Rešeni zadaci iz programskog jezika

Java Servlet i JavaServerPages

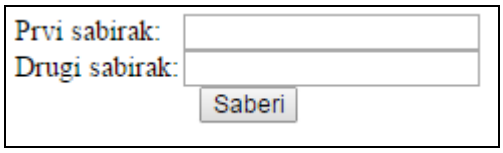
Studenti koji uvide greške u ovim materijalima, imaju neke primedbe, predloge, pohvale ili na drugi način žele da pomognu u pripremi materijala za ovaj kurs, mogu se javiti na e-mail:

drazen.draskovic@etf.bg.ac.rs

/verzija 10.01.2015./

Primer 1 - Aritmetička operacija sabiranja

Korišćenjem Java Servlet tehnologije napisati servlet koji vrši sabiranje dva broja. Unos podataka treba izvršiti kroz HTML formu.



Rešenje:

```
//web page: index.html

<!DOCTYPE html>
<html>
    <head>
        <title></title>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    </head>
    <body>
        <form action="pozoviServlet" method="POST">
            <table>
                <tr>
                    <td>Prvi sabirak:</td>
                    <td><input type="text" name="prvi" value="" /> </td>
                </tr>
                <tr>
                    <td>Drugi sabirak:</td>
                    <td><input type="text" name="drugi" value="" /> </td>
                </tr>
                <tr>
                    <td colspan="2" align="center">
                        <input type="submit" value="saberi" />
                    </td>
                </tr>
            </table>
        </form>

    </body>
</html>
```

```
//source file: PrviServlet.java

import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class PrviServlet extends HttpServlet {
    //metoda za obradu i HTTP GET i HTTP POST zahteva
    protected void processRequest(HttpServletRequest request,
                                  HttpServletResponse response)
    throws ServletException, IOException {
```

```
        response.setContentType("text/html;charset=UTF-8");
        PrintWriter out = response.getWriter();
        String prvi=request.getParameter("prvi");
        String drugi=request.getParameter("drugi");
        int a=0, b=0;
        boolean ispravno=true;
        double c=0.0;

        try{
            a=Integer.parseInt(prvi);
            b=Integer.parseInt(drugi);
            c = Double.parseDouble(prvi);
        }catch(NumberFormatException nfe){ispravno=false;}

        int zbir=a+b;

        try {
            out.println("<html>");
            out.println("<head>");
            out.println("<title>Primer2</title>");
            out.println("</head>");
            if(ispravno){
             out.println("<body bgcolor=\"00ffff\">");
             out.println("<h1>Zbir je: " + zbir + "</h1>");
            }
            else{
             out.println("<body bgcolor=\"ff0000\">");
             out.println("<h1>Neispravni brojevi!</h1>");
            }
            out.println("</body>");
            out.println("</html>");
        } finally {
            out.close();
        }
    }

    protected void doGet(HttpServletRequest request,
                         HttpServletResponse response)
    throws ServletException, IOException {
        processRequest(request, response);
    }

    protected void doPost(HttpServletRequest request,
                          HttpServletResponse response)
    throws ServletException, IOException {
        processRequest(request, response);
    }

    public String getServletInfo() {
        return "Short description";
    }
}
```

//XML file: WEB-INF/web.xml
//fajl koji generise okruzenje Netbeans prilikom kreiranja servleta

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app version="2.5" 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_2_5.xsd">
    <servlet>
        <servlet-name>imeServleta</servlet-name>
        <servlet-class>PrviServlet</servlet-class>
    </servlet>
    <servlet-mapping>
        <servlet-name>imeServleta</servlet-name>
        <url-pattern>/pozoviServlet</url-pattern>
    </servlet-mapping>
    <session-config>
        <session-timeout>
            30
        </session-timeout>
    </session-config>
    <welcome-file-list>
        <welcome-file>index.html</welcome-file>
    </welcome-file-list>
</web-app>
```

Primer 2 - Pozivanje servleta iz drugog servleta

Korišćenjem Java Servlet tehnologije napisati HTML stranu koja preko linka poziva prvi servlet, koji zatim vrši prosleđivanje na drugi servlet, korišćenjem metode sendRedirect. Na drugom servletu napraviti formu koja ima tekstualno polje za unos imena korisnika i dugme za potvrdu, kojim se poziva treći servlet. Na trećem servletu pročitati podatak (GET parametar), koji je prosleđen tom servletu od strane drugog.

Rešenje:

```
//web page: index.html

<!DOCTYPE html>
<html>
    <head>
        <title>Primer sa prosledjivanjem izmedju servleta</title>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    </head>
    <body>
        <a href="prviServlet">Kliknite na link da pozovete prvi servlet!</a>
        <!-- pozivom servleta preko linka poziva se doGet() metod-->
    </body>
</html>
```

```
//source file: servleti/PrviServlet.java

package servleti;

import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
public class PrviServlet extends HttpServlet {

    protected void doGet(HttpServletRequest request,
                         HttpServletResponse response)
    throws ServletException, IOException {
        PrintWriter izlaz = response.getWriter();
        try {
            izlaz.println("<html>");
            izlaz.println("<head>");
            izlaz.println("<title>Servlet prviServlet</title>");
            izlaz.println("</head>");
            izlaz.println("<body>");
            izlaz.println("<h1>Ovo je rezultat prvog!</h1>");

            izlaz.println("</body>");
            izlaz.println("</html>");
        } finally {
        }
        response.sendRedirect("drugiServlet");
    }
}
```

```
//source file: servleti/DrugiServlet.java

package servleti;

import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class DrugiServlet extends HttpServlet {

    protected void doGet(HttpServletRequest request,
                         HttpServletResponse response)
    throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        PrintWriter izlaz = response.getWriter();
        try {
            izlaz.println("<html>");
            izlaz.println("<head>");
            izlaz.println("<title>Servlet drugiServlet</title>");
            izlaz.println("</head>");
            izlaz.println("<body>");
            izlaz.println("<h1>Ovo je rezultat drugog servleta,
                           a ne prvog!</h1>");
            izlaz.println("<form action=\"treciServlet\" method=\"get\">");
            izlaz.println("Unesite ime: <input type=\"text\" name=\"ime\">");
            izlaz.println("<input type=\"submit\" value=\"Pozovi treci
                           servlet\">");
            izlaz.println("</form>");
            izlaz.println("</body>");
            izlaz.println("</html>");
        } finally {
            izlaz.close();
        }
    }
}

//source file: servleti/TreciServlet.java

package servleti;

import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class TreciServlet extends HttpServlet {
    protected void doGet(HttpServletRequest request,
                         HttpServletResponse response)
    throws ServletException, IOException {

        response.setContentType("text/html;charset=UTF-8");
        PrintWriter out = response.getWriter();
        try {
            String ime=request.getParameter("ime");
            out.println("<html>");
            out.println("<head>");
            out.println("<title>Servlet treciServlet</title>");
            out.println("</head>");
```

```
            out.println("<body>");
            out.println("<h1>Dobrodosli " + ime+ "</h1>");
            out.println("<br/>");
            out.println("<a href=\"index.html\">Povratak na
                        prvu stranu!</a>");
            out.println("</body>");
            out.println("</html>");
         } finally {
            out.close();
        }
    }
}
```

//XML file: WEB-INF/web.xml
//fajl koji generise okruzenje Netbeans prilikom kreiranja servleta

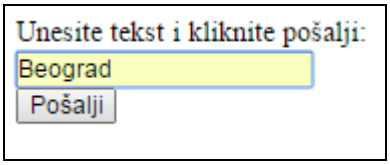
```
<?xml version="1.0" encoding="UTF-8"?>
<web-app version="2.5" xmlns="http://java.sun.com/xml/ns/javaee"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd">
    <servlet>
        <servlet-name>prviServlet</servlet-name>
        <servlet-class>servleti.PrviServlet</servlet-class>
    </servlet>
    <servlet>
        <servlet-name>drugiServlet</servlet-name>
        <servlet-class>servleti.DrugiServlet</servlet-class>
    </servlet>
    <servlet>
        <servlet-name>treciServlet</servlet-name>
        <servlet-class>servleti.TreciServlet</servlet-class>
    </servlet>
    <servlet-mapping>
        <servlet-name>prviServlet</servlet-name>
        <url-pattern>/prviServlet</url-pattern>
    </servlet-mapping>
    <servlet-mapping>
        <servlet-name>drugiServlet</servlet-name>
        <url-pattern>/drugiServlet</url-pattern>
    </servlet-mapping>
    <servlet-mapping>
        <servlet-name>treciServlet</servlet-name>
        <url-pattern>/treciServlet</url-pattern>
    </servlet-mapping>
    <session-config>
        <session-timeout>
            30
        </session-timeout>
    </session-config>
    <welcome-file-list>
        <welcome-file>index.html</welcome-file>
    </welcome-file-list>
</web-app>
```

Primer 3 - Prosleđivanje putem zahteva i sesije

Korišćenjem Java Servlet tehnologije napisati HTML stranu koja ima formu sa tekstualnim poljem i dugme za potvrdu. HTML strana treba da pozove prvi servlet, koji će da taj uneti parametar (tekst unet kroz formu) proslediti drugom servletu korišćenjem:

a) sendRedirect metode
b) objekta klase RequestDispatcher i forward metode

U prvom servletu osim ovog parametra koji unosi korisnik, treba postaviti još dva parametra, jedan koji se prosleđuje drugom servletu preko zahteva (*request*-a) i jedan koji se prosleđuje drugom servletu preko sesije (odnosno objekta klase *HttpSession*).



Rešenje:

```
//web page: index.html

<!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 pošalji:<br>
            <input type="text" name="mojparametar"/><br/>
            <input type="submit" value="Pošalji"/>
        </form>
    </body>
</html>
```

```
//source file: servleti/Servlet1.java

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 {
```

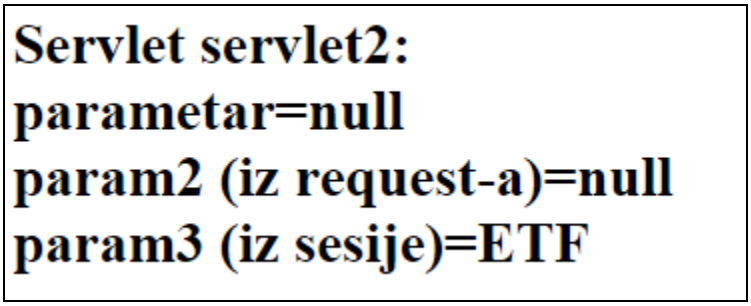
```
        String ime="ETF";
        request.setAttribute("param2", ime);

