

САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ  
ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ

*Кафедра Вычислительной техники*

## **Лабораторная работа №8**

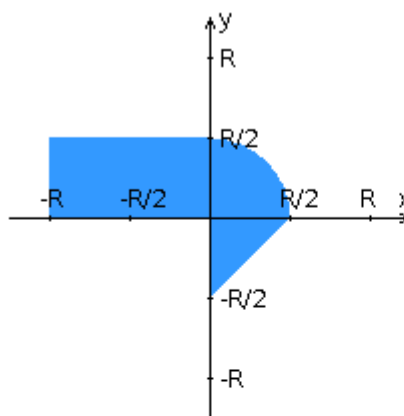
Выполнил:  
студент II курса группы 2125  
Припадчев Артём

Проверит:  
Харитонов А.Е.

Санкт-Петербург  
2014

**Задание:** Доработать приложение из лабораторной работы №7 следующим образом:

- **ControllerServlet** по-прежнему должен определять по URL тип ресурса, к которому обращается клиент и делегировать управление соответствующему сервлету или JSP-странице.
- Вместо WebFormServlet нужно разработать **JSP-страницу**; список требований к ней приведён ниже.
- **AreaCheckServlet** по-прежнему должен осуществлять проверку попадания точки в область на координатной плоскости. Результат проверки должен помещаться в контекст приложения, HTTP-сессию или Bean-компонент в соответствии с вариантом задания.



изменение X: Radio {'-4','-3','-2','-1','0','1','2','3','4'}

изменение Y: Text {-3 ... 5}

изменение R: Button {'1','1.5','2','2.5','3'}

метод HTTP: GET

список результатов сохранять в контексте приложения

**Разработанная JSP-страница должна содержать:**

1. "Шапку", содержащую ФИО студента, номер группы и номер варианта.
2. Форму, отправляющую данные на сервер.
3. Набор полей для задания координат точки и радиуса области в соответствии с вариантом задания.
4. Сценарий на языке JavaScript, осуществляющий валидацию значений, вводимых пользователем в поля формы.
5. Картинку, изображающую область на координатной плоскости (в соответствии с вариантом задания). Если радиус области установлен, клик курсором мыши по этой картинке должен обрабатываться JavaScript-функцией, определяющей координаты точки, по которой кликнул пользователь. В противном случае, после клика по картинке должно выводиться сообщение о невозможности определения координат точки.
6. Таблицу с результатами предыдущих проверок. Список результатов должен браться из контекста приложения, HTTP-сессии или Bean-компонента в зависимости от варианта.

Разработанное веб-приложение необходимо развернуть на сервере [GlassFish](https://www.glassfish.org/).

## Исходный код

### request.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
  <head>
    <script language="JavaScript">
      function validateForm()
      {
        var val = parseInt(document.getElementById('y').value);
        if(val != document.getElementById('y').value)
        {
          window.alert("Incorrect type of Y!");
          return false;
        }
        if (val < -3 || val > 5)
        {
          window.alert("Y [-3;5]");
          return false;
        }
        return true;
      }
    </script>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Lab Room 7</title>
  </head>
  <body>
    <div align="center" style="color:white; background-color: blue; padding: 3px">
      <h3>Pripadchev Artyom <br/>
      group 2125 <br/>
      var. 21261
      </h3>
    </div>
    <br/>
    <table border="1" id="table1" cellspacing="1" cellpadding="1" align = "center">
      <tr>
        <td rowspan="2" valign="top"></td>
        <td valign="top" align = "center" height="20">Параметры</td>
      </tr>
      <tr>
        <td align = "center">
          <form method="POST" onsubmit="return validateForm()">
            <div align="left">
              x =
              <input type="radio" name="x" value="-4" checked>
              <input type="radio" name="x" value="-3">
              <input type="radio" name="x" value="-2">
              <input type="radio" name="x" value="-1">
              <input type="radio" name="x" value="0">
              <input type="radio" name="x" value="1">
              <input type="radio" name="x" value="2">
            </div>
          </form>
        </td>
      </tr>
    </table>
  </body>
</html>
```



```

    }
    //Определяем границы объекта, в нашем случае картинки.
    y0=document.getElementById("table1").offsetTop ;
    x0=document.getElementById("table1").offsetLeft ;
    // Пересчитываем координаты и выводим их алертом.
    x = x-x0 - 2;
    y = y-y0 - 2;
    x = (x-110)/80 * document.getElementById('r').value;
    y = (110 - y)/80 * document.getElementById('r').value;
    alert('x = ' + x.toFixed(3) + ' ' + 'y = ' + y.toFixed(3));
}
</script>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>Lab Room 7</title>
</head>
<body>
<div align="center" style="color:white; background-color: blue; padding: 3px">
  <h3>Pripadchev Artyom <br/>
  group 2125 <br/>
  var. 21258
  </h3>
</div>
<br/>
  <table border="1" id="table1" cellspacing="1" cellpadding="1" align = "center">
    <tr>
      <td rowspan="2" valign="top"></td>
      <td valign="top" align = "center" height="20">Параметры</td>
    </tr>
    <tr>
      <td align = "center">
        <input name="r" value="<%=request.getAttribute("r")%>" style="width: 34px;"
        id="r" type="hidden" />
        <br/>
        x = <%=request.getAttribute("x")%> <br />
        y = <%=request.getAttribute("y")%> <br />
        R = <%=request.getAttribute("r")%><br />
        Result: <%=request.getAttribute("result")%> <br />
      <br/>
      Последние результаты:
      <table border="1" cellspacing="0" cellpadding="5" align = "center">
        <tr>
          <td>x</td>
          <td>y</td>
          <td>R</td>
          <td>Result</td>
        </tr>
        <tr>
          <td><%=
for(int i=0; i<ControllerServlet.results.size();i++)
{ %>
          <tr>
          <td><%=ControllerServlet.results.get(i).x%></td>

```

```

                <td><%=ControllerServlet.results.get(i).y%></td>
                <td><%=ControllerServlet.results.get(i).R%></td>
                <td><%=ControllerServlet.results.get(i).res%></td>
            </tr>
            <% }%>
        </table>
    </td>
</tr>
</table>
<br />
<div align="center"><a href="/Lab7/home">Назад</a></div>
</body>
</html>

```

### AreaCheckServlet

```
package controller;
```

```
import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
```

```
@WebServlet(name = "AreaCheckServlet", urlPatterns = {"/response"})
public class AreaCheckServlet extends HttpServlet {
    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        String x = request.getParameter("x");
        String y = request.getParameter("y");
        String r = request.getParameter("r");
        response.setContentType("text/html");
        request.setAttribute("x", x);
        request.setAttribute("y", y);
        request.setAttribute("r", r);
        Mark mark = new Mark(Double.parseDouble(x),Double.parseDouble(y));
        Area area = new Area(Double.parseDouble(r));
        String result;
        if(area.hitInArea(mark))
            result = "IN";
        else
            result = "OUT";
        ControllerServlet.results.add(new Result(x.toString(),y.toString(),r.toString(),result));
        request.setAttribute("results", ControllerServlet.results);
        request.setAttribute("result", result);
        RequestDispatcher dispatcher = request.getRequestDispatcher("WEB-INF/response.jsp");

        if (dispatcher != null) {

```

```
dispatcher.forward(request, response);
```

```
}
```

```
}
```

```
}
```

**Вывод:** в процессе выполнения работы познакомился с возможностями jsp-страниц, контекстом сервлета, а также с базовыми основами работы JS.