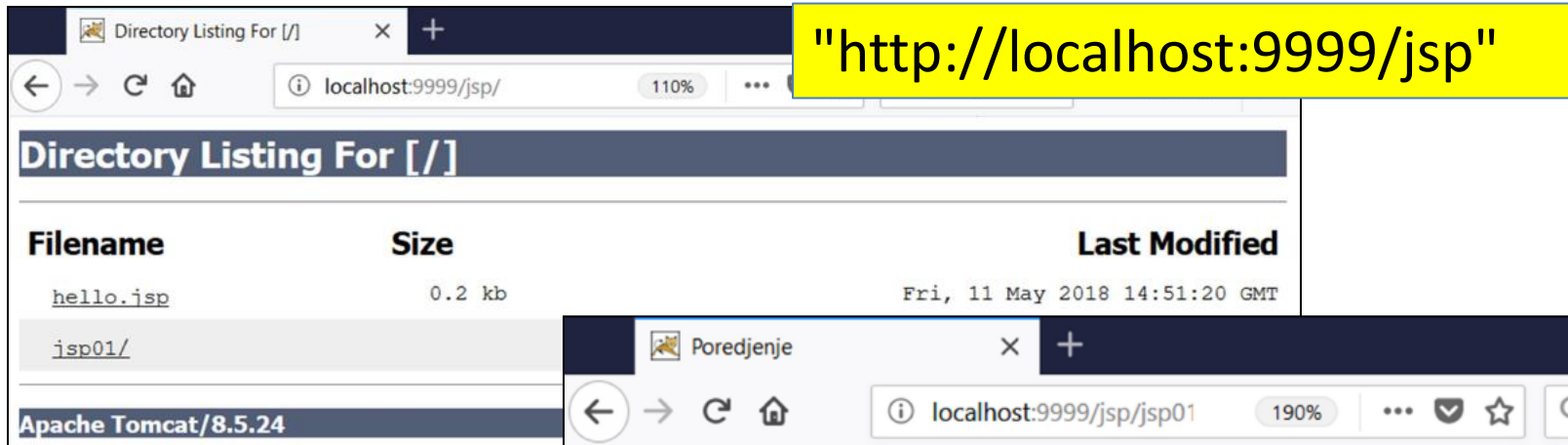
- Типично, *JSP engine* проверава да ли постоји сервлет за *JSP* датотеку и да ли је датум модификације на *JSP* старији од сервлета. Ако је *JSP* старији од свог генерисаног сервлета, *JSP* контејнер претпоставља да се *JSP* није променио и да генерисани сервлет и даље одговара садржају *JSP*. Ово чини процес ефикаснијим него код других скриптних језика (као што је *PHP*) па самим тим и брже се извршава.
- *JSP* страница је само још један начин писања сервлета без потребе да програмер буде експерт за Јава програмирање. *JSP* страница се обрађује исто као и регуларни сервлет, осим потребе да програмер извршава компајлирање.

Задатак 1 из збирке задатака (JSP)

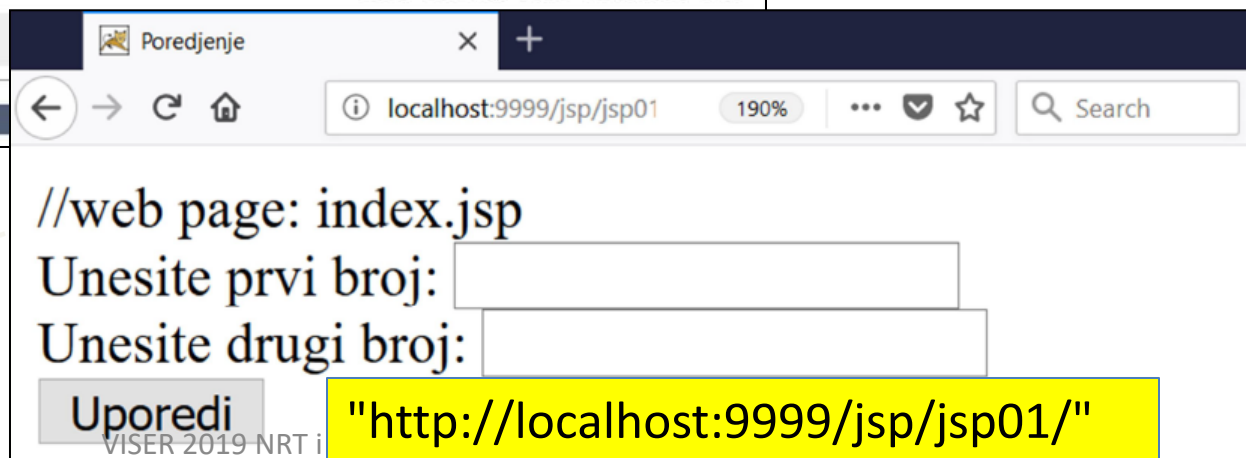
- Коришћењем Java Server Pages технологије написати JSP страну која преко HTML форме прихвата два броја, прослеђује их другој JSP страни и упоређује их:
 - Уколико је први број мањи од другог, треба исписати "Razlika je manja od nule" и обојити позадину у плаву боју
 - Уколико је први број већи од другог, треба исписати "Razlika je veća od nule" и обојити позадину у зелену боју
 - Уколико су бројеви исти, треба исписати "Razlika je nula" и обојити позадину у жуту боју
 - У случају грешке у уносу бројева, на црвеној позадини написати "Greska!" и исписати врсту грешке

Унос два броја (JSP)

