

	Datum	14.12.2022
Informatik Web Application Engineering	Übungsblatt 10	
Thomas Feilhauer	Zu lösen bis 21.12.2022, 8:00 Uhr	

Vorbereitungsaufgabe (in der Übungsstunde zu lösen):

Entwickeln Sie eine Web-Applikation auf Basis von JSP unter Verwendung von JSTL, die in einer Schleife die Zahlen von 1 bis 5 hintereinander ausgibt und anschließend durch eine Komma-separierte Liste der Buchstaben a bis f iteriert und davon die Elemente mit den Indizes 2 bis 4 in einer unordered list ausgibt. Verwenden Sie in der JSPs nur Tags (speziell JSTL-Tags) und JSP-Ausdrücke mit EL, aber keine Scriptlets.

Folgende Web-Seiten können beim Lösen der Aufgabe(n) nützlich sein:

<https://www.javatpoint.com/jstl>

Aufgabe 15: JSP-Tabelle**(3 Punkte)**

Schreiben Sie eine **JSP**, die dynamisch Tabellen generiert. Sehen Sie dabei zwei Input-Felder vor, über die die Anzahl der Spalten und Zeilen (jeweils ≤ 16) der Tabelle eingegeben werden können. Wenn der User das Formular abschickt, verarbeitet die JSP die Anfrage und generiert die Tabelle. In jedem Feld sollen dynamisch die Koordinaten des Feldes (Spalte, Zeile) eingetragen werden. Ferner soll die Hintergrundfarbe der Zelle abhängig von den Zellen-Koordinaten generiert werden. Dabei ist insbesondere zu beachten, dass sich die Hintergrundfarben von benachbarten Zellen unterscheiden sollen.

Setzen Sie in Ihren JSPs die **JSTL** Core Tags ein, insbes. für die Iterationen. Verwenden Sie in den JSPs nur Tags und JSP-Ausdrücke mit EL, aber keine Scriptlets.

Dokumentieren Sie, welchen Web-Container Sie einsetzen und das Deployment ihrer Web-Applikation darauf.

Testen Sie das Deployment der WAR außerhalb der IDE und dokumentieren Sie die resultierende URL, unter der die Einstiegsseite Ihrer Webapplikation vom localhost im Web-Browser aufgerufen werden kann.

Neben dem Source-Code und der Dokumentation reichen Sie bitte ein deploybares WAR-File zusammen mit Ihrer Lösung ein.

Aufgabe 16: Sessions**(5 Punkte)**

Erweitern Sie Ihre *Bibliotheksverwaltungs-Web-Site* um ein **Session-Management**. Dabei soll für alle Benutzer:Innen festgehalten werden, welche Seiten der *Bibliotheksverwaltungs-Web-Site* sie seit ihrem Login angesehen haben. Halten Sie die Titel (entsprechen den Link-Titeln in der Navigationsleiste) der besuchten Seiten zusammen mit der Uhrzeit des jeweiligen Seitenaufrufs in der Reihenfolge des Besuchs in einem Attribut des entsprechenden **impliziten Objekts** fest und zeigen Sie die Titel mit der Uhrzeit dem User an, wenn er den Link „Historie“, der noch in der Navigationsleiste anzulegen ist, anklickt.

Integrieren Sie die vorzunehmenden Auswertungen und Operationen in die bestehenden Java-Komponenten (**Servlets, JSPs, JavaBeans**) Ihrer **MVC-Architektur**. Setzen Sie in Ihren JSPs die **JSTL** Core Tags ein, wo sinnvoll.

Verwenden Sie in den JSPs nur Tags, keine Scriptlets und keinen Java-Code. Greifen Sie aus den JSPs heraus nur lesend auf JavaBeans zu und rufen Sie keine Methoden auf. Alle eingehenden Requests sollten vom Controller-Servlet entgegengenommen und verarbeitet werden. Jeder Response an den Benutzer soll aus einer eigenständigen JSP heraus erzeugt werden.

Verwenden Sie für die Implementierung **JakartaEE 9+** mit dem **jakarta**-Namespace (nicht die Klassen im javax-Namespace).

Dokumentieren Sie, welchen Web-Container Sie einsetzen und das Deployment ihrer Web-Applikation darauf.

Testen Sie das Deployment der WAR außerhalb der IDE und dokumentieren Sie die resultierende URL, unter der die Einstiegsseite Ihrer Webapplikation vom localhost im Web-Browser aufgerufen werden kann.

Neben dem Source-Code und der Dokumentation reichen Sie bitte ein deploybares WAR-File zusammen mit Ihrer Lösung ein.

Hinweis für Deployment von WebApps mit JSTL unter Tomcat 10:

Download jakarta.servlet.jsp.jstl-api-3.0.0.jar & jakarta.servlet.jsp.jstl-3.0.1.jar:

<https://repo.maven.apache.org/maven2/org/glassfish/web/jakarta.servlet.jsp.jstl/3.0.1/jakarta.servlet.jsp.jstl-3.0.1.jar>

<https://repo.maven.apache.org/maven2/jakarta/servlet/jsp/jstl/jakarta.servlet.jsp.jstl-api/3.0.0/jakarta.servlet.jsp.jstl-api-3.0.0.jar>

Stellen Sie jakarta.servlet.jsp.jstl-api-3.0.0.jar & jakarta.servlet.jsp.jstl-3.0.1.jar für das Deployment in das /WEB-INF/lib Verzeichnis der zu deployenden Web-Applikation ein.

Dies ist die letzte Aufgabe zur *Bibliotheksverwaltungs-Web-Site*.