        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);
    }
}
```

//source file: servleti/Servlet2.java

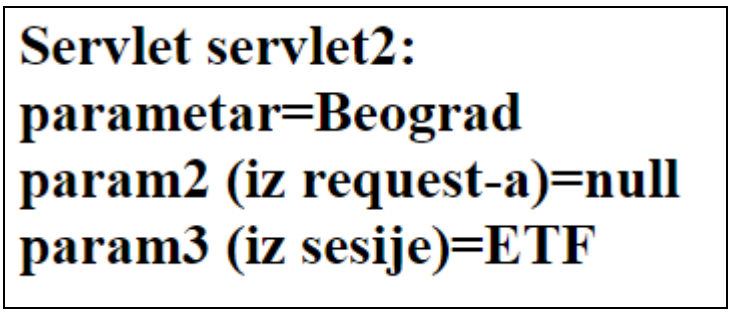
```
package servleti;

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();
        }
    }
}
```

```
//XML file: WEB-INF/web.xml
//fajl koji generise okruzenje Netbeans prilikom kreiranja servleta
<?xml version="1.0" encoding="UTF-8"?>
<web-app version="2.5" 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_2_5.xsd">
    <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>
    <session-config>
        <session-timeout>
            30
        </session-timeout>
    </session-config>
    <welcome-file-list>
        <welcome-file>index.html</welcome-file>
    </welcome-file-list>
</web-app>
```

Rezultat izvršavanja sa sendRedirect metodom:



Rezultat izvršavanja sa RequestDispatcher i forward metodom:



Primer 4 - Realizacija narudžbenice korišćenjem sesije

Korišćenjem Java Servlet tehnologije napisati HTML stranu koja ima formu sa tekstualnim poljem za dodavanje proizvoda u korpu za kupovinu i dugme za potvrdu. HTML strana treba da pozove servlet, koji će da taj uneti parametar (tekst unet kroz formu) ubaciti u niz proizvoda koji su do tada dodati u korpu. Kada se proizvod doda, treba prikazati spisak svih naručenih proizvoda koji se trenutno nalaze u korpi i mogućnost da se korisnik vrati na početnu stranicu i nastavi kupovinu, odnosno dodavanje novih proizvoda u korpu. Trenutno stanje korpe čuvati u sesiji (objektu klase *HttpSession*).

Rešenje:

```
//web page: index.html

<!DOCTYPE html>
<html>
    <head>
      <title>E-korpa</title>
    </head>
    <body bgcolor="#FDF5E6">
        <center>
            <h1>Narudžbenica</h1>
            <form action="prikaz" method="GET">
                Unesite artikal koji zelite da porucite:
                <input type="TEXT" name="noviProizvod"
                 value="Sok od breskve"><P>
                <input type="SUBMIT" value="Dodaj u korpu">
            </form>
        </center>
    </body>
</html>
```

```
//source file: servleti/PrikazKorpe.java

package servleti;

import java.io.*;
import java.util.*;
import javax.servlet.*;
import javax.servlet.http.*;

/** Servlet prikazuje listu narucenih artikala.
 *  Svi artikli u korpi cuvaju se kao ArrayList.
 *  Posto servlet moze vise puta da se pozove,
 *  sadrzaj korpe se prenosi preko sesije.
 */

@SuppressWarnings("unchecked")
public class PrikazKorpe extends HttpServlet {

  public void doGet(HttpServletRequest request,
          HttpServletResponse response)
          throws ServletException, IOException {
```

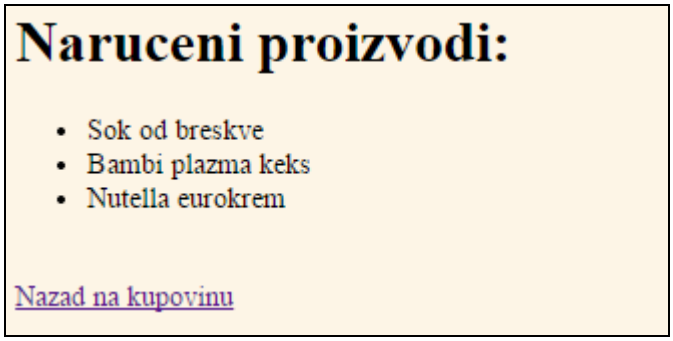
```
    HttpSession sesija = request.getSession();
    ArrayList<String> prethodniProizvodi =
            (ArrayList<String>) sesija.getAttribute("prethodniArtikli");
    if (prethodniProizvodi == null) {
      prethodniProizvodi = new ArrayList<String>();
      sesija.setAttribute("prethodniArtikli", prethodniProizvodi);
    }
    String novi = request.getParameter("noviProizvod");
    if ((novi != null)
            && (!novi.trim().equals(""))) {
      prethodniProizvodi.add(novi);
    }
    response.setContentType("text/html");
    PrintWriter out = response.getWriter();
    String naslov = "Naruceni proizvodi:";
    String docType =
            "<!DOCTYPE HTML PUBLIC \"-//W3C//DTD HTML 4.0 "
            + "Transitional//EN\">\n";
    out.println(docType
            + "<html>\n"
            + "<head><title>" + naslov + "</title></head>\n"
            + "<body bgcolor=\"#FDF5E6\">\n"
            + "<h1>" + naslov + "</h1>");
    if (prethodniProizvodi.isEmpty()) {
      out.println("<i>Nema nijednog artikla.</i>");
    } else {
      out.println("<ul>");
      for (String stavka : prethodniProizvodi) {
        out.println("  <li>" + stavka);
      }
      out.println("</ul>");
    }
    out.println("<br/>");
    out.println("<a href=\"e-korpa.html\"> Nazad na kupovinu </a>");
    out.println("</body></html>");
  }
}
```

//XML file: WEB-INF/web.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app version="2.5" 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_2_5.xsd">
    <session-config>
        <session-timeout>
            30
        </session-timeout>
    </session-config>
    <servlet>
        <servlet-name>prikaz</servlet-name>
        <servlet-class>servleti.PrikazKorpe</servlet-class>
    </servlet>
```

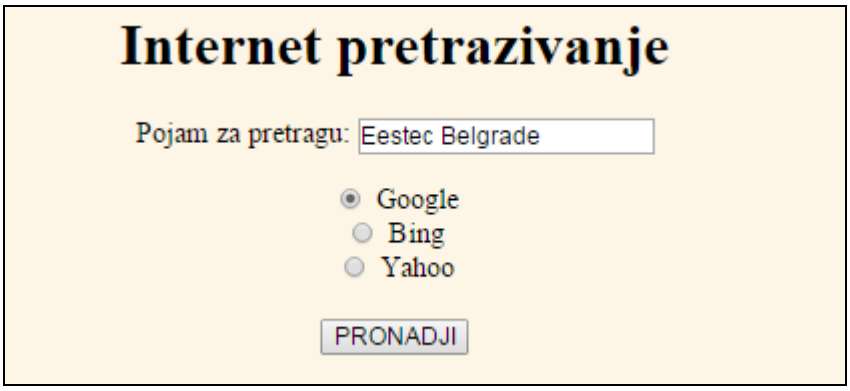
```
    <servlet-mapping>
        <servlet-name>prikaz</servlet-name>
        <url-pattern>/prikaz</url-pattern>
    </servlet-mapping>
    <welcome-file-list>
        <welcome-file>e-korpa.html</welcome-file>
    </welcome-file-list>
</web-app>
```

Prikaz korpe sa naručenim proizvodima:



Primer 5 - Pretraživanje pojmova korišćenjem različitih pretraživača

Korišćenjem Java Servlet tehnologije napraviti HTML formu koja služi za pretraživanje pojmova korišćenjem različitih internet pretraživača. Korisnik treba da ima mogućnost da u formi unese u tekstualno polje pojam koji želi da pronađe i da u listi ponuđenih internet pretraživača (eng. search engines) odabere od kog pretraživača želi da dobije pojam. Lista internet pretraživača treba da bude prikazana kao lista radio dugmadi, po jedan za svaki pretraživač (Google, Bing i Yahoo).



Rešenje:

Servlet *FormaPretrazivanja* treba da napravi HTML formu koja služi za unos parametara pretrage - pojma koji pretražujemo i izbor pretraživača. Pojam koji pretražujemo treba da se unese u tekstualno polje, koje će biti prosleđeno upitu pretrage, a upit će biti formiran u zavisnosti koji pretraživač smo odabrali. Izbor pretraživača izvršićemo odabirom jednog radio dugmeta, a nazivi pretraživača dobijaju se iz klase *VrstePretrazivaca*.

```
//source file: FormaPretrazivanja.java

import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class FormaPretrazivanja extends HttpServlet {
  public void doGet(HttpServletRequest request,
                    HttpServletResponse response)
      throws ServletException, IOException {
    response.setContentType("text/html");
    PrintWriter out = response.getWriter();
    String naslov = "Internet pretrazivanje";
    String URLadresa = "odgovor";
    String docType =
      "<!DOCTYPE HTML PUBLIC \"-//W3C//DTD HTML 4.0 " +
      "Transitional//EN\">\n";
    out.println
      (docType +
       "<HTML>\n" +
       "<HEAD><TITLE>" + naslov + "</TITLE></HEAD>\n" +
       "<BODY BGCOLOR=\"#FDF5E6\">\n" +
       "<CENTER>\n" +
       "<H1>" + naslov + "</H1>\n" +
       "<FORM ACTION=\"" + URLadresa + "\">\n" +
       "  Pojam za pretragu: \n" +
       "  <INPUT TYPE=\"TEXT\" NAME=\"pojamzapretragu\"><P>\n");

    Pretrazivac[] vrste = VrstePretrazivaca.getPretrazivaci();
    for(int i=0; i<vrste.length; i++) {
      String naziv = vrste[i].getIme();
      out.println("<INPUT TYPE=\"RADIO\" " +
                  "NAME=\"tipPretrazivaca\" " +
                  "VALUE=\"" + naziv + "\">\n");
      out.println(naziv + "<BR>\n");
    }
    out.println
      ("<BR>  <INPUT TYPE=\"SUBMIT\" VALUE=\"PRONADJI\">\n" +
       "</FORM>\n" +
       "</CENTER></BODY></HTML>");
  }
}
```

Klasa *Pretrazivac* služi da poveže ime pretraživača sa njegovom URL adresom. Metoda *napraviURL* treba da izgradi rezultujuću stranicu za svaki pretraživač posebno, tako što na osnovnu URL adresu pretraživača i sufiksa za ključ koji taj pretraživač koristi u svom upitu (http://...?nekiKljuc=") doda pojam koji se traži u URL-enkodovanom formatu (internet+programiranje).