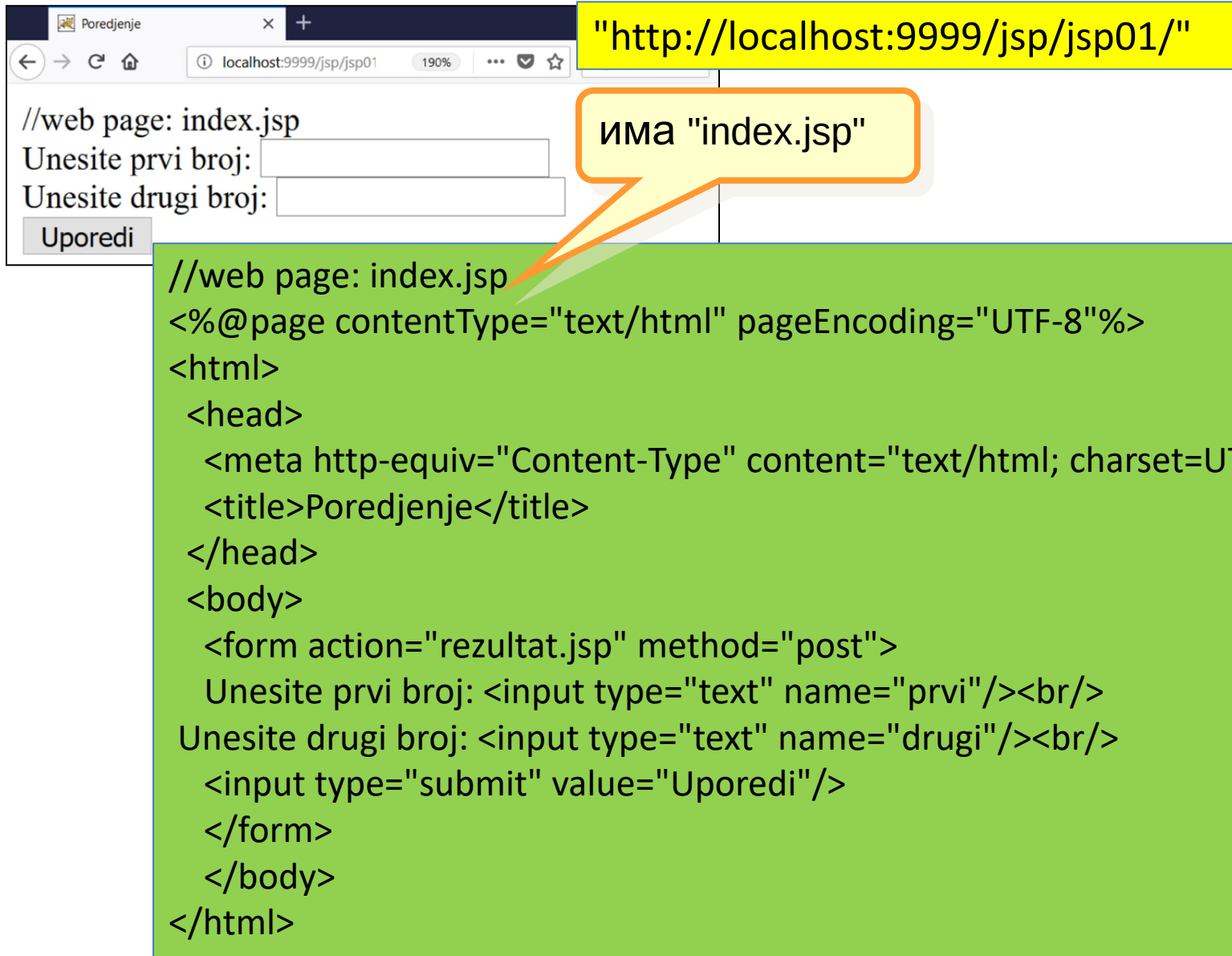
- Унесе се адреса у прегледачу "http://localhost:9999/jsp" без навођења, "hello.jsp", а нема ни "index.html"
- "http://localhost:9999/jsp/jsp01/" постоји "index.jsp", али нема "index.html"



има "index.jsp"



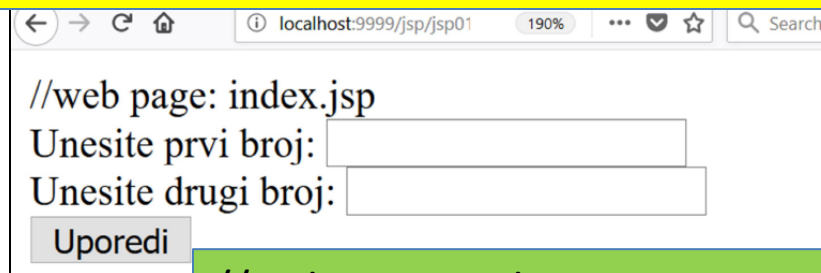
index.jsp (JSP)



The image shows a web browser window with the title "Poredjenje" and the address bar displaying "localhost:9999/jsp/jsp01". The page content includes the text "//web page: index.jsp", two input fields labeled "Unesite prvi broj:" and "Unesite drugi broj:", and a button labeled "Uporedi". A yellow callout box points to the address bar with the text "http://localhost:9999/jsp/jsp01/". Another yellow callout box points to the first input field with the text "има 'index.jsp'". Below the browser window, the source code of the JSP page is displayed on a green background.

```
//web page: index.jsp
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Poredjenje</title>
  </head>
  <body>
    <form action="rezultat.jsp" method="post">
      Unesite prvi broj: <input type="text" name="prvi"/><br/>
      Unesite drugi broj: <input type="text" name="drugi"/><br/>
      <input type="submit" value="Uporedi"/>
    </form>
  </body>
</html>
```

"http://localhost:9999/jsp/jsp01/"



localhost:9999/jsp/jsp01 190% Search

//web page: index.jsp

Unesite prvi broj:

Unesite drugi broj:

Uporedi

сервер прихвата "index.jsp"
као да стоји "index.html" у адреси
сервер шаље прегледачу *html* фајл
као да је у питању статичка веб страна.

```
//web page: index.jsp
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Poredjenje</title>
  </head>
  <body>
    <form action="rezultat.jsp" method="post">
      Unesite prvi broj: <input type="text" name="prvi"/><br/>
      Unesite drugi broj: <input type="text" name="drugi"/><br/>
      <input type="submit" value="Uporedi"/>
    </form>
  </body>
</html>
```

"http://localhost:9999/jsp/jsp01/"

```
//web page: rezultat.jsp
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!-- direktiva -->
<%@ page errorPage="error.jsp" %>
<!-- deklaracija -->
<%! int a, b, razlika;
    String prvi, drugi;
%>
<!-- skriptet -->
<%
    prvi = request.getParameter("prvi");
    drugi = request.getParameter("drugi");
    a = Integer.parseInt(prvi);
    b = Integer.parseInt(drugi);
    razlika = a - b;
%>
```

```
<body>
  <form action="rezultat.jsp" method="post">
    Unesite prvi broj: <input type="text" name="prvi"/><br/>
    Unesite drugi broj: <input type="text" name="drugi"/><br/>
    <input type="submit" value="Uporedi"/>
  </form>
</body>
```

```

<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>
      <%if (razlika < 0) {%>
        Manje
      <%} else if (razlika > 0) {%>
        Veće
      <%} else {%>
        Nula
      <%}%>
    </title>
  </head>

```

```

    <form action="rezultat.jsp" method="post">
      Unesite prvi broj: <input type="text" name="prvi"/><br/>
      Unesite drugi broj: <input type="text" name="drugi"/><br/>
      <input type="submit" value="Uporedi"/>
    </form>

```

```

<body bgcolor="<%if (razlika < 0) {%>
  blue
<%} else if (razlika > 0) {%>
  green
<%} else {%>
  yellow
<%} %>">

```



```

<h2>
    Razlika je
    <%if (razlika < 0) {%>
        manja od nule.
    <%} else if (razlika > 0) {%>
        veća od nule.
    <%} else {%>
        nula.
    <%}%>
</h2>
</body>
</html>

```

```

<form action="rezultat.jsp" method="post">
    Unesite prvi broj: <input type="text" name="prvi"/><br/>
    Unesite drugi broj: <input type="text" name="drugi"/><br/>
    <input type="submit" value="Uporedi"/>
</form>

