Университет ИТМО

Кафедра вычислительной техники

Программирование интернет-приложений

Лабораторная работа № 7

Вариант 1732

Выполнил:

Сорокин Юрий

Группа Р3217

Преподаватель:

Харитонова А.Е.

СПб

2016 г.

### Лабораторная работа #7

Разработать веб-приложение на базе сервлетов, определяющее попадание точки на координатной плоскости в заданную область.

Приложение должно включать в себя 3 сервлета:

* **ControllerServlet**, определяющий тип запроса, и, в зависимости от того, содержит ли запрос информацию о координатах точки и радиусе, делегирующий его обработку одному из последующих сервлетов. Все запросы внутри приложения должны передаваться этому сервлету (по методу GET или POST в зависимости от варианта задания), остальные сервлеты с веб-страниц напрямую вызываться не должны.
* **WebFormServlet**, формирующий HTML-страницу с веб-формой. Должен обрабатывать все запросы, не содержащие сведений о координатах точки и радиусе области.
* **AreaCheckServlet**, осуществляющий проверку попадания точки в область на координатной плоскости и формирующий HTML-страницу с результатами проверки. Должен обрабатывать все запросы, содержащие сведения о координатах точки и радиусе области.

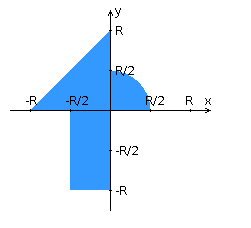
**Страница, возвращаемая WebFormServlet, должна содержать:**

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. Любые некорректные значения (например, буквы в координатах точки или отрицательный радиус) должны блокироваться.

**Страница, возвращаемая AreaCheckServlet, должна содержать:**

1. Таблицу, содержащую полученные параметры.
2. Результат вычислений - факт попадания или непопадания точки в область.
3. Ссылку на страницу с веб-формой для формирования нового запроса.

Разработанное веб-приложение необходимо развернуть на сервере [GlassFish](http://glassfish.java.net/" \t "_blank).



public class ControllerServlet extends HttpServlet {

@Override

protected void doGet(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {

req.getRequestDispatcher("/view").forward(req, resp);

}

@Override

protected void doPost(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {

if(req.getParameter("YSelector") != null) req.getRequestDispatcher("/check").forward(req,resp);

else req.getRequestDispatcher("/view").forward(req,resp);

}

}

public class WebFormServlet extends HttpServlet {

@Override

protected void doGet(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {

req.setCharacterEncoding("UTF-8");

resp.setCharacterEncoding("UTF-8");

PrintWriter out = resp.getWriter();

out.print("<!DOCTYPE html>\n" +

"<html lang=\"en\">\n" +

"<head>\n" +

" <meta charset=\"UTF-8\">\n" +

" <title>ПИП Лаба №7</title>\n" +

"</head>\n" +

"<body>\n" +

" <header>\n" +

" <p>Павлишин Александр Андреевич</p>\n" +

" <p>Группа P3217</p>\n" +

" <p>Вариант 1728</p>\n" +

" </header>\n" +

" <form method=\"POST\" id=\"forma\" action=\"/lab7/main\">\n" +

" <p>X value :\n" +

" <input type=\"checkbox\" name=\"x1\" value=\"-5\">-5 " +

" <input type=\"checkbox\" name=\"x2\" value=\"-4\">-4" +

" <input type=\"checkbox\" name=\"x3\" value=\"-3\">-3" +

" <input type=\"checkbox\" name=\"x4\" value=\"-2\">-2" +

" <input type=\"checkbox\" name=\"x5\" value=\"-1\">-1" +

" <input type=\"checkbox\" name=\"x6\" value=\"0\">0" +

" <input type=\"checkbox\" name=\"x7\" value=\"1\">1" +

" <input type=\"checkbox\" name=\"x8\" value=\"2\">2" +

" <input type=\"checkbox\" name=\"x9\" value=\"3\">3" +

" </p>\n" +

" <p>Y value {-3..3}:\n" +

" <input type=\"text\" id=\"textY\" name=\"YSelector\" placeholder=\"-3..3\" onchange=\"validateForm()\" onkeydown=\"javascript:if(13==event.keyCode){return false;}\" required>\n" +

" </p>\n" +

" <p>R value:\n" +

" <input type=\"submit\" name=\"r1\" value=\"1\">\n" +

" <input type=\"submit\" name=\"r2\" value=\"2\">\n" +

" <input type=\"submit\" name=\"r3\" value=\"3\">\n" +

" <input type=\"submit\" name=\"r4\" value=\"4\">\n" +

" <input type=\"submit\" name=\"r5\" value=\"5\">\n" +

" </p>\n" +

" </form>\n" +

" <img src=\"http://i.imgur.com/39S9xWy.png\">\n" +

"</body>\n" +

"</html>\n" +

"<script>\n" +

" function validateForm(){\n" +

" var elementList = document.querySelectorAll(\"input[type=submit]\");\n"+

" var y\_val=document.getElementById(\"textY\").value;\n" +

" y\_val=y\_val.replace(\",\",\".\");\n" +

" var y\_valid=!((y\_val == \"\") || !(!isNaN(parseFloat(y\_val)) && isFinite(y\_val)) || (y\_val > 3) || (y\_val < -5));\n" +