//source file: Pretrazivac.java

```
public class Pretrazivac {
  private String ime, URLadresa;

  public Pretrazivac(String ime,
                     String URLadresa) {
    this.ime = ime;
    this.URLadresa = URLadresa;
  }

  public String napraviURL(String pojam) {
    return(URLadresa + pojam);
  }

  public String getIme() {
    return(ime);
  }
}
```

Klasa *VrstePretrazivaca* ima statički niz pretraživača, koji sadrži njihove adrese (poziva se konstruktor prethodne klase *Pretrazivac* koji pravi po jedan objekat za svaki internet pretraživač).

//source file: VrstePretrazivaca.java

```
public class VrstePretrazivaca {
  private static Pretrazivac[] pretrazivaci =
    { new Pretrazivac("Google",
                      "http://www.google.com/#q="),
      new Pretrazivac("Bing",
                      "http://www.bing.com/search?q="),
      new Pretrazivac("Yahoo",
                      "http://search.yahoo.com/bin/search?p=")
    };

  public static Pretrazivac[] getPretrazivaci() {
    return(pretrazivaci);
  }

  public static String napraviURL(String searchEngineName,
                                  String searchString) {
    Pretrazivac[] searchSpecs = getPretrazivaci();
    String URLadresa = null;
    for(Pretrazivac spec : searchSpecs) {
      if (spec.getIme().equalsIgnoreCase(searchEngineName)) {
        URLadresa = spec.napraviURL(searchString);
        break;
      }
```

```
    }
    return(URLadresa);
  }
}
```

Servlet *OdgovorPretrazivaca* dobija se kao rezultat izvršavanja forme u servletu *FormaPretrazivanja*. Cilj ovog servleta je da prihvati pojam za pretragu i vrstu pretraživača iz forme koju popunjava korisnik i da pošalje upit internet pretraživaču koji je odabran u toj formi. Ako je pretraživač izabran u listi radio dugmadi, šalje se odgovor 302 korišćenjem *sendRedirect* metode, a ako pretraživač nija izabran, šalje se odgovor 404 korišćenjem *sendError*.

URL enkoder menja blanko (space) znak sa znakom + i sve druge nealfabetske karaktere konvertuje u "%XY" gde je XY heksadecimalna vrednost ASCII (ili ISO Latin-1) karaktera. Veb pregledači uvek enkoduju vrednosti forme, ali *getParameter* metoda ih automatski dekoduje.

//source file: OdgovorPretrazivaca.java

```
import java.io.*;
import java.net.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class OdgovorPretrazivaca extends HttpServlet {
  public void doGet(HttpServletRequest request,
                    HttpServletResponse response)
      throws ServletException, IOException {

   String kljuc = request.getParameter("pojamzapretragu");
    if (kljuc == null || kljuc.isEmpty()) {
      reportProblem(response, "Niste uneli pojam koji zelite da pronadjete.");
      return;
    }

    kljuc = URLEncoder.encode(kljuc, "UTF-8");
    String imepretrazivaca = request.getParameter("tipPretrazivaca");
    if (imepretrazivaca == null || imepretrazivaca.isEmpty()) {
      reportProblem(response, "Niste uneli pretrazivac koji koristite.");
      return;
    }
    String pretragaURL = VrstePretrazivaca.napraviURL(imepretrazivaca, kljuc);
    if (pretragaURL != null) {
      response.sendRedirect(pretragaURL);
    } else {
      reportProblem(response, "Nije prepoznat pretrazivac.");
    }
  }

  private void reportProblem(HttpServletResponse response, String message)
      throws IOException {
    response.sendError(response.SC_NOT_FOUND, message);
  }
}
```

//XML file: WEB-INF/web.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app        version="2.5"        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_2_5.xsd">
    <servlet>
        <servlet-name>odgovor</servlet-name>
        <servlet-class>OdgovorPretrazivaca</servlet-class>
    </servlet>
    <servlet-mapping>
        <servlet-name>odgovor</servlet-name>
        <url-pattern>/odgovor</url-pattern>
    </servlet-mapping>
    <servlet>
        <servlet-name>index</servlet-name>
        <servlet-class>FormaPretrazivanja</servlet-class>
    </servlet>
    <servlet-mapping>
        <servlet-name>index</servlet-name>
        <url-pattern>/index</url-pattern>
    </servlet-mapping>
    <session-config>
        <session-timeout>
            30
        </session-timeout>
    </session-config>
    <welcome-file-list>
        <welcome-file>index</welcome-file>
    </welcome-file-list>
</web-app>
```

Primer 6 - Brojač i rad sa fajlom

Korišćenjem Java Servlet tehnologije napraviti servlet koji čita početno stanje brojača iz tekstualnog fajla brojacposeta.txt i inkrementira taj brojač. Servletu treba omogućiti da se za svaki ponovni poziv tog servleta uvećava taj brojač u tekstualnom fajlu. Koristiti metode *init* za čitanje početne vrednosti brojača i *destroy* za ažuriranje vrednosti brojača u fajlu. Ukoliko tekstualni fajl inicijalno ne postoji na lokaciji sa koje čitamo, treba ga kreirati, a brojač postaviti na početnu vrednost 0.

Rešenje:

```
//web page: index.html

<!DOCTYPE HTML>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Brojac</title>
    </head>
    <body>
        <h2>Kliknite na servlet!</h2>
        <br/>
        <a href="brojacservlet">click</a>
    </body>
</html>
```

```
//source file: paket/MyServlet.java

package paket;

import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class MyServlet extends HttpServlet {

  int count;

  @Override
  public void init(ServletConfig config) throws ServletException {
    // Prvo pozivamo init metod bazne klase
    super.init(config);

    // Trazimo fajl u kome stoji ranija vrednost brojaca
    try {
      File file = new File("c:/brojacposeta.txt");
      FileReader fileReader = new FileReader(file);
      BufferedReader bufferedReader = new BufferedReader(fileReader);
      String pocetno = bufferedReader.readLine();
      count = Integer.parseInt(pocetno);
      return;
    }
```

```
    catch (FileNotFoundException e1) { }  // Ako stanje nije sacuvano
    catch (IOException e2) { }            // Ako ima problem prilikom citanja
    catch (NumberFormatException e3) { }  // Ako sacuvano stanje nije broj

    // Nismo nasli nigde zapisano inicijalno stanje brojaca, pa ga resetujemo

    count = 0;
  }

  @Override
  public void doGet(HttpServletRequest req, HttpServletResponse res)
                              throws ServletException, IOException {

        res.setContentType("text/html;charset=UTF-8");
        PrintWriter out = res.getWriter();
        count++;
        try {
            out.println("<html>");
            out.println("<head>");
            out.println("<title>Brojac poziva</title>");
            out.println("</head>");
            out.println("<body>");
            out.println("Od kad je sveta i veka, ovaj servlet je pozvan " +
                count + " puta.");
            out.println("<a href=\"brojacservlet\">klikni ponovo</a>");

            out.println("</body>");
            out.println("</html>");
        }
        finally{
            out.close();
        }
  }

  @Override
  public void destroy() {
    saveState();
  }

  public void saveState() {
    // Pokusavamo da zapisemo stanje brojaca na perzistentnu memoriju (fajl)
    try {
      File file = new File("c:/brojacposeta.txt");
      FileWriter fileWriter = new FileWriter(file);
      String stanje = Integer.toString(count);
      fileWriter.write(stanje, 0, stanje.length());
      fileWriter.close();
      return;
    }
    catch (IOException e) {  // ako postoji problem prilikom upisa
    }
  }
}
```

```
//XML file: WEB-INF/web.xml

<?xml version="1.0" encoding="UTF-8"?>
<web-app        version="2.5"        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_2_5.xsd">
    <servlet>
        <servlet-name>moj_servlet</servlet-name>
        <servlet-class>paket.MyServlet</servlet-class>
    </servlet>
    <servlet-mapping>
        <servlet-name>moj_servlet</servlet-name>
        <url-pattern>/brojacservlet</url-pattern>
    </servlet-mapping>
    <session-config>
        <session-timeout>
            30
        </session-timeout>
    </session-config>
    <welcome-file-list>
        <welcome-file>index.html</welcome-file>
    </welcome-file-list>
</web-app>
```

Primer 7 - Loto

Korišćenjem Java Servlet tehnologije napraviti servlet koji korisniku prikazuje 7 slučajno odabranih brojeva na igri loto, koristeći metodu *Math.random()*. Brojeve odabrati prilikom inicijalizacije servleta (metoda *init()*). Brojevi neka budu u opsegu od 1 do 37.

Rešenje:

```
//source file: Loto.java

import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

/** Primer koriscenja init i
 *  getLastModified metoda kod servleta.
 */

public class Loto extends HttpServlet {
  private long modTime;
  private int[] brojevi = new int[7];

  /** Ova init metoda se poziva samo kada se servlet
   *  prvi put poziva, pre obrade prvog request-a.
   */
```

```
  public void init() throws ServletException {
    // zaokruzuje na pribliznu vrednost
    modTime = System.currentTimeMillis()/1000*1000;
    for(int i=0; i<brojevi.length; i++) {
      brojevi[i] = randomNum();
    }
  }

  /** Metoda doGet vraca listu koji je izracunala metoda init. */

  public void doGet(HttpServletRequest request,
                    HttpServletResponse response)
      throws ServletException, IOException {
    response.setContentType("text/html");
    PrintWriter out = response.getWriter();
    String naslov = "Tvoja loto kombinacija";
    String docType =
      "<!DOCTYPE HTML>\n";
    out.println(docType +
                "<HTML>\n" +
                "<HEAD><TITLE>" + naslov + "</TITLE></HEAD>\n" +
                "<BODY BGCOLOR=\"yellow\">\n" +
                "<H1 ALIGN=CENTER>" + naslov + "</H1>\n" +
                "<B>Izracunali smo slucajnu kombinaciju " +
                brojevi.length +" loto brojeva najboljih za Vas: " +
                ".</B> <OL>");

    for(int i=0; i<brojevi.length; i++) {
      out.println("  <LI>" + brojevi[i]);
    }
    out.println("</OL>" +
                "</BODY></HTML>");
  }