```

//web page: error.jsp

```

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@page isErrorPage="true" %>
<html>
    <head>

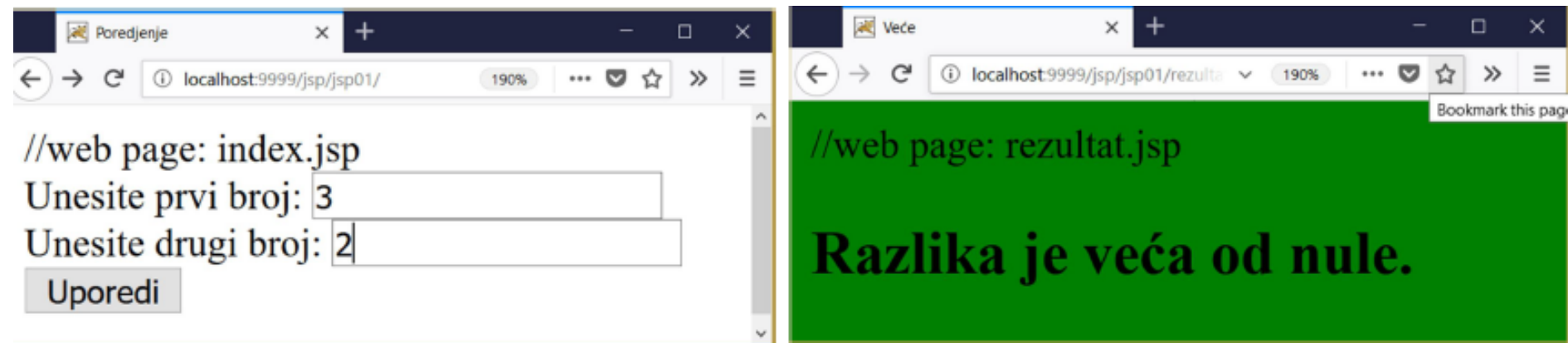
```

```

        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Error Page</title>
    </head>
    <body bgcolor=red>
        <h2>Greska!</h2>
        <h3><%=exception%></h3>
    </body>
</html>

```

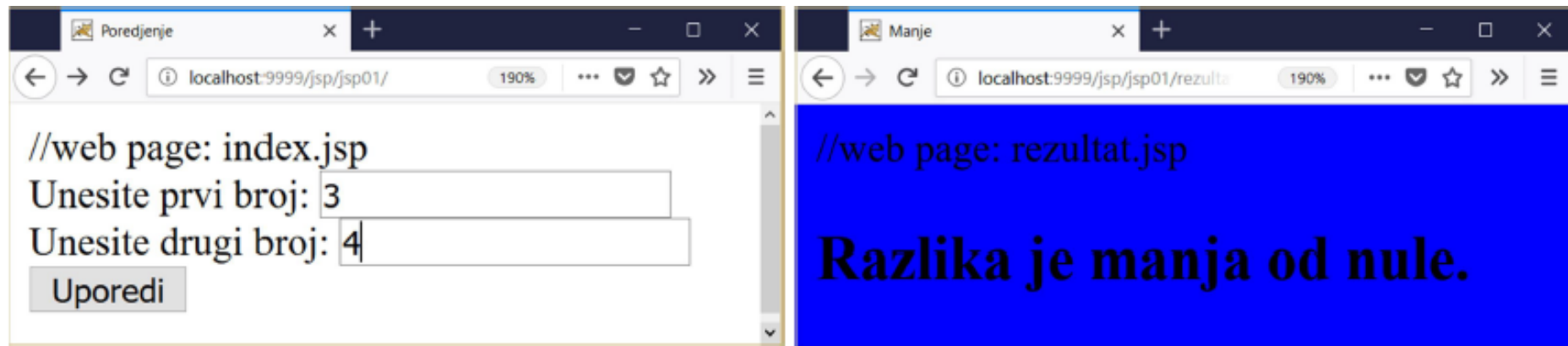
Тестирање



Унос података у "http://localhost:9999/jsp/jsp01/"

приказ у "http://localhost:9999/jsp/jsp01/rezultat.jsp"

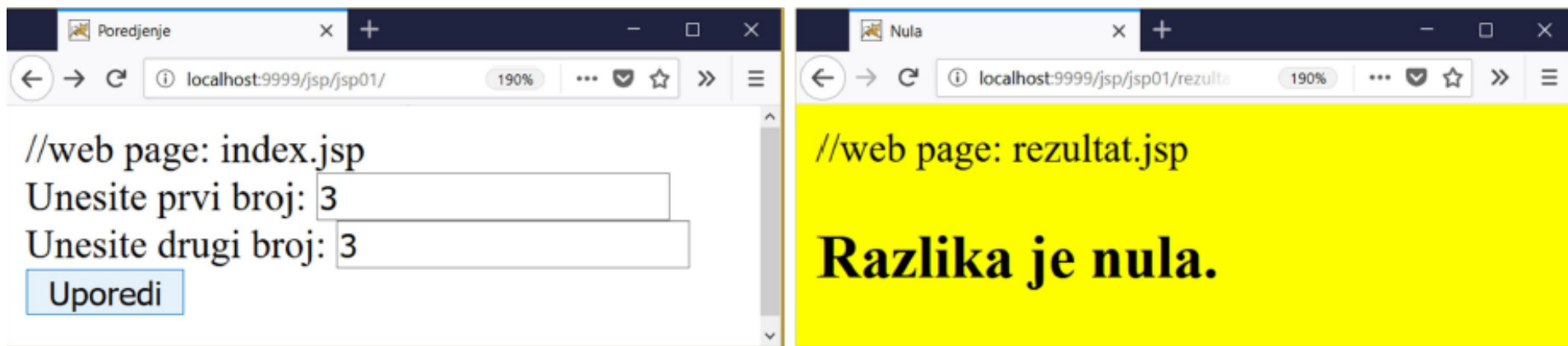
Тестирање



Унос података у "http://localhost:9999/jsp/jsp01/"

приказ у "http://localhost:9999/jsp/jsp01/rezultat.jsp"

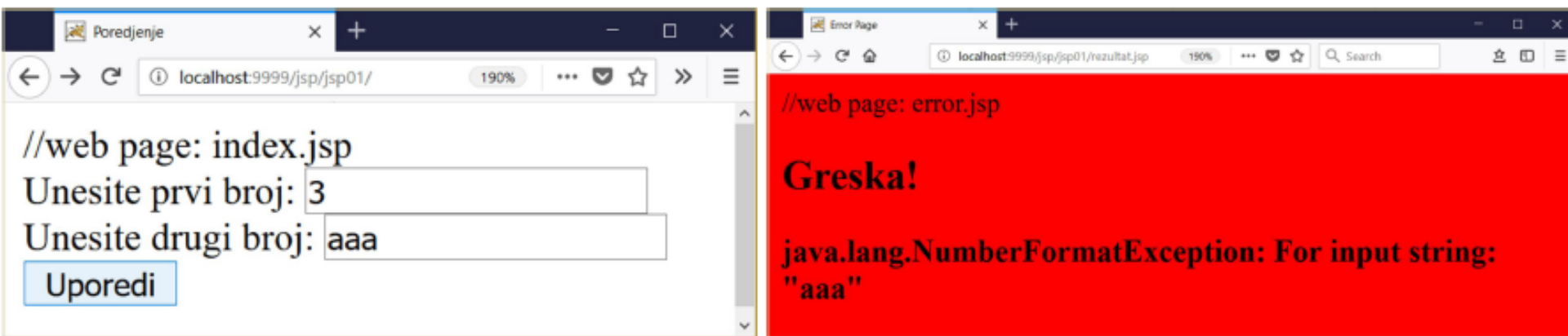
Тестирање



Унос података у "http://localhost:9999/jsp/jsp01/"

приказ у "http://localhost:9999/jsp/jsp01/rezultat.jsp"

Тестирање



Унос података у "http://localhost:9999/jsp/jsp01/"

приказ у "http://localhost:9999/jsp/jsp01/rezultat.jsp"

Уколико се уради измена у било ком фајлу, тада се измена може радити док се тестира апликација, и одмах након измене се добије резултат са измењеним кодом. Не треба да се ресетује сервер, или класа компајлира

