

JAVA TECHNOLÓGIÁK 2009.

5. GYAKORLAT

Cél:

- JSP komponensek: *szkriptletek* `<% Java kód %>` és *kifejezések* `<%= Java kifejezés %>`
- Dinamikus űrlap készítése JSP komponenssel

Folytassa az előző gyakorlaton elkezdett feladatot!

Az alkalmazáserver tulajdonságaiban tegye lehetővé a HTTP monitorizálást! (Server-->Properties---> Enable HTTP Monitor)

I. Dinamikus űrlap készítése: `add_course.jsp`

- Törölje az `error.jsp` komponenst
- Módosítsa az `add_course` JSP komponenst úgy, hogy hiba esetén az űrlapadatokat feldolgozó szervlet ennek adhassa át a vezérlést. Az űrlap tetején jelenítse meg a hibalistát.
- Módosítsa az `AddNewCourseServlet` szervletet: hiba esetén adja át a vezérlést az `add_course.jsp` lapnak

```
<body>
  <!--Hibák kiirratasa piros szinnel-->
  <font color="red">
    <ul>
      <%
        java.util.List errorMsgs =
          (java.util.List)request.getAttribute("errorMsgs");
        if( errorMsgs != null){
          java.util.Iterator it=errorMsgs.iterator();
          while(it.hasNext()){
            out.println("<li>"+(String)it.next()+"</li>");
          }
        }
      %>
    </ul>
  </font>
  <!--Űrlap megjelenitese -->
  <form action="add_course.do" method="post">
    <%
      String paramName = request.getParameter("name");
      if( paramName == null )paramName = "";
    %>
    Név:<input type="text" name="name" size="50" value = '<%=paramName%>' >
    <br><br>
    <%
      String paramDescription = request.getParameter("description");
      if( paramDescription == null )paramDescription = "";
    %>
    Leírás: <input type="text" name="description" size="100"
      value= '<%=paramDescription%>' ><br><br>
    <%
      String paramPrice = request.getParameter("price");
      if( paramPrice == null )paramPrice = "";
    %>
    Ár : <input type="text" name="price" value = '<%=paramPrice%>'><br><br>
    <input type="submit" value="Elkuld">
  </form>
</body>
</html>
```

JAVA TECHNOLÓGIÁK 2009.

5. GYAKORLAT

II. Tanfolyamok törlése – dinamikus űrlap

Implementálja a tanfolyam törlését. Két fizikai komponens lesz, egy **megjelenítési**, amelyben lehetősége lesz a tanfolyamok kiválasztására, és egy **vezérlési**, amelyben lebonyolítja az adatbázis műveletet a `CourseDAO` modell komponens segítségével.

A **megjelenítési** komponens legyen egy JSP lap, a **vezérlési** komponensnek használjon szervletet. Tegye lehetővé egyszerre több tanfolyam kiválasztását is, használja a *jelölőnégyzet* (*checkbox*) komponenst. Például:

Válassza ki a törlendő tanfolyamokat!

- C++
- Java SE
- Java Web Development

Törles

megjelenítési rész:

- a tanfolyamok listáját adatbázisból kell kivenni és megjeleníteni az űrlapon
- `<input type="checkbox" name="actualName" value="actualValue" />`
- minden egyes jelölőnégyzetnél használja ugyanazt a name mezőnevet: pl. **courses**
- a value mezőbe tegye be a tanfolyam azonosítóját (ID)

vezérlési rész:

- kérdezze le a **courses** paraméter értékeit a `String[] getParameterValues()` metódussal (az értékek száma == kijelölések száma)
- végezze el a fizikai törlést a `CourseDAO` megfelelő metódusa segítségével (előzetesen bővítse a `CourseDAO`-t, tegye lehetővé a törlést)

III. Helyettesítse a `ListAllCoursesServlet` szervletet egy `list_courses` JSP lappal. Végezze el a szükséges módosítást az `index.jsp` lapon is.