" if (y\_valid) {\n" +

" for(i=0;i<elementList.length;i++) elementList[i].disabled=false; \n" +

" }else{\n" +

" for(i=0;i<elementList.length;i++) elementList[i].disabled=true; \n" +

" alert(\"Check Y field please\"); \n" +

" }\n" +

" }\n" +

"</script>\n");

out.close();

}

}

public class AreaCheckServlet extends HttpServlet {

@Override

protected void doPost(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {

req.setCharacterEncoding("UTF-8");

resp.setCharacterEncoding("UTF-8");

String YBuf = req.getParameter("YSelector").replace(',', '.');

ArrayList<String> XBuf = new ArrayList<String>(10);

String RBuf = " ";

PrintWriter out = resp.getWriter();

for(int i=1;i<=9;i++) if (req.getParameter("x" + i) != null) XBuf.add(req.getParameter("x" + i));

for(int i=1;i<=5;i++) if (req.getParameter("r" + i) != null) RBuf = req.getParameter("r" + i);

float Y = Float.NaN;

float R = Float.NaN;

ArrayList<Float> X = new ArrayList<Float>(10);

int size = XBuf.size();

try {

Y = Float.parseFloat(YBuf);

R = Float.parseFloat(RBuf);

for(int k=0;k<size;k++)

X.add(Float.parseFloat(XBuf.get(k)));

} catch (NumberFormatException e) {

out.println(e);

}

//for(int j=0;j<size;j++)

//out.println(XBuf.get(j));

for(int j=0;j<size;j++) {

if (!validate(X.get(j), Y, R)) {

out.print("<!DOCTYPE html>\n" +

"<html lang=\"en\">\n" +

"<head>\n" +

" <meta charset=\"UTF-8\">\n" +

" <title>ПИП Лаба №7</title>\n" +

"</head>\n" +

"<body>\n" +

" <p>Неверные данные</p>\n" +

" <p>X должен быть целым числом {-5..3}. X"+j+"="+X.get(j)+"</p>\n" +

" <p>Y должен быть числом {-3..3}. Y=" + Y + "</p>\n" +

" <p>R должен быть положительным целым числом {1..5} . R="+R+"</p>\n" +

"</body>\n" +

"</html>");

out.close();

return;

}

}

out.print("<!DOCTYPE html>\n" +

"<html lang=\"en\">\n" +

"<head>\n" +

" <meta charset=\"UTF-8\">\n" +

" <title>ПИП Лаба №7</title>\n" +

"</head>\n" +

"<body>\n" +

" <img src=\"http://i.imgur.com/39S9xWy.png\">\n" +

"\t\t<table align='left'>\n" +

"\t\t <tr>\n" +

"\t\t <th>X: </th>\n");

for(int j=0;j<size;j++) out.print("\t\t <td>"+X.get(j)+"</td>\n");

out.print("\t\t </tr>\n" +

"\t\t <tr>\n" +

"\t\t <th>Y: </td>\n");

for(int j=0;j<size;j++) out.print("\t\t <td>"+Y+"</td>\n");

out.print("\t\t </tr>\n" +

"\t\t <tr>\n" +

"\t\t <th>R: </td>\n");

for(int j=0;j<size;j++) out.print("\t\t <td>"+R+"</td>\n");

out.print("\t\t <tr>\n" +

"\t\t <td>In area: </td>\n");

for(int j=0;j<size;j++) out.print("\t\t <td>" + (inFigure(X.get(j), Y, R)? "Yes" : "No") +"</td>\n");

out.print("\t\t </tr>\n" +

"\t\t</table>\n" +

"\t\t<p align=\"left\"><a href=\"/lab7/main\">Вернуться</a></p>"+

"</body>\n" +

"</html>");

out.close();

}

boolean inFigure(float X, float Y, float R) {

return (Y >= 0 && X <= 0 && (Y\*Y + X\*X <= R\*R/4)) ||

(Y >= -R / 2 && Y <= 0 && X <= R / 2 && X >= 0 ) ||

(Y >= 0 && X <= 0 && Y <= X + R / 2);

}

boolean validate(float X, float Y, float R) {

boolean Xtrue = (!Float.isNaN(X) && ((int)X == X) && (X >= -5) && (X <= 3));

boolean Rtrue = (((int)R == R) && (R >= 1) && (R <= 5) && !Float.isNaN(R));

boolean Ytrue = (!Float.isNaN(Y) && (Y <= 3) && (Y >= -3));

return Xtrue && Rtrue && Ytrue;

}

}

**Вывод**

При выполнении данной работы, я ознакомился с работой с сервлетами.