"index.jsp"

```
//web page: index.jsp
```

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
```

```
<html>
```

```
<head>
```

```
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
```

```
<title>Poredjenje</title>
```

```
</head>
```

```
<body>
```

```
<form action="rezultat.jsp" method="post">
```

```
Unesite prvi broj: <input type="text" name="prvi"/><br/>
```

```
Unesite drugi broj: <input type="text" name="drugi"/><br/>
```

```
<input type="submit" value="Uporedi"/>
```

```
</form>
```

```
</body>
```

```
</html>
```

"rezultat.jsp"

```
//web page: rezultat.jsp
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!-- direktiva -->
<%@ page errorPage="error.jsp" %>
<!-- deklaracija -->
<%! int a, b, razlika;
    String prvi, drugi;
%>
<!-- skriptni -->
<%
    prvi = request.getParameter("prvi");
    drugi = request.getParameter("drugi");
    a = Integer.parseInt(prvi);
    b = Integer.parseInt(drugi);
    razlika = a - b;
%>
```

```

<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>
      <%if (razlika < 0) {%>
        Manje
      <%} else if (razlika > 0) {%>
        Veće
      <%} else {%>
        Nula
      <%}%>
    </title>
  </head>
  <body bgcolor="<%if (razlika < 0) {%>
    blue
  <%} else if (razlika > 0) {%>
    green
  <%} else {%>
    yellow
  <%} %>">
    <h2>
      Razlika je
      <%if (razlika < 0) {%>
        manja od nule.
      <%} else if (razlika > 0) {%>
        veća od nule.
      <%} else {%>
        nula.
      <%}%>
    </h2>
  </body>
</html>

```

"rezultat.jsp"

"error.jsp"

```
//web page: error.jsp
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@page isErrorPage="true" %>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Error Page</title>
  </head>
  <body bgcolor=red>
    <h2>Greska!</h2>
    <h3><%=exception%></h3>
  </body>
</html>
```

JSP и MVC сервлети

- ip **12** PT&SCHE.pdf JSP и MVC сервлети,
решен задатак 2 из области JSP из збирке
корак-по-корак

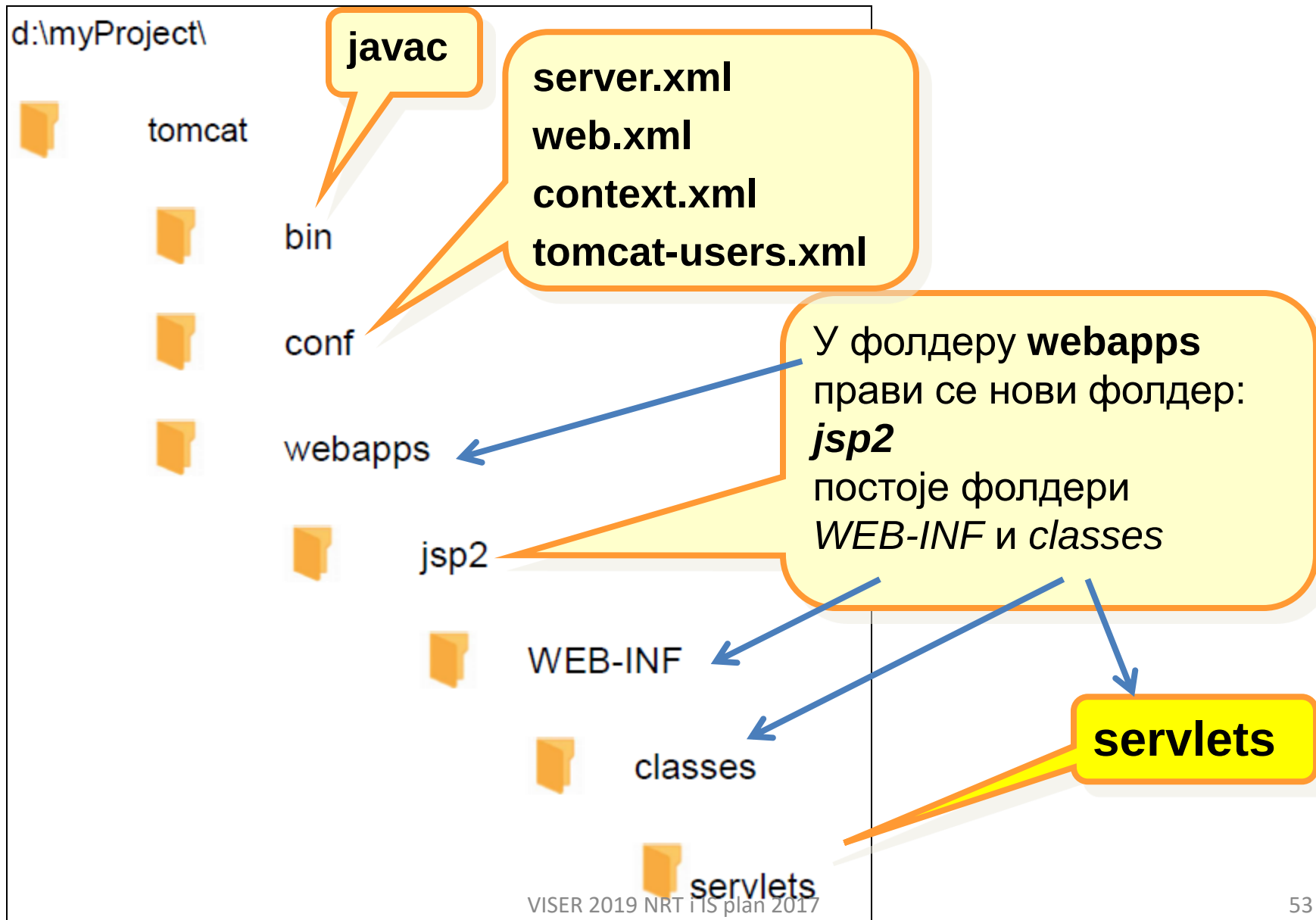
Model View Controller (MVC)

- Израда веб апликације у којој се користе основни сервлети и *JSP – Model View Controller (MVC)*
- Састоји се у томе да се за презентацију (приказ на клијентском рачунару) користи *JSP*, а за израчунавања се користи класичан сервлет
- Да се мења у презентацији – користи се *JSP*, како би се избегло компајлирање и што пре урадила апликација
- Направи се сервлет коришћењем Јава класе, компајлирањем, и нема потребе да се било шта више мења

Model View Controller (MVC)

- Структура директоријума – бира се назив апликације, "jsp2", фолдер "jsp2" се налази испод "webapps« а где се налазе све апликације на серверу
- Остала структура као за класичан сервлет
- "classes", има још један поддиректоријум произвољно назван "servlets", у који треба да буду смештене све јава класе
- Јава класе треба да садрже онај део функционалности који нешто израчунава и који не треба често да се мења

Избор назива фолдера и фајлова, структура на серверу

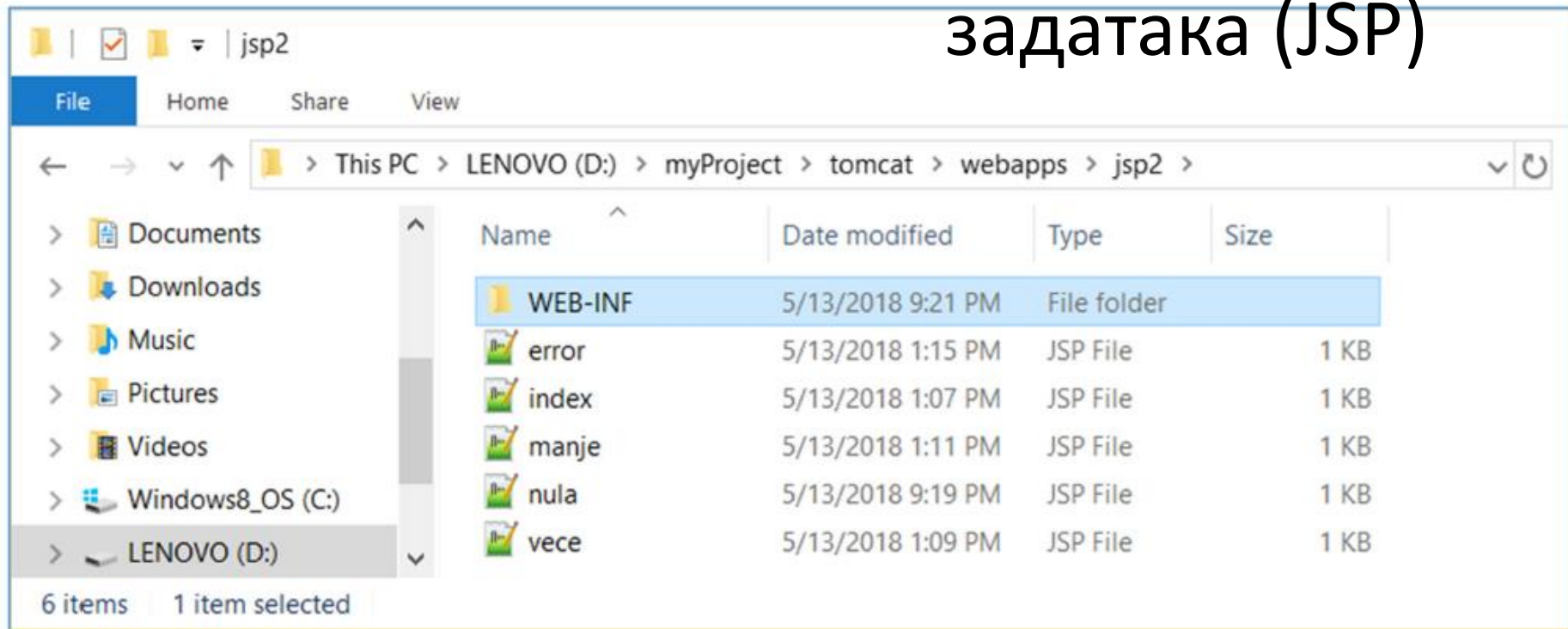


Задатак 2 из збирке задатака (JSP)

- Коришћењем Java Server Pages технологије написати JSP стране које преко HTML форме прихватају два броја, прослеђују их Јава сервлету, који их упоређује и генерише приказе у складу са резултатом обраде
 - ✓ Уколико је први број **већи** од другог, сервлет преусмерава на страну "vese.jsp" и обоји позадину у зелену боју
 - ✓ Уколико је први број **мањи** од другог, сервлет преусмерава на страну "manje.jsp" и обоји позадину у плаву боју
 - ✓ Уколико је разлика **једнака** 0, сервлет преусмерава на страну "nula.jsp" и обоји позадину у жуту боју
 - ✓ У случају **грешке** у уносу бројева, на црвеној позадини написати "Greska!"
- Све провере треба реализовати на сервлету

Фајлови у фолдер "jsp2" се налази одмах испод "webapps"

Задатак 2 из збирке задатака (JSP)



"index.jsp":

Задатак 2 из збирке задатака (JSP)

"index.jsp":