  /** Standarda servisna metoda koja uporedjuje ovaj datum
   *  sa datumom u If-Modified-Since u headeru requesta.
   *  Ako je getLastModified datum kasniji ili ako ne postoji
   *  polje If-Modified-Since u headeru, doGet metoda se poziva
   *  uobicajno. Ako getLastModified datum predstavlja isti
   *  ili raniji, servisna metoda ce kao odgovor
   *  poslati 304 (Not Modified) i nece pozvati doGet metodu.
   *  Pretrazivac ce u tom slucaju iskoristiti kesiranu
   *  verziju te stranice (tj tog servleta).
   */

  public long getLastModified(HttpServletRequest request) {
    return(modTime);
  }

  // Slucaj ceo broj izmedju 1 i 37
  // random metoda vraca broj izmedju 0 i 1

  private int randomNum() {
    return((int)(1 + Math.random() * 37));
  }
}
```

Primer 8 - Kolačići

Korišćenjem Java Servlet tehnologije napraviti servlet koji proverava da li u internet pregledaču postoji kolačić (*cookie*) koji pokazuje da li je korisnik već posećivao datu veb stranicu, pa ako ne postoji da ispiše „Dobrodošli!“, a ako kolačić postoji, odnosno korisnik je već posećivao datu veb stranicu, da se ispiše „Lepo je videti Vas opet“.

Rešenje:

```
//source file: cookie/MyCookie.java

package cookie;

import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class MyCookie extends HttpServlet {

  public void doGet(HttpServletRequest request, HttpServletResponse response)
          throws ServletException, IOException {

    boolean newuser = true;
    Cookie[] cookies = request.getCookies();
    if (cookies != null) {
      for (int i = 0; i < cookies.length; i++) {
        Cookie c = cookies[i];
        if ((c.getName().equals("stariPosetilac"))
            && (c.getValue().equals("yes"))) {
          newuser = false;
          break;
        }
      }
    }
    String title;
    if (newuser) {
      Cookie posetilacCookie =
              new Cookie("stariPosetilac", "yes");
//            posetilacCookie.setMaxAge(60 * 60 * 24 * 365);
      posetilacCookie.setMaxAge(60);
      response.addCookie(posetilacCookie);
      title = "Dobrodosli!";
    } else {
      title = "Lepo je videti Vas opet";
    }

    response.setContentType("text/html;charset=UTF-8");
    PrintWriter izlaz = response.getWriter();

    try {
      izlaz.println("<html>");
      izlaz.println("<head>");
      izlaz.println("<title>" + title + "</title>");
      izlaz.println("</head>");
```

```
      izlaz.println("<body>");
      izlaz.println("<h1>" + title + "</h1>");
      izlaz.println("</body>");
      izlaz.println("</html>");
    } finally {
      izlaz.close();
    }
  }
}
```

Primer 9 - Korisnici

Korišćenjem Java Servlet tehnologije napraviti servlet koji prikazuje HTML stranu sa formom za logovanje korisnika, u koju korisnik treba da unese svoje korisničko ime i svoju lozinku (kredencijali), i ukoliko su korisnički kredencijali ispravni (postoje u bazi podataka u tabeli *KorisnikInfo*) korisniku prikazati stranicu sa svojim ličnim podacima (servlet *Prikaz*), a ukoliko podaci nisu dobri korisniku prikazati stranicu sa greškom (servlet *Greska*). Takođe, potrebno je realizovati još tri servleta: servlete za promenu ličnih podataka (servlet *Promena*, koji prikazuje formu za izmenu korisničkih podataka i koji poziva za ažuriranje tabele u bazi drugi servlet *Izmena*) i servlet koji služi prekidanje sesije i izlogovanje iz sistema (servlet *Logout*).

Izgled tabela u bazi *dbkorisnik*:

KorisnikInfo		
#	Naziv kolone	Tip
1	username (PK)	varchar(32)
2	password	varchar(32)
3	first_name	varchar(32)
4	last_name	varchar(32)
5	email	varchar(32)
6	phone	varchar(32)

Rešenje:

```
//source file: util/DB.java

package util;

import java.sql.Connection;
import java.sql.DriverManager;

public class DB {

    private static DB instance;
    private static final int MAX_CON = 5;
    private static final Connection[] bafer = new Connection[MAX_CON];
    private int first = 0, last = 0, free = MAX_CON;

    private DB() { //za MySQL
        try {
            Class.forName("com.mysql.jdbc.Driver");
            for (int i = 0; i < MAX_CON; i++) {
                bafer[i] =
                        DriverManager.getConnection("jdbc:mysql://
                        localhost:3306/dbkorisnik", "root", "");
            }
        } catch (Exception e) {
            e.printStackTrace();
        }
    }

    /* za Access
     private DB(){
     try{
           Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
           for(int i=0; i<MAX_CON; i++)
           bafer[i] = DriverManager.getConnection("jdbc:odbc:dbkorisnik");
     }catch(Exception e){}
     }
     */
    public static DB getInstance() {
        if (instance == null) {
            instance = new DB();
        }
        return instance;
    }

    public synchronized Connection getConnection() {
        if (free == 0) {
            return null;
        }
        free--;
        Connection con = bafer[first];
        first = (first + 1) % MAX_CON;
        return con;
    }
```

```
    public synchronized void putConnection(Connection con) {
        if (con == null) {
            return;
        }
        free++;
        bafer[last] = con;
        last = (last + 1) % MAX_CON;
    }
}
```

```
//source file: beans/Korisnik.java
//Java bean

package beans;

public class Korisnik {

    private String username = "";
    private String password = "";
    private String ime = "";
    private String prezime = "";
    private String email = "";
    private String telefon = "";

    public String getUsername() {
        return username;
    }

    public void setUsername(String username) {
        this.username = username;
    }

    public String getPassword() {
        return password;
    }

    public void setPassword(String password) {
        this.password = password;
    }

    public String getIme() {
        return ime;
    }

    public void setIme(String ime) {
        this.ime = ime;
    }

    public String getPrezime() {
        return prezime;
    }

    public void setPrezime(String prezime) {
        this.prezime = prezime;
    }
```

```
    public String getEmail() {
        return email;
    }

    public void setEmail(String email) {
        this.email = email;
    }

    public String getTelefon() {
        return telefon;
    }

    public void setTelefon(String telefon) {
        this.telefon = telefon;
    }
}
```

//source file: servleti/Index.java

```
package servleti;

import beans.Korisnik;
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class Index extends HttpServlet {

    protected void processRequest(HttpServletRequest request,
                                  HttpServletResponse response)
            throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        PrintWriter out = response.getWriter();
        HttpSession sesija = request.getSession();
        Korisnik korisnik = (Korisnik) sesija.getAttribute("korisnik");
        String username = "", password = "";
        if (korisnik != null) {
            username = korisnik.getUsername();
            password = korisnik.getPassword();
        }
        String poruka = (String) request.getAttribute("poruka");
        if (poruka == null) {
            poruka = "Dobrodosli!";
        }
        try {
            out.println("<html>");
            out.println("<head>");
            out.println("<title>Korisnicka aplikacija</title>");
            out.println("</head>");
            out.println("<body>");
            out.println("<h3>" + poruka + " </h3><br/><br/>");
            out.println("Molimo, ulogujte se: ");
            out.println("<table><tr><td>Korisnicko ime:</td>");
            out.println("<form action=\"login\" method=\"POST\">");
            out.println("<td><input type=\"text\" name=\"username\" value=\""
                        + username + "\" width=\"20\"></td></tr>");
            out.println("<tr><td>Lozinka:</td>");
```

```
            out.println("<td><input type=\"password\" name=\"password\"
                        value=\"" + password + "\"  width=\"20\"></td></tr>");
            out.println("<tr><td><input type=\"submit\"
                         value=\"Ulogujte se\"</td>");
            out.println("<td><input type=\"reset\"
                        value=\"Ponistite\"</td><tr></table>");
            out.println("</form>");
            out.println("</body>");
            out.println("</html>");
        } finally {
            out.close();
        }
    }

    @Override
    protected void doGet(HttpServletRequest request,
                         HttpServletResponse response)
            throws ServletException, IOException {
        processRequest(request, response);
    }

    @Override
    protected void doPost(HttpServletRequest request,
                          HttpServletResponse response)
            throws ServletException, IOException {
        processRequest(request, response);
    }
}
```

//source file: servleti/Login.java

```
package servleti;

import beans.*;
import java.io.*;
import java.sql.*;
import javax.servlet.*;
import javax.servlet.http.*;
import util.DB;

public class Login extends HttpServlet {

  protected void processRequest(HttpServletRequest request,
                                HttpServletResponse response)
          throws ServletException, IOException {
    HttpSession sesija = request.getSession();
    String poruka = null;
    String username = request.getParameter("username");
    String password = request.getParameter("password");

    Korisnik korisnik = new Korisnik();
    korisnik.setUsername(username);
    korisnik.setPassword(password);

    sesija.setAttribute("korisnik", korisnik);
    if (username.isEmpty() || password.isEmpty()) {
      poruka = "Niste popunili sva polja!";
```

```
      request.setAttribute("poruka", poruka);
      RequestDispatcher rd = request.getRequestDispatcher("index");
      rd.forward(request, response);
    }

    Connection con = null;
    Statement stmt = null;
    ResultSet rs = null;
    String address = "prikaz";
    try {
      con = DB.getInstance().getConnection();
      stmt = con.createStatement();
      String upit = "select * from KorisnikInfo where username='" + username +
                    "' and password='" + password + "';";
      rs = stmt.executeQuery(upit);
      if (rs.next()) {
        String email = rs.getString("email");
        String ime = rs.getString("first_name");
        String prezime = rs.getString("last_name");
        String telefon = rs.getString("phone");
        korisnik.setEmail(email);
        korisnik.setIme(ime);
        korisnik.setPrezime(prezime);
        korisnik.setTelefon(telefon);
        stmt.close();
      } else {
        poruka = "Neispravno korisnicko ime i/ili lozinka! Pokusajte ponovo.";
        request.setAttribute("poruka", poruka);
        korisnik.setPassword("");
        address = "index";
      }
    } catch (SQLException ex) {
      sesija.invalidate();
      String errormsg = ex.getMessage();
      request.setAttribute("errormsg", errormsg);
      address = "error";
    } finally {
      DB.getInstance().putConnection(con);
    }

    request.getRequestDispatcher(address).forward(request, response);
  }

  @Override
  protected void doGet(HttpServletRequest request,
                       HttpServletResponse response)
          throws ServletException, IOException {
    processRequest(request, response);
  }

  @Override
  protected void doPost(HttpServletRequest request,
                        HttpServletResponse response)
          throws ServletException, IOException {
    processRequest(request, response);
  }
}
```

