

# 10<sup>th</sup> homework assignment; JAVA, Academic year 2011/2012; FER

As usual, please see the last page. I mean it! You are back? OK. Here we have six problems for you to solve.

## Introduction

For this homework you will create a simple web application. So start by creating a new Eclipse project just the way we have done it during the last lecture class. Name this application *aplikacija2*. In each problem that follows I will describe a functionality that you are required to add into this application. Once done, you will prepare two files and upload them to Ferko:

- a ZIP archive of your eclipse project,
- a WAR archive of your finished web application that can be deployed and tested.

## Problem 1.

Create a page `index.jsp`. Render on it a link “Background color chooser” that will lead to a new page `colors.jsp`. Render on that page four links with texts `WHITE`, `RED`, `GREEN` and `CYAN`. Clicking each of these links will trigger a servlet mapped to `/setcolor` and will remember the selected color as user's session attribute `pickedBgCol`.

You will use this information for rendering the background for each of web pages in this homework including the `index.jsp` and `colors.jsp`. If there is no information on selected color in users session data, use the white as default.

*Hint:* to specify a background color for a HTML page you can either use appropriate attribute of BODY tag or you can link each page to use a `css` file which you then dynamically create (either by JSP or servlet and in which you dynamically write the color to be used).

## Problem 2.

Allow the user to obtain a table of squares for all integer numbers in a range determined by URL parameters `a` and `b` (if `a` is missing, assume `a=0`; if `b` is missing, assume `b=20`; if `a > b`, swap them; if `b > a+20`, set `b` to `a+20`). This action must be accessible on local URL `/squares` and must be mapped to a servlet that will do all the calculations. Once the data are calculated, forward rendering of a resulting page to a JSP `/WEB-INF/pages/squares.jsp`, by using `request.getRequestDispatcher(...).forward(...)` call. Add a link to this action in `index.jsp` with parameters `a=100` and `b=120`.

## Problem 3.

Create a page `stories/funny.jsp` that contains some not too long but funny story. The color of the font used for stories text must be each time randomly chosen (you can randomly pick some color from predefined selection of colors). Add a link to this in `index.jsp`.

Add a filter that is triggered on every request matching `/stories/*` and that will allow the user to see the story only if current time minute's is even (for example, 2:32 and 5:46 but not in 2:31 or 5:47). On odd minutes it will display an appropriate messages telling that the requested page is currently not available (prepare this message as a separate JSP file).

### **Problem 4.**

Create a page `report.jsp` that contains a heading “*OS usage*”, a paragraph “*Here are the results of OS usage in survey that we completed.*” and with a dynamically created image showing Pie Chart. For the creation of the image you will utilize a servlet mapped to `/reportImage`. The servlet will create the requested image by using `jfreechart` library which is freely available. The pie chart should look like the one in simple tutorial available at address:

<http://www.vogella.com/articles/JFreeChart/article.html>

This tutorial shows how to display the chart using Swing. Your job is to adjust the shown code and to modify it as needed. The libraries that are needed you will have to download and place in `WEB-INF/lib` directory in order to make them available to servlet container, and you will have to include them in your classpath as well in Eclipse (“Add to build path”). Since your sevlet will now generate an image, do not forget to set appropriate response's content type, depending on the image format you will use – I recommend *png*.

### **Problem 5.**

Create an action mapped to `/powers` that accepts a three parameters  $a$  (integer from  $[-100,100]$ )  $b$  (integer from  $[-100,100]$ ) and  $n$  (where  $n \geq 1$  and  $n \leq 5$ ). If any parameter is invalid, you should display appropriate message (forward request to some prepared JSP).

Your action should dynamically create a Microsoft Excel document with  $n$  pages. On page  $i$  there must be a table with two columns. The first column should contain integer numbers from  $a$  to  $b$ . The second column should contain  $i$ -th powers of these numbers.

For a creation of XLS document you should use a library developed as part of Apache POI project which is also freely available for download. There are many tutorials available on Internet. For a quick and simple please check:

<http://www.roseindia.net/answers/viewqa/Java-Beginners/14946-how-to-create-an-excel-file-using-java.html>

Add a link to this action on `index.jsp` with a parameters  $a=1$ ,  $b=100$ ,  $n=3$ .

### **Problem 6.**

Create `/appinfo.jsp` that displays how long is this web application running. In order to do so, create a servlet context listener that adds information when it was called into servlet context's attributes (for example, store the result of `System.currentTimeMillis()`). In your JSP, check what is the current time, look up the stored time by your listener and calculate the difference. Format the duration nicely (for example, instead of writing a huge number of milliseconds, output something like 2 days 11 hours 25 minutes 17 seconds and 342 milliseconds).

Add a link to this action on `index.jsp`.

**Please note.** You can consult with your peers and exchange ideas about this homework *before* you start actual coding. Once you open your IDE and start coding, consultations with others (except with me) will be regarded as cheating. You can not use any of preexisting code or libraries for this homework (whether it is yours old code or someones else). Document your code!

In order to solve this homework, create a blank Eclipse Java Project and write your code inside. Once you are done, export project as a ZIP archive and upload this archive on Ferko before the deadline. Do not forget to lock your upload or upload will not be accepted.