```
//web page: index.jsp
<%@page contentType="text/html" pageEncoding="UTF-8"%>

<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>MVC prvi primer</title>
  </head>
  <body>
    <h2>MVC pristup</h2>
    <h4>Model View Controller (MVC) design pattern,
      using basic Servlets and JSPs</h4>
    <p>//http://localhost:9999/jsp/jsp02/</p>

    <form action="Uporedi" method="post">
      Unesite prvi broj: <input type="text" name="prvi"/><br/>
      Unesite drugi broj: <input type="text" name="drugi"/><br/>
      <input type="submit" value="Uporedi"/>
    </form>
  </body>
</html>
```


"error.jsp":

Задатак 2 из збирке задатака (JSP)

"error.jsp":

//web page: error.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
```

```
<html>
```

```
  <head>
```

```
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
```

```
    <title>Error Page</title>
```

```
  </head>
```

```
  <body bgcolor="#ff0000">
```

```
    <h3>Greska!</h3>
```

```
  </body>
```

```
</html>
```

"manje.jsp":

Задатак 2 из збирке задатака (JSP)

"manje.jsp":

```
//web page: manje.jsp
```

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
```

```
<html>
```

```
  <head>
```

```
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
```

```
    <title>Manje</title>
```

```
  </head>
```

```
  <body bgcolor="#0000ff">
```

```
    <h3>Razlika je manja od nule.</h3>
```

```
  </body>
```

```
</html>
```

"nula.jsp":

Задатак 2 из збирке задатака (JSP)

"nula.jsp":

```
//web page: nula.jsp
```

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
```

```
<html>
```

```
  <head>
```

```
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
```

```
    <title>Nula</title>
```

```
  </head>
```

```
  <body bgcolor="#ffff00">
```

```
    <h3>Razlika je nula.</h3>
```

```
  </body>
```

```
</html>
```

"vece.jsp":

Задатак 2 из збирке задатака (JSP)

"vece.jsp":

//web page: vece.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
```

```
<html>
```

```
  <head>
```

```
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
```

```
    <title>Vece</title>
```

```
  </head>
```

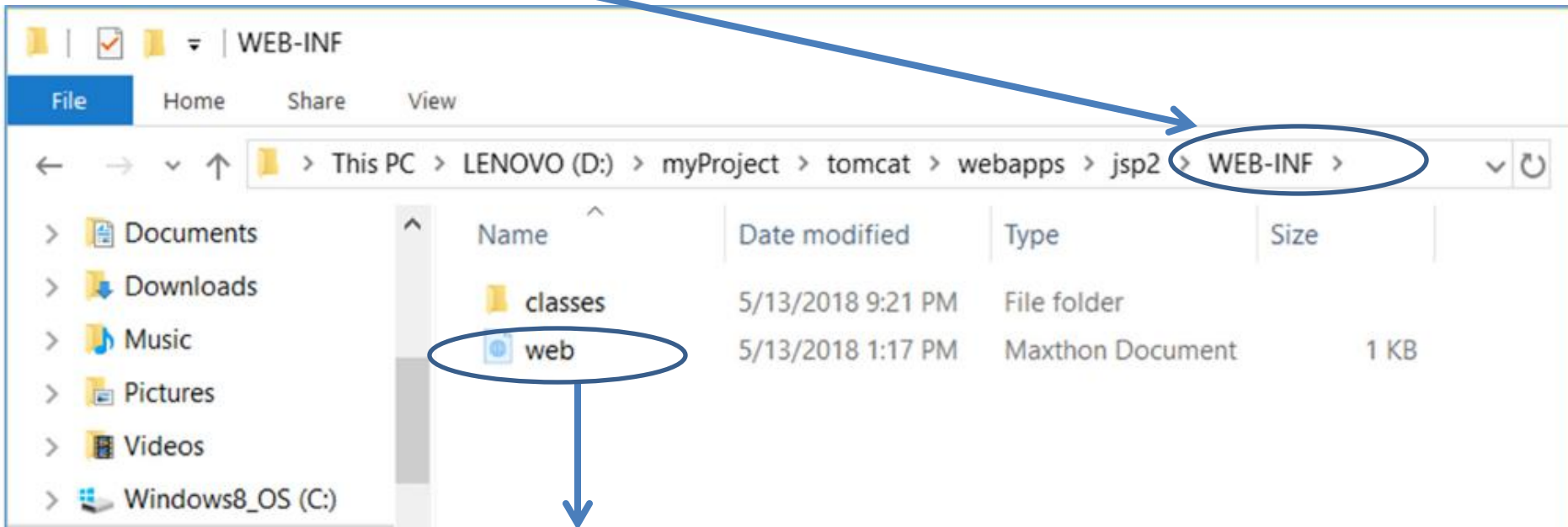
```
  <body bgcolor="#00ff00">
```