```
//source file: servleti/Prikaz.java

package servleti;

import beans.*;
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class Prikaz extends HttpServlet {
  protected void processRequest(HttpServletRequest request,
                                HttpServletResponse response)
          throws ServletException, IOException {
    response.setContentType("text/html;charset=UTF-8");
    PrintWriter out = response.getWriter();
    HttpSession sesija = request.getSession();
    Korisnik korisnik = (Korisnik) sesija.getAttribute("korisnik");
    String poruka = (String) request.getAttribute("poruka");

    if (poruka == null) {
      poruka = "Vasi podaci (prethodna stranica je \"promena\")";
    }

    try {
      out.println("<html>");
      out.println("<head>");
      out.println("<title>Korisnicka aplikacija</title>");
      out.println("</head>");
      out.println("<body>");
      out.println("<h3>" + poruka + "</h3><br/><br/>");
      out.println("<table width=\"40%\" border=\"3\">"
              + "<tr><td width=\"50%\">Korisnicko ime:</td>"
              + "<td>" + korisnik.getUsername() + "</td></tr>");
      out.println("<tr><td>Ime:</td>"
              + "<td>" + korisnik.getIme() + "</td></tr>");
      out.println("<tr><td>Prezime:</td>"
              + "<td>" + korisnik.getPrezime() + "</td></tr>");
      out.println("<tr><td>Email:</td>"
              + "<td>" + korisnik.getEmail() + "</td></tr>");
      out.println("<tr><td>Telefon:</td>"
              + "<td>" + korisnik.getTelefon() + "</td></tr>");
      out.println("</table><br/>");
      out.println("<a href=\"promena\">Promenite podatke</a>");
      out.println("<a href=\"logout\">Izlogujte se</a>");
      out.println("</body>");
      out.println("</html>");
    } finally {
      out.close();
    }
  }

  @Override
  protected void doGet(HttpServletRequest request,
                       HttpServletResponse response)
          throws ServletException, IOException {
    processRequest(request, response);
  }
```

```
  @Override
  protected void doPost(HttpServletRequest request,
                        HttpServletResponse response)
          throws ServletException, IOException {
    processRequest(request, response);
  }
}
```

//source file: servleti/Greska.java

```
package servleti;

import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class Greska extends HttpServlet {

  protected void processRequest(HttpServletRequest request,
                                HttpServletResponse response)
          throws ServletException, IOException {
    response.setContentType("text/html;charset=UTF-8");
    String errormsg = (String) request.getAttribute("errormsg");
    PrintWriter out = response.getWriter();
    try {
      out.println("<html>");
      out.println("<head>");
      out.println("<title>Greska!</title>");
      out.println("</head>");
      out.println("<body>");
      out.println("<h1>Greska u radu sa bazom podataka</h1>");
      out.println("<br/>" + errormsg + "<br/>");
      out.println("</body>");
      out.println("</html>");
    } finally {
      out.close();
    }
  }

  @Override
  protected void doGet(HttpServletRequest request,
                       HttpServletResponse response)
          throws ServletException, IOException {
    processRequest(request, response);
  }

  @Override
  protected void doPost(HttpServletRequest request,
                        HttpServletResponse response)
          throws ServletException, IOException {
    processRequest(request, response);
  }
}
```

```
//source file: servleti/Promena.java

package servleti;

import beans.Korisnik;
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class Promena extends HttpServlet {

  protected void processRequest(HttpServletRequest request,
                                HttpServletResponse response)
          throws ServletException, IOException {
    response.setContentType("text/html;charset=UTF-8");
    PrintWriter out = response.getWriter();
    HttpSession sesija = request.getSession();
    Korisnik korisnik = (Korisnik) sesija.getAttribute("korisnik");

    try {
      out.println("<html>");
      out.println("<head>");
      out.println("<title>Korisnicka aplikacija</title>");
      out.println("</head>");
      out.println("<body>");
      out.println("<h3>Vasi podaci koje mozete menjati</h3><br/><br/>");
      out.println("<form action=\"izmena\" method=\"post\" >");
      out.println("<table width=\"40%\" border=\"3\">"
              + "<tr><td width=\"50%\">Ime:</td>"
              + "<td><input type=\"text\" name=\"ime\" value=\""
              + korisnik.getIme() + "\"/></td></tr>");
      out.println("<tr><td>Prezime:</td>"
              + "<td><input type=\"text\" name=\"prezime\" value=\""
              + korisnik.getPrezime() + "\"/></td></tr>");

      out.println("<tr><td>Email:</td>"
              + "<td><input type=\"text\" name=\"email\" value=\""
              + korisnik.getEmail() + "\"/></td></tr>");
      out.println("<tr><td>Telefon:</td>"
              + "<td><input type=\"text\" name=\"telefon\" value=\""
              + korisnik.getTelefon() + "\"/></td></tr>");
      out.println("</table><br/>");
      out.println("<input type=\"submit\" value=\"Prihvatite izmenu\"/>");
      out.println("<input type=\"reset\" value=\"Odbacite izmenu\"/>");

      out.println("</form></body>");
      out.println("<a href=\"prikaz\">Nazad</a>");
      out.println("</html>");
    } finally {
      out.close();
    }
  }
```

```
  @Override
  protected void doGet(HttpServletRequest request,
                       HttpServletResponse response)
          throws ServletException, IOException {
    processRequest(request, response);
  }

  @Override
  protected void doPost(HttpServletRequest request,
                        HttpServletResponse response)
          throws ServletException, IOException {
    processRequest(request, response);
  }
}
```

//source file: servleti/Izmena.java

```
package servleti;

import beans.Korisnik;
import java.io.*;
import java.sql.*;
import javax.servlet.*;
import javax.servlet.http.*;
import util.DB;

public class Izmena extends HttpServlet {

  protected void processRequest(HttpServletRequest request,
                                HttpServletResponse response)
          throws ServletException, IOException {
    HttpSession sesija = request.getSession();
    Korisnik korisnik = (Korisnik) sesija.getAttribute("korisnik");
    String ime = request.getParameter("ime");
    String prezime = request.getParameter("prezime");
    String telefon = request.getParameter("telefon");
    String email = request.getParameter("email");

    korisnik.setIme(ime);
    korisnik.setPrezime(prezime);
    korisnik.setTelefon(telefon);
    korisnik.setEmail(email);
    String upit = "update KorisnikInfo "
            + "set first_name='" + ime + "',last_name='" + prezime
            + "', phone='" + telefon + "', email='" + email + "'"
            + "where username='" + korisnik.getUsername() + "';";

    Connection con = null;
    Statement stmt = null;
    String address = "prikaz";
    try {
      con = DB.getInstance().getConnection();
      stmt = con.createStatement();
      stmt.executeUpdate(upit);
      stmt.close();
    } catch (SQLException ex) {
      sesija.invalidate();
```

```
      String errormsg = ex.getMessage();
      request.setAttribute("errormsg", errormsg);
      address = "error";
    } finally {
      DB.getInstance().putConnection(con);
    }
    request.setAttribute("poruka", "Podaci su uspesno izmenjeni");
    request.getRequestDispatcher(address).forward(request, response);
  }

  @Override
  protected void doGet(HttpServletRequest request,
                       HttpServletResponse response)
          throws ServletException, IOException {
    processRequest(request, response);
  }

  @Override
  protected void doPost(HttpServletRequest request,
                        HttpServletResponse response)
          throws ServletException, IOException {
    processRequest(request, response);
  }
}
```

//source file: servleti/Logout.java

```
package servleti;

import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class Logout extends HttpServlet {

  protected void processRequest(HttpServletRequest request,
                                HttpServletResponse response)
          throws ServletException, IOException {
    HttpSession sesija = request.getSession();
    sesija.invalidate();
    response.setContentType("text/html;charset=UTF-8");
    PrintWriter out = response.getWriter();
    try {
      out.println("<html>");
      out.println("<head>");
      out.println("<title>Dovidjenja</title>");
      out.println("</head>");
      out.println("<body>");
      out.println("<h1>Dovidjenja</h1>");
      out.println("<a href=\"index\">nazad na login stranu</a>");
      out.println("</body>");
      out.println("</html>");
    } finally {
      out.close();
    }
  }
```

```
  @Override
  protected void doGet(HttpServletRequest request,
                       HttpServletResponse response)
          throws ServletException, IOException {
    processRequest(request, response);
  }

  @Override
  protected void doPost(HttpServletRequest request,
                        HttpServletResponse response)
          throws ServletException, IOException {
    processRequest(request, response);
  }
}
```

//XML file: WEB-INF/web.xml

```
<?xml version="1.0" encoding="UTF-8"?>

<web-app         version="2.5"         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_2_5.xsd">

    <servlet>
        <servlet-name>index</servlet-name>
        <servlet-class>servleti.Index</servlet-class>
    </servlet>
    <servlet>
        <servlet-name>login</servlet-name>
        <servlet-class>servleti.Login</servlet-class>
    </servlet>
    <servlet>
        <servlet-name>greska</servlet-name>
        <servlet-class>servleti.Greska</servlet-class>
    </servlet>
    <servlet>
        <servlet-name>promena</servlet-name>
        <servlet-class>servleti.Promena</servlet-class>
    </servlet>
    <servlet>
        <servlet-name>izmena</servlet-name>
        <servlet-class>servleti.Izmena</servlet-class>
    </servlet>
    <servlet>
        <servlet-name>prikaz</servlet-name>
        <servlet-class>servleti.Prikaz</servlet-class>
    </servlet>
    <servlet>
        <servlet-name>logout</servlet-name>
        <servlet-class>servleti.Logout</servlet-class>
    </servlet>
    <servlet-mapping>
        <servlet-name>index</servlet-name>
        <url-pattern>/index</url-pattern>
    </servlet-mapping>
```