```
    <h3>Razlika je veka od nule.</h3>
```

```
  </body>
```

```
</html>
```

У фолдеру "WEB-INF" су фајлови са сервлетима; "web.xml"



```
<?xml version="1.0" encoding="ISO-8859-1"?>
<web-app version="3.0"
  xmlns="http://java.sun.com/xml/ns/javaee"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
http://java.sun.com/xml/ns/javaee/web-app_3_0.xsd">

  <!-- To save as "...\\WEB-INF\\web.xml" -->
```

"web.xml"

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<web-app version="3.0"
  xmlns="http://java.sun.com/xml/ns/javaee"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
http://java.sun.com/xml/ns/javaee/web-app_3_0.xsd">
```

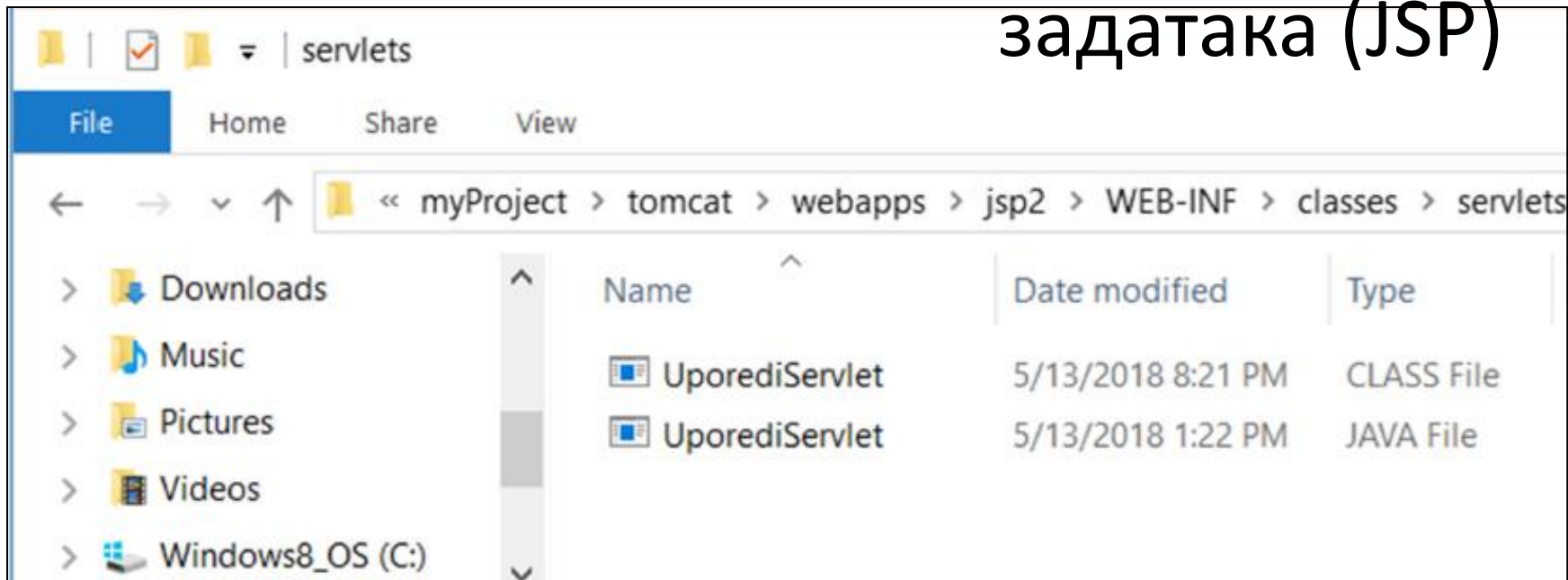
```
  <!-- To save as "...\\WEB-INF\\web.xml" -->
```

```
  <servlet>
    <servlet-name>Uporedi</servlet-name>
    <servlet-class>servlets.UporediServlet</servlet-class>
  </servlet>
  <servlet-mapping>
    <servlet-name>Uporedi</servlet-name>
    <url-pattern>/Uporedi</url-pattern>
  </servlet-mapping>
  <session-config>
    <session-timeout>
      30
    </session-timeout>
  </session-config>
  <welcome-file-list>
    <welcome-file>index.jsp</welcome-file>

  </welcome-file-list>
</web-app>
```

"UporediServlet.java"

Задатак 2 из збирке задатака (JSP)



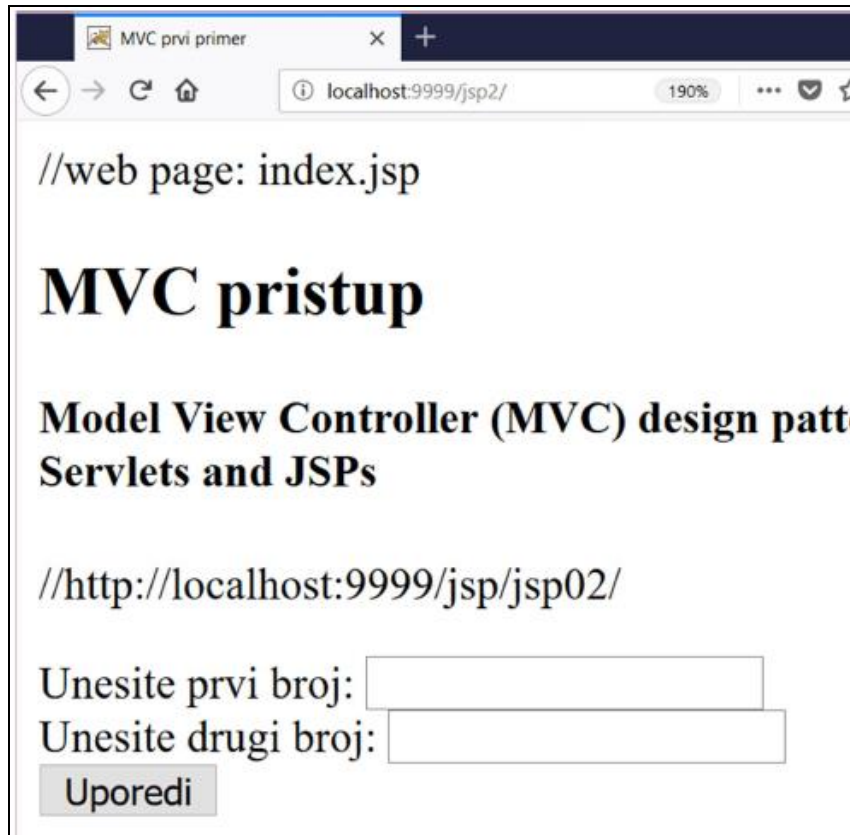
```
package servlets;  
import java.io.*;  
import javax.servlet.*;  
import javax.servlet.http.*;
```

```
public class UporediServlet extends HttpServlet {  
    @Override  
    protected void doPost(HttpServletRequest request,  
                           HttpServletResponse response)  
        throws ServletException, IOException {  
        String prvi = request.getParameter("prvi");  
        String drugi = request.getParameter("drugi");  
        String address;  
        int a, b;  
        try {  
            a = Integer.parseInt(prvi);  
            b = Integer.parseInt(drugi);  
            int razlika = a - b;  
            if (razlika > 0) {  
                address = "/vece.jsp";  
            } else if (razlika < 0) {  
                address = "/manje.jsp";  
            } else {  
                address = "/nula.jsp";  
            }  
        } catch (NumberFormatException nfe) {  
            address = "/error.jsp";  
        }  
        RequestDispatcher dispatcher =  
request.getRequestDispatcher(address);  
        dispatcher.forward(request, response);  
    }  
}
```