```
    <servlet-mapping>
        <servlet-name>login</servlet-name>
        <url-pattern>/login</url-pattern>
    </servlet-mapping>
    <servlet-mapping>
        <servlet-name>greska</servlet-name>
        <url-pattern>/error</url-pattern>
    </servlet-mapping>
    <servlet-mapping>
        <servlet-name>promena</servlet-name>
        <url-pattern>/promena</url-pattern>
    </servlet-mapping>
    <servlet-mapping>
        <servlet-name>izmena</servlet-name>
        <url-pattern>/izmena</url-pattern>
    </servlet-mapping>
    <servlet-mapping>
        <servlet-name>prikaz</servlet-name>
        <url-pattern>/prikaz</url-pattern>
    </servlet-mapping>
    <servlet-mapping>
        <servlet-name>logout</servlet-name>
        <url-pattern>/logout</url-pattern>
    </servlet-mapping>
    <session-config>
        <session-timeout>
            30
        </session-timeout>
    </session-config>
    <welcome-file-list>
        <welcome-file>index</welcome-file>
    </welcome-file-list>
</web-app>
```

Primer 10 - Poređenje brojeva

Korišćenjem Java Server Pages tehnologije napisati JSP stranu koja preko HTML forme prihvata dva broja, prosleđuje ih drugoj JSP strani i upoređuje ih. Ukoliko je prvi broj manji od drugog, treba ispisati „Razlika je manja od nule“ i obojiti pozadinu u plavu boju, ukoliko je prvi broj veći od drugog, treba ispisati „Razlika je veća od nule“ i obojiti pozadinu u zelenu boju, a ukoliko su brojevi isti, treba ispisati „Razlika je nula“ i obojiti pozadinu u žutu boju. Sve provere treba realizovati na JSP stranama.

Rešenje:

//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>
```

//web page: rezultat.jsp

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

<!-- direktiva -->
<%@ page errorPage="error.jsp" %>

<!-- deklaracija -->
<%! int a, b, razlika;
    String prvi, drugi;
%>

<!-- skriplet -->
<%
    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 {%>
          red
          <%} %>">

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

//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>
        <h2>Greska!</h2>
        <h3><%=exception%></h3>
    </body>
</html>
```

Primer 11 - Poređenje brojeva (MVC pristup)

Korišćenjem Java Server Pages tehnologije napisati JSP stranu koja preko HTML forme prihvata dva broja, prosleđuje ih Java Servletu, koji ih upoređuje. Ukoliko je prvi broj veći od drugog, servlet nas preusmerava na stranu *vece.jsp*, ukoliko je prvi broj manji od drugog, servlet nas preusmerava na stranu *manje.jsp*, a ukoliko je razlika jednaka 0, servlet nas preusmerava na stranu *nula.jsp*. Sve provere treba realizovati na servletu.

Rešenje:

//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 resavanja problema - prednosti i mane</H2>
       <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>
```

//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">
        <h1>Razlika je veca od nule.</h1>
    </body>
</html>
```

//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">
        <h1>Razlika je manja od nule.</h1>
    </body>
</html>
```

//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="#ff0000">
        <h1>Razlika je nula.</h1>
    </body>
</html>
```

//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">
        <h2>Greska!</h2>
    </body>
</html>
```

//source file: servlets/UporediServlet.java

```
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";
        }
```

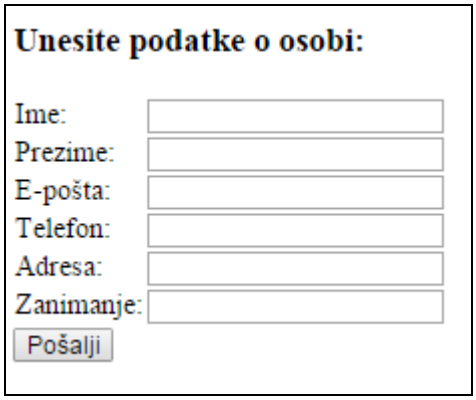
```
        RequestDispatcher dispatcher = request.getRequestDispatcher(address);
        dispatcher.forward(request, response);
    }
}
```

//XML file: WEB-INF/web.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app         version="2.5"         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_2_5.xsd">

    <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>
```

Primer 12 - Podaci o osobi (JSP sa binovima)

Korišćenjem Java Server Pages tehnologije napisati JSP stranu koja preko HTML forme prihvata podatke o osobi: ime, prezime, adresu e-pošte, telefon, adresu i zanimanje i prosleđuje ih drugoj JSP strani, koja prikazuje te podatke. Za prikaz podataka potrebno je koristiti Java binove (*Java beans*).



Rešenje:

//web page: index.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE HTML>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Rad sa binovima</title>
    </head>
    <body>
        <h3>Unesite podatke o osobi:</h3>
        <form action="prikaz.jsp" method="post">
            <table>
                <tr>
                    <td>Ime:</td>
                    <td><input type="text" name="ime"/></td>
                </tr>
                <tr>
                    <td>Prezime:</td>
                    <td><input type="text" name="prezime"/></td>
                </tr>
                <tr>
                    <td>E-pošta:</td>
                    <td><input type="text" name="email"/></td>
                </tr>
                <tr>
                    <td>Telefon:</td>
                    <td><input type="text" name="telefon"/></td>
                </tr>
                <tr>
                    <td>Adresa:</td>
                    <td><input type="text" name="adresa"/></td>
                </tr>
                <tr>
                    <td>Zanimanje:</td>
                    <td><input type="text" name="zanimanje"/></td>
                </tr>
            </table>
            <input type="submit" value="Pošalji">
        </form>
    </body>
</html>
```

//source file: beans/Korisnik.java

```
package beans;
public class Korisnik {
    private String ime = "";
    private String prezime = "";
    private String email = "";
    private String telefon = "";
    private String adresa = "";
    private String zanimanje = "";
```

```
    public String getIme() {
        return ime;
    }

    public void setIme(String ime) {
        this.ime = ime;
    }

    public String getPrezime() {
        return prezime;
    }

    public void setPrezime(String prezime) {
        this.prezime = prezime;
    }

    public String getEmail() {
        return email;
    }

    public void setEmail(String email) {
        this.email = email;
    }

    public String getTelefon() {
        return telefon;
    }

    public void setTelefon(String telefon) {
        this.telefon = telefon;
    }

    public String getAdresa() {
        return adresa;
    }

    public void setAdresa(String adresa) {
        this.adresa = adresa;
    }

    public String getZanimanje() {
        return zanimanje;
    }

    public void setZanimanje(String zanimanje) {
        this.zanimanje = zanimanje;
    }
}
```

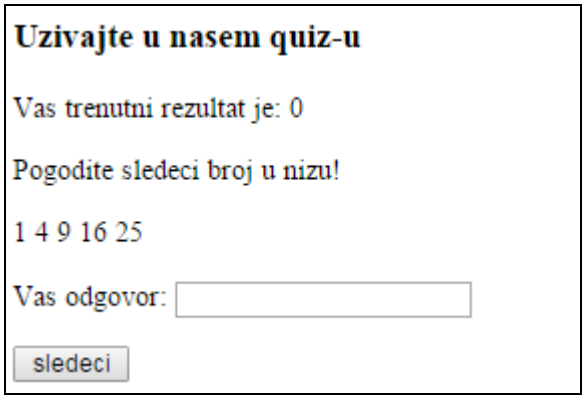
//web page: prikaz.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Prikaz</title>
```

```
    </head>
    <jsp:useBean id="korisnik" class="beans.Korisnik"  scope="session"/>
    <jsp:setProperty name="korisnik" property="*"/>
    <body>
        <table>
            <tr>
                <td>Ime:</td>
                <td>${korisnik.ime}</td>
            </tr>
            <tr>
                <td>Prezime:</td>
                <td>${korisnik.prezime}</td>
            </tr>
            <tr>
                <td>e-mail:</td>
                <td>${korisnik.email}</td>
            </tr>
            <tr>
                <td>Telefon:</td>
                <td>${korisnik.telefon}</td>
            </tr>
            <tr>
                <td>Adresa:</td>
                <td>${korisnik.adresa}</td>
            </tr>
            <tr>
                <td>Zanimanje:</td>
                <td>${korisnik.zanimanje}</td>
            </tr>
        </table>
    </body>
</html>
```

Primer 13 - Kviz

Korišćenjem tehnologija Java Servlet i Java Server Pages napraviti mini aplikaciju za kviz matematičkog znanja. Korisnik treba da ima mogućnost da na veb strani vidi pitanje, polje za unos odgovora i dugme za potvrdu, kao i trenutni rezultat koji je postigao u kviz. Prikaz pitanja i unos odgovora treba da se izvrši na JSP strani, a sve provere treba da se izvrše na servletu.



Rešenje:

//web page: index.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE HTML>
<html>
    <head>
        <title>JSP Page</title>
    </head>

    <body>
        <H2>Dobrodosli na nas kviz!</H2><br/>
        Kliknite za pocetak <a href="QuizHandler"> kviza </a>
    </body>
</html>
```

//source file: servlets/QuizHandlerServlet.java

```
package servlets;

import beans.*;
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class QuizHandlerServlet extends HttpServlet {

    protected void processRequest(HttpServletRequest request,
                                  HttpServletResponse response)
            throws ServletException, IOException {
        HttpSession session = request.getSession(true);
        QuizBean quizBean = (QuizBean) session.getAttribute("quizBean");

        String address = "quiz.jsp";

        if ((quizBean == null) || (quizBean.isFinished() == true)) {
            // pocetak kviza
            quizBean = new QuizBean();
            session.setAttribute("quizBean", quizBean);

        } else {
            // negde u toku kviza
            int userAnswer = 0;
            boolean ok = true;
            try {
                userAnswer = Integer.parseInt(request.getParameter("answer"));
            } catch (NumberFormatException nfe) {
                ok = false;
            }

            if (ok && (quizBean.checkAnswer(userAnswer) == true)) {
                quizBean.incrementScore();
            }
```

```
            quizBean.setCurrentIndex(quizBean.getCurrentIndex() + 1);

            if (quizBean.isFinished() == true) {
                address = "score.jsp";
            }
        }

        response.sendRedirect(address);
        //svi objekti su sacuvani u sesiji
        //pa je bolja varijanta sa sendRedirect
    }
    @Override
    protected void doGet(HttpServletRequest request,
                         HttpServletResponse response)
            throws ServletException, IOException {
        processRequest(request, response);
    }
    @Override
    protected void doPost(HttpServletRequest request,
                          HttpServletResponse response)
            throws ServletException, IOException {
        processRequest(request, response);
    }
}
```

//source file: beans/ProblemBean.java

```
package beans;
public class ProblemBean {

    private int[] sequence;
    private int solution;

    /**
     * Kreiranje nove instance ProblemBean
     */
    public ProblemBean() {
    }

    public ProblemBean(int[] sequence, int solution) {
        this.sequence = new int[sequence.length];

        for (int i = 0; i < sequence.length; i++) {
            this.sequence[i] = sequence[i];
        }

        this.solution = solution;
    }

    // PROPERTY: sequence
    public int[] getSequence() {
        return sequence;
    }

    public void setSequence(int[] sequence) {
        this.sequence = sequence;
    }
```

```
    // PROPERTY: solution
    public int getSolution() {
        return solution;
    }

    public void setSolution(int solution) {
        this.solution = solution;
    }

    public String toString() {
        String result = "";

        for (int i = 0; i < sequence.length; i++) {
            result += sequence[i] + " ";
        }

        return result;
    }
}
```