Задатак 2 из збирке задатака (JSP)

"UporediServlet.java"

Задатак 2 из збирке задатака (JSP)



Стартује се локални сервер и унесе адреса у клијентовом прегледачу "http://localhost:9999/jsp2".

Сервер добија адресу од клијента, проналази фолдер "jsp2" у фолдеру у коме су апликације, "webapps"

У "jsp2" нема "index.html" али има "index.jsp"

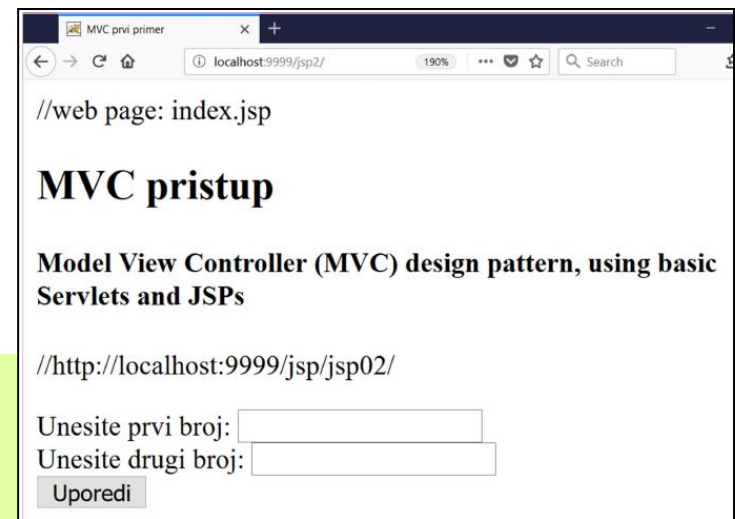
Сервер извршава "index.jsp", и шаље клијентовом прегледачу садржај који клијент види

Задатак 2 из збирке задатака (JSP)

//web page: index.jsp

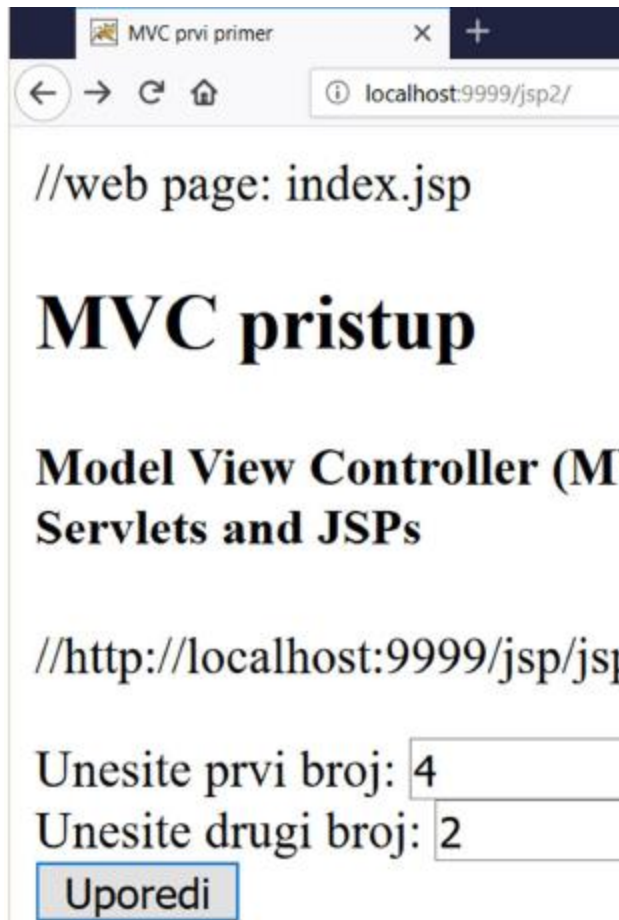
```
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>MVC prvi primer</title>
  </head>
  <body>
    <h2>MVC pristup</h2>
    <h4>Model View Controller (MVC) design pattern,
      using basic Servlets and JSPs</h4>
    <p>//http://localhost:9999/jsp/jsp02/</p>

    <form action="Uporedi" method="post">
      Unesite prvi broj: <input type="text" name="prvi"/><br/>
      Unesite drugi broj: <input type="text" name="drugi"/><br/>
      <input type="submit" value="Uporedi"/>
    </form>
  </body>
</html>
```



Задатак 2 из збирке задатака (JSP)

После уноса, клика "Uporedi", прегледач додаје израз `"/Uporedi"` у адресу `"http://localhost:9999/jsp2"`, адреса је `"http://localhost:9999/jsp2/Uporedi"`. Сервер проверава у `"WEB-INF"` у `"web.xml"` шта треба да ради, проналази да се сервлет мапира на јава класу која је у `"classes"`, у фолдеру `"servlets"`



MVC prvi primer

localhost:9999/jsp2/

//web page: index.jsp

MVC pristup

Model View Controller (MVC) design pattern, using basic Servlets and JSPs

//http://localhost:9999/jsp/jsp02/

Unesite prvi broj: 4

Unesite drugi broj: 2

Uporedi

проналази класу за израчунавање `"/servlets/UporediServlet.class"`. У зависности од резултата, сервер позива JSP фајлове `"error.jsp"`, `"manje.jsp"`, `"nula.jsp"` или `"vece.jsp"`

Унос података у "http://localhost:9999/jsp2"

Тестирање

//web page: index.jsp

MVC pristup

Model View Controller (MVC) design pattern, using basic Servlets and JSPs

//http://localhost:9999/jsp/jsp02/

Unesite prvi broj:

Unesite drugi broj:

приказ "http://localhost:9999/jsp2/Uporedi"