//source file: beans/QuizBean.java

```
package beans;
import java.util.ArrayList;

public class QuizBean {

    private ArrayList<ProblemBean> problems = new ArrayList<ProblemBean>();
    private int currentIndex;
    private int score;

    /** pravljenje nove instance QuizBean */
    public QuizBean() {
        problems.add(new ProblemBean(new int[]{1, 4, 9, 16, 25}, 36));
        // kvadrati
        problems.add(new ProblemBean(new int[]{1, 1, 2, 3, 5}, 8));
        // fibonaci
        problems.add(new ProblemBean(new int[]{3, 1, 4, 1, 5}, 9));
        // pi
        problems.add(new ProblemBean(new int[]{2, 3, 5, 7, 11}, 13));
        // prosti
        problems.add(new ProblemBean(new int[]{1, 2, 4, 8, 16}, 32));
        // stepeni dvojke
        problems.add(new ProblemBean(new int[]{0, 7, 26, 63, 124}, 215));
        // kubovi -1

        currentIndex = 0;
        score = 0;
    }

    // PROPERTY: problems
    public void setProblems(ArrayList problems) {
        this.problems = problems;
        currentIndex = 0;
        score = 0;
    }
```

```
    // PROPERTY: score
    public int getScore() {
        return score;
    }

    public void incrementScore() {
        score++;
    }

    // PROPERTY: currentIndex
    public int getCurrentIndex() {
        return currentIndex;
    }

    public void setCurrentIndex(int currentIndex) {
        this.currentIndex = currentIndex;
    }

    // PROPERTY: currentProblem
    public ProblemBean getCurrentProblem() {
        return problems.get(currentIndex);
    }

    public boolean checkAnswer(int answer) {
        return getCurrentProblem().getSolution() == answer;
    }

    public boolean isFinished() {
        return currentIndex == problems.size();
    }
}
```

//web page: quiz.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE HTML>
<html>
    <head>
        <title> * * * Number Quiz * * *</title>
    </head>
    <body>
        <jsp:useBean id="quizBean" type="beans.QuizBean" scope="session"/>

        <form name="quiz" action="QuizHandler" method="post">
            <h3>Uzivajte u nasem quiz-u</h3>
            <p> Vas trenutni rezultat je:
                <jsp:getProperty name="quizBean" property="score"/>
            </p>
            <p> Pogodite sledeci broj u nizu! </p>
            <p><jsp:getProperty name="quizBean" property="currentProblem"/>
            </p>
            <p> Vas odgovor: <input type="text" name="answer"></p>
            <p> <input type="submit" value=" sledeci "></p>
        </form>
    </body>
</html>
```

//web page: score.jsp

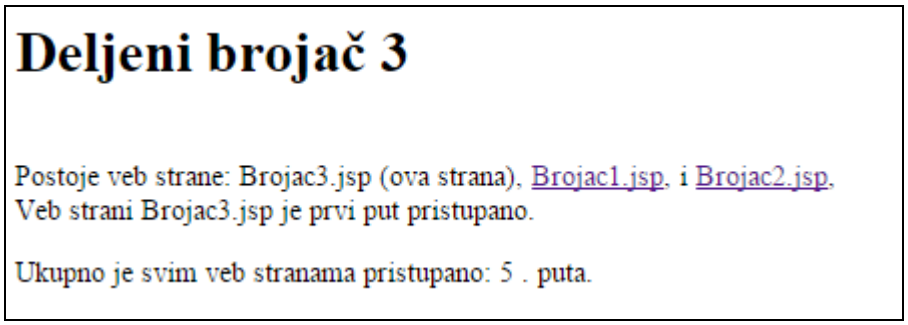
```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE HTML>
<html>
    <head>
        <title> * * * Number Quiz * * *</title>
    </head>

    <body>
        <jsp:useBean id="quizBean" type="beans.QuizBean" scope="session"/>

        <h3> Hvala sto ste igrali nas kviz. </h3>
        <p> Vas rezultat je:
            <jsp:getProperty name="quizBean" property="score"/>
        </p>
        <br/>
        Kliknite <a href="QuizHandler"> ovde </a>
         ako zelite ponovo da ucestvujete u kvizu.

    </body>
</html>
```

//XML file: WEB-INF/web.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app        version="2.5"        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_2_5.xsd">

    <servlet>
        <servlet-name>QuizHandlerServlet</servlet-name>
        <servlet-class>servlets.QuizHandlerServlet</servlet-class>
    </servlet>
    <servlet-mapping>
        <servlet-name>QuizHandlerServlet</servlet-name>
        <url-pattern>/QuizHandler</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>
```

Primer 14 - Brojači

Korišćenjem Java Server Pages tehnologije napraviti tri JSP strane Brojac1.jsp, Brojac2.jsp i Brojac3.jsp, kojima korisnik naizmenično pristupa, ali tako da se u Java binu (bean) čuva podatak o prvoj pristupanoj strani i o ukupnom broju pristupanja svim stranama.



Rešenje:

```
//web page: Brojac1.jsp

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE HTML>

<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Deljeni brojač 1</title>
    </head>
    <body>
        <!-- Obratiti paznju na startnu stranicu u web.xml-->
        <H1>Deljeni brojač 1</H1><br/>
            <jsp:useBean id="counter"
                         class="beans.AccessCountBean"
                         scope="application">
            <!--Uslovno se izvrsava, samo onda kada se bean zaista kreira-->
            <jsp:setProperty name="counter"
                             property="firstPage"
                             value="Brojac1.jsp" />
        </jsp:useBean>
        Postoje veb strane: Brojac1.jsp (ova strana),
        <A HREF="Brojac2.jsp">Brojac2.jsp</A>, i
        <A HREF="Brojac3.jsp">Brojac3.jsp</A>,
        <BR/>
        Veb strani <jsp:getProperty name="counter" property="firstPage" />
        je prvi put pristupano.
        <P>
            Ukupno je svim stranicama pristupano:
            <jsp:getProperty name="counter" property="accessCount" />
            . puta.
            <jsp:setProperty name="counter"
                             property="accessCount"
                             value="1" />
```

```
    </BODY>
</HTML>
```

//web page: Brojac2.jsp

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

<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Deljeni brojač 2</title>
    </head>
    <body>
        <!-- Obratiti paznju na startnu stranicu u web.xml-->
        <H1>Deljeni brojač 2</H1><br/>
            <jsp:useBean id="counter"
                         class="beans.AccessCountBean"
                         scope="application">
            <!--Uslovno se izvrsava, samo onda kada se bean zaista kreira-->
            <jsp:setProperty name="counter"
                             property="firstPage"
                             value="Brojac2.jsp" />
        </jsp:useBean>
        Postoje veb strane: Brojac2.jsp (ova strana),
        <A HREF="Brojac1.jsp">Brojac1.jsp</A>, i
        <A HREF="Brojac3.jsp">Brojac3.jsp</A>,
        <BR/>
        Veb strani <jsp:getProperty name="counter" property="firstPage" />
        je prvi put pristupano.
        <P>
            Ukupno je svim stranicama pristupano:
            <jsp:getProperty name="counter" property="accessCount" />
            . puta.
            <jsp:setProperty name="counter"
                             property="accessCount"
                             value="1" />
    </BODY>
</HTML>
```

//web page: Brojac3.jsp

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

<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Deljeni brojač 3</title>
    </head>
    <body>
        <!-- Obratiti paznju na startnu stranicu u web.xml-->
        <H1>Deljeni brojač 3</H1><br/>
            <jsp:useBean id="counter"
                         class="beans.AccessCountBean"
                         scope="application">
```

```
            <!--Uslovno se izvrsava, samo onda kada se bean zaista kreira-->
            <jsp:setProperty name="counter"
                             property="firstPage"
                             value="Brojac3.jsp" />
        </jsp:useBean>
        Postoje veb strane: Brojac3.jsp (ova strana),
        <A HREF="Brojac1.jsp">Brojac1.jsp</A>, i
        <A HREF="Brojac2.jsp">Brojac2.jsp</A>,
        <BR/>
        Veb strani <jsp:getProperty name="counter" property="firstPage" />
        je prvi put pristupano.
        <P>
            Ukupno je svim veb stranama pristupano:
            <jsp:getProperty name="counter" property="accessCount" />
            . puta.
            <jsp:setProperty name="counter"
                             property="accessCount"
                             value="1" />
    </BODY>
</HTML>
```

//source file: beans/AccessCountBean.java

```
package beans;

public class AccessCountBean {
      //objekat koji se cuva u okviru objekta application
      // i traje do restartovanja web servera

    private String firstPage;
    private int accessCount = 1;

    public String getFirstPage() {
        return (firstPage);
    }

    public void setFirstPage(String firstPage) {
        this.firstPage = firstPage;
    }

    public int getAccessCount() {
        return (accessCount);
    }

    public void setAccessCount(int increment) {
        accessCount = accessCount + increment;
    }
}
```

//XML file: WEB-INF/web.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app        version="2.5"        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_2_5.xsd">
    <session-config>
        <session-timeout>
            30
        </session-timeout>
    </session-config>
    <welcome-file-list>
        <welcome-file>Brojac3.jsp</welcome-file>
        </welcome-file-list>
    </web-app>
```

Primer 15 - Korisnici (MVC)

Korišćenjem Java Servlet i Java Server Pages tehnologija napraviti JSP stranu koja prikazuje formu za logovanje korisnika, u koju korisnik treba da unese svoje korisničko ime i svoju lozinku (kredencijali), i ukoliko su korisnički kredencijali ispravni (postoje u bazi podataka u tabeli *KorisnikInfo*) korisniku prikazati stranicu sa svojim ličnim podacima (JSP strana *prikaz.jsp*), a ukoliko podaci nisu dobri korisniku prikazati stranicu sa greškom (JSP strana *error.jsp*). Takođe, potrebno je realizovati još dve JSP strane: za promenu ličnih podataka (JSP strana *promena.jsp*) i strana sa porukom „Doviđenja“ kada korisnik pritisne dugme za odjavljivanje iz sistema (JSP strana *kraj.jsp*). Rad sa podacima treba realizovati korišćenjem servleta.

Izgled tabela u bazi *dbkorisnik*:

KorisnikInfo		
#	Naziv kolone	Tip
1	username (PK)	varchar(32)
2	password	varchar(32)
3	first_name	varchar(32)
4	last_name	varchar(32)
5	email	varchar(32)
6	phone	varchar(32)

Rešenje:

```
//source file: util/DB.java

package util;

import java.sql.Connection;
import java.sql.DriverManager;

public class DB {

    private static DB instance;
    private static final int MAX_CON = 5;
    private static final Connection[] bafer = new Connection[MAX_CON];
    private int first = 0, last = 0, free = MAX_CON;

    private DB() { //za MySQL
        try {
            Class.forName("com.mysql.jdbc.Driver");
            for (int i = 0; i < MAX_CON; i++) {
                bafer[i] =
                  DriverManager.getConnection
                  ("jdbc:mysql://localhost:3306/dbkorisnik", "root", "");
            }
        } catch (Exception e) {
            e.printStackTrace();
```

```
        }
    }

    /* za Access
     private DB(){
     try{
     Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
     for(int i=0; i<MAX_CON; i++)
     bafer[i] = DriverManager.getConnection("jdbc:odbc:dbkorisnik");
     }catch(Exception e){}
     }
     */
    public static DB getInstance() {
        if (instance == null) {
            instance = new DB();
        }
        return instance;
    }

    public synchronized Connection getConnection() {
        if (free == 0) {
            return null;
        }
        free--;
        Connection con = bafer[first];
        first = (first + 1) % MAX_CON;
        return con;
    }

    public synchronized void putConnection(Connection con) {
        if (con == null) {
            return;
        }
        free++;
        bafer[last] = con;
        last = (last + 1) % MAX_CON;
    }
}
```

//source file: beans/Korisnik.java
//Java bean

```
package beans;

public class Korisnik {

    private String username = "";
    private String password = "";
    private String email = "";
    private String telefon = "";
    private String ime = "";
    private String prezime = "";

    public String getUsername() {
        return username;
    }
```

```
    public void setUsername(String username) {
        this.username = username;
    }

    public String getPassword() {
        return password;
    }

    public void setPassword(String password) {
        this.password = password;
    }

    public String getEmail() {
        return email;
    }

    public void setEmail(String email) {
        this.email = email;
    }

    public String getTelefon() {
        return telefon;
    }

    public void setTelefon(String telefon) {
        this.telefon = telefon;
    }

    public String getIme() {
        return ime;
    }

    public void setIme(String ime) {
        this.ime = ime;
    }

    public String getPrezime() {
        return prezime;
    }

    public void setPrezime(String prezime) {
        this.prezime = prezime;
    }
}

//source file: servleti/Login.java
//servlet za logovanje korisnika u sistem

package servleti;

import beans.*;
import java.io.*;
import java.sql.*;
import javax.servlet.*;
import javax.servlet.http.*;
import util.DB;
```

```
public class Login extends HttpServlet {

    protected void doPost(HttpServletRequest request,
                          HttpServletResponse response)
            throws ServletException, IOException {
        HttpSession sesija = request.getSession();
        String poruka = "";
        String username = (String) request.getParameter("username");
        String password = (String) request.getParameter("password");
        Korisnik korisnik = new Korisnik();
        korisnik.setUsername(username);
        korisnik.setPassword(password);
        sesija.setAttribute("korisnik", korisnik);
        if (username.isEmpty() || password.isEmpty()) {
            poruka = "Niste popunili sva polja!";
            request.setAttribute("poruka", poruka);
            RequestDispatcher rd = request.getRequestDispatcher("/index.jsp");
            rd.forward(request, response);
        }
        Connection con = null;
        Statement stmt = null;
        ResultSet rs = null;
        String address = "prikaz.jsp";
        try {
            con = DB.getInstance().getConnection();
            stmt = con.createStatement();
            String upit = "select * from KorisnikInfo where username='" +
                          username + "' and password='" + password + "';";
            rs = stmt.executeQuery(upit);
            if (rs.next()) {
                String email = rs.getString("email");
                String ime = rs.getString("first_name");
                String prezime = rs.getString("last_name");
                String telefon = rs.getString("phone");
                korisnik.setEmail(email);
                korisnik.setIme(ime);
                korisnik.setPrezime(prezime);
                korisnik.setTelefon(telefon);
                stmt.close();
                request.setAttribute("poruka", "Vasi podaci");
            } else {
                poruka = "Neispravno korisnicko ime i lozinka!
                          Pokusajte ponovo.";
                request.setAttribute("poruka", poruka);
                korisnik.setPassword("");
                address = "index.jsp";
                stmt.close();
            }

        } catch (SQLException ex) {
            sesija.invalidate();
            String errormsg = ex.getMessage();
            request.setAttribute("errormsg", errormsg);
            address = "error.jsp";
        } finally {
            DB.getInstance().putConnection(con);
        }
```

```
        RequestDispatcher rd = request.getRequestDispatcher(address);
        rd.forward(request, response);
    }
}
```

//source file: servleti/Logout.java
//servlet za izlogovanje korisnika iz sistema

```
package servleti;

import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class Logout extends HttpServlet {

    protected void processRequest(HttpServletRequest request,
                                  HttpServletResponse response)
            throws ServletException, IOException {
        request.getSession().invalidate();
        request.getRequestDispatcher("kraj.jsp").forward(request, response);
    }
    @Override
    protected void doGet(HttpServletRequest request,
                         HttpServletResponse response)
            throws ServletException, IOException {
        processRequest(request, response);
    }
    @Override
    protected void doPost(HttpServletRequest request,
                          HttpServletResponse response)
            throws ServletException, IOException {
        processRequest(request, response);
    }
}
```

//source file: servleti/Izmena.java
//servlet za promenu podataka o korisniku

```
package servleti;

import beans.Korisnik;
import java.io.*;
import java.sql.*;
import javax.servlet.*;
import javax.servlet.http.*;
import util.DB;

public class Izmena extends HttpServlet {

    @Override
    protected void doPost(HttpServletRequest request,
                          HttpServletResponse response)
            throws ServletException, IOException {
        HttpSession sesija = request.getSession();
        Korisnik korisnik = (Korisnik) sesija.getAttribute("korisnik");
```

```
        String ime = (String) request.getParameter("ime");
        String prezime = (String) request.getParameter("prezime");
        String telefon = (String) request.getParameter("telefon");
        String email = (String) request.getParameter("email");

        korisnik.setIme(ime);
        korisnik.setPrezime(prezime);
        korisnik.setTelefon(telefon);
        korisnik.setEmail(email);
        String upit = "update KorisnikInfo "
                + "set first_name='" + ime + "',last_name='" + prezime
                + "', phone='" + telefon + "', email='" + email + "'"
                + "where username='" + korisnik.getUsername() + "';";

        Connection con = null;
        Statement stmt = null;
        String address = "prikaz.jsp";
        try {
            con = DB.getInstance().getConnection();
            stmt = con.createStatement();
            stmt.executeUpdate(upit);
            stmt.close();
            con.close();
        } catch (SQLException ex) {
            sesija.invalidate();
            String errormsg = ex.getMessage();
            request.setAttribute("errormsg", errormsg);
            address = "error";
        } finally {
            DB.getInstance().putConnection(con);
        }
        request.setAttribute("poruka", "Podaci su uspesno izmenjeni");
        RequestDispatcher rd = request.getRequestDispatcher(address);
        rd.forward(request, response);
    }
}
```

//web page: index.jsp
//prva strana za logovanje korisnika

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

<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Korisnička aplikacija</title>
    </head>
    <body>
        <h3>${(poruka!=null) ? poruka : "Dobrodošli!"} </h3><br/><br/>
        <form action="login" method="POST">
            <table>
                <tr>
                    <td>Korisničko ime:</td>
                    <td><input type="text" name="username"
                        value="${korisnik.username}" size="20"/></td>
                </tr>
```

```
                <tr>
                    <td>Lozinka:</td>
                    <td><input type="password" name="password"
                        value="${korisnik.password}" size="20"/></td>
                </tr>
                <tr>
                    <td><input type="submit" value="Ulogujte se"/></td>
                    <td><input type="reset" value="Poništite"/></td>
                </tr>
            </table>
        </form>
    </body>
</html>
```

//web page: kraj.jsp
//poslednja strana kada se korisnik izloguje

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

<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Doviđenja</title>
    </head>
    <body>
        <h1>Doviđenja</h1>
        <a href="index.jsp">Na početak</a>
    </body>
</html>
```

//web page: prikaz.jsp
//prva strana nakon logovanja, koja prikazuje podatke o korisniku

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

<html>
    <head>
        <title>Korisnička aplikacija</title>
    </head>
    <body>
        <h3>${poruka}</h3><br/><br/>
        <table width="40%" border="3">
            <tr><td width="50%"> Korisničko ime:</td>
                <td>${korisnik.username} </td></tr>
            <tr><td>Ime:</td>
                <td>${korisnik.ime}</td></tr>
            <tr><td>Prezime:</td>
                <td>${korisnik.prezime}</td></tr>
            <tr><td>Email:</td>
                <td>${korisnik.email}</td></tr>
            <tr><td>Telefon:</td>
                <td>${korisnik.telefon}</td></tr>
        </table><br/>
        <a href="promena.jsp">Promenite podatke</a>
        <a href="logout">Izlogujte se</a>
```

```
    </body>
</html>
```

```
//web page: promena.jsp
//strana na kojoj korisnik moze promeniti svoje podatke

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE HTML>

<html>
    <head>
        <title>Korisnička aplikacija</title>
    </head>
    <body>
       <h3>Vaši podaci koje možete menjati</h3><br/><br/>
       <form action="izmena" method="post" >
          <table width="40%" border="3">
              <tr><td width="50%">Ime:</td>
              <td><input type="text" name="ime" value="${korisnik.ime}"
                  size="30"/></td></tr>
              <tr><td>Prezime:</td>
              <td><input type="text" name="prezime"
                  value="${korisnik.prezime}" size="30"/></td></tr>
              <tr><td>Email:</td>
              <td><input type="text" name="email" value="${korisnik.email}"
                  size="30"/></td></tr>
              <tr><td>Telefon:</td>
              <td><input type="text" name="telefon"
                  value="${korisnik.telefon}" size="30" /></td></tr>
          </table><br/>
          <input type="submit" value="Prihvatite izmenu"/>
        </form>
     </body>
</html>
```

```
//web page: error.jsp
//strana na kojoj se prikazuje poruka o gresci

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE HTML>

<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Greška</title>
    </head>
    <body>
        <h1>Greška u radu sa bazom podataka</h1>
        <h3>${errormsg}</h3>
    </body>
</html>
```