Vece

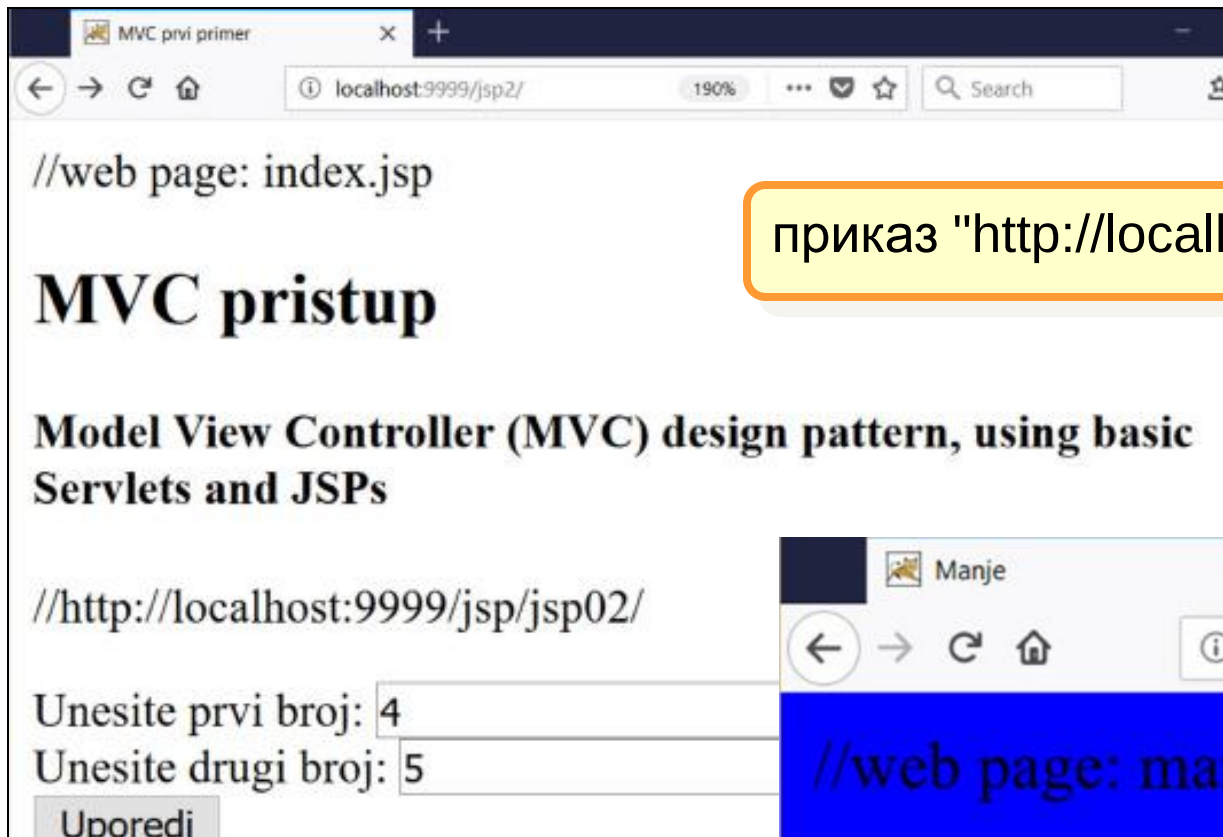
localhost:9999/jsp2/Uporedi

//web page: vece.jsp

Razlika je veca od nule.

Унос података у "http://localhost:9999/jsp2"

Тестирање



The screenshot shows a web browser window titled "MVC prvi primer". The address bar displays "localhost:9999/jsp2/". The page content includes a comment "//web page: index.jsp", the heading "MVC pristup", and a description "Model View Controller (MVC) design pattern, using basic Servlets and JSPs". Below this, there is a comment "//http://localhost:9999/jsp/jsp02/". The form contains two input fields: "Unesite prvi broj:" with the value "4" and "Unesite drugi broj:" with the value "5". At the bottom of the form is a button labeled "Uporedi".

//web page: index.jsp

MVC pristup

Model View Controller (MVC) design pattern, using basic Servlets and JSPs

//http://localhost:9999/jsp/jsp02/

Unesite prvi broj: 4

Unesite drugi broj: 5

Uporedi

приказ "http://localhost:9999/jsp2/Uporedi"



The screenshot shows a web browser window titled "Manje". The address bar displays "localhost:9999/jsp2/Uporedi". The page content includes a comment "//web page: manje.jsp" and the text "Razlika je manja od nule." in a large, bold font.

//web page: manje.jsp

Razlika je manja od nule.

Унос података у "http://localhost:9999/jsp2"

Тестирање

The screenshot shows a web browser window with the title "MVC prvi primer". The address bar displays "localhost:9999/jsp2/". The page content includes a comment "//web page: index.jsp", the heading "MVC pristup", and a description "Model View Controller (MVC) design pattern, using basic Servlets and JSPs". Below this is another comment "//http://localhost:9999/jsp/jsp02/". At the bottom, there are two input fields: "Unesite prvi broj:" with the value "4" and "Unesite drugi broj:" with the value "4". A blue button labeled "Uporedi" is positioned below the second input field.

приказ "http://localhost:9999/jsp2/Uporedi"

The screenshot shows a web browser window with the title "Nula". The address bar displays "localhost:9999/jsp2/Uporedi". The page content is on a yellow background and includes a comment "//web page: nula.jsp" and the text "Razlika je nula." in a large, bold font.

Унос података у "http://localhost:9999/jsp2"

Тестирање

MVC prvi primer

localhost:9999/jsp2/ 190%

//web page: index.jsp

MVC pristup

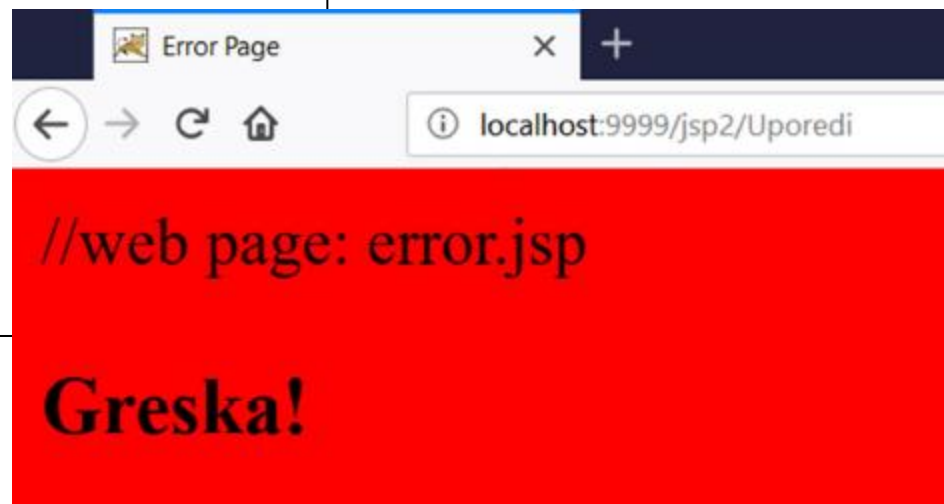
Model View Controller (MVC) design pattern, using basic Servlets and JSPs

//http://localhost:9999/jsp/jsp02/

Unesite prvi broj:

Unesite drugi broj:

приказ "http://localhost:9999/jsp2/Uporedi"



Profesor dr Miroslav Lutovac
mlutovac@viser.edu.rs

Ova prezentacija je nekomercijalna.

Slajdovi mogu da sadrže materijale preuzete sa Interneta, stručne i naučne građe, koji su zaštićeni Zakonom o autorskim i srodnim pravima.

Ova prezentacija se može koristiti samo privremeno tokom usmenog izlaganja nastavnika u cilju informisanja i upućivanja studenata na dalji stručni, istraživački i naučni rad i u druge svrhe se ne sme koristiti –

Član 44 - Dozvoljeno je bez dozvole autora i bez plaćanja autorske naknade za nekomercijalne svrhe nastave:

(1) javno izvođenje ili predstavljanje objavljenih dela u obliku neposrednog poučavanja na nastavi;

- ZAKON O AUTORSKOM I SRODNIM PRAVIMA

("Sl. glasnik RS", br. 104/2009 i 99/2011)