Университет ИТМО

Кафедра ВТ

Программирование Интернет-приложений

Лабораторная работа № 6

Работу выполнил:

Студент 2 курса

Группы № 2125

Назарьев Сергей

Санкт-Петербург

2015 г.

**Цель работы:**

Разработать PHP-скрипт, определяющий попадание точки на координатную плоскость в заданную область и создать HTML-страницу, которая формирует данные для отправки их на обработку этим скриптом.

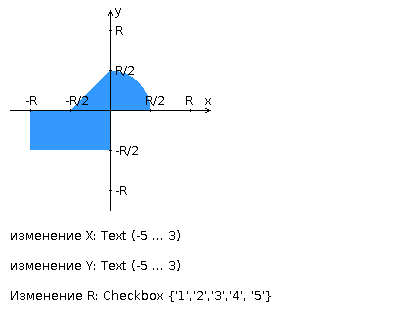
Параметр R и координата точки должны передаваться скрипту посредством HTTP-запроса.Скрипт должен выполнять валидацию данных и возвращать HTML-страницу с таблицей, содержащей полученные параметры и результат вычислений - факт попадания или непопадания точки в область.

Кроме того, ответ должен содержать данные о текущем времени и времени работы скрипта.

**Разработанная HTML-страница должна удовлетворять следующим требованиям:**

* Для расположения текстовых и графических элементов необходимо использовать блочную верстку.
* Данные формы должны передаваться на обработку посредством POST-запроса.
* Таблицы стилей должны располагаться в самом веб-документе.
* При работе с CSS должно быть продемонстрировано использование селекторов дочерних элементов, селекторов идентификаторов, селекторов элементов, селекторов псевдоклассов а также такие свойства стилей CSS, как наследование и каскадирование.
* HTML-страница должна иметь "шапку", содержащую ФИО студента, номер группы и новер варианта. При оформлении шапки необходимо явным образом задать шрифт (fantasy), его цвет и размер в каскадной таблице стилей.
* Отступы элементов ввода должны задаваться в пикселях.

**Вопросы к защите лабораторной работы:**

1. Протокол HTTP. Структура запросов и ответов, методы запросов, коды ответов сервера, заголовки запросов и ответов.
2. Язык разметки HTML. Особенности, основные теги и атрибуты тегов.
3. Структура HTML-страницы. Объектная модель документа (DOM).
4. HTML-формы. Задание метода HTTP-запроса. Правила размещения форм на страницах, виды полей ввода.
5. Каскадные таблицы стилей (CSS). Структура - правила, селекторы. Виды селекторов, особенности их применения. Приоритеты правил. Преимущества CSS перед непосредственным заданием стилей через атрибуты тегов.
6. CGI - определение, назначение, ключевые особенности.
7. FastCGI - особенности технологии, преимущества и недостатки относительно CGI.
8. Язык PHP - синтаксис, типы данных, встраивание в веб-страницы, правила обработки HTTP-запросов. Особенности реализации принципов ООП в PHP. 

**Код программы:**

<?php  
function placeholders($filename, $placeholders)  
{  
    return strtr(file\_get\_contents($filename), $placeholders);  
}  
  
$time  = microtime();  
$time  = explode(' ', $time);  
$time  = $time[1] + $time[0];  
$start = $time;  
  
if (!isset($\_POST['submit'])) {  
    echo placeholders("index.tpl", array(  
        "{{ERROR}}" => ""  
    ));  
    exit();  
}  
  
$r = (is\_numeric($\_POST['r']) ? floatval($\_POST['r']) : NULL);  
$x = (is\_numeric($\_POST['x']) ? floatval($\_POST['x']) : NULL);  
$y = (is\_numeric($\_POST['y']) ? floatval($\_POST['y']) : NULL);  
  
if ($r == NULL || $x == NULL || $y == NULL || $x < -5 || $x > 3 || $y < -5 || $y > 3) {  
    echo placeholders("index.tpl", array(  
        "{{ERROR}}" => "Validation failed! (x [-5..3], y [-5..3], r [1,2,3,4,5])"  
    ));  
    exit;  
}  
  
$rh     = $r / 2;

$result =  ($x >= -$r && $x <= 0 && $y >= -$rh && $y <= 0)

|| ($x >= 0 && $x <= $rh && (($x \* $x + $y \* $y) <= $rh \* $rh))

|| ($x < 0 && $y > 0 && $y < $x + $rh);  
  
$time       = microtime();  
$time       = explode(' ', $time);  
$time       = $time[1] + $time[0];  
$finish     = $time;  
$total\_time = round(($finish - $start), 4);  
  
echo placeholders("result.tpl", array(  
    "{{CURRENT\_TIME}}" => date('Y-m-d H:i:s'),  
    "{{LOAD\_TIME}}" => $total\_time,  
    "{{R}}" => $r,  
    "{{X}}" => $x,  
    "{{Y}}" => $y,  
    "{{RESULT}}" => ($result ? "Попало" : "Не попало")  
));  
  
?>

<!doctype html>

<html>

<head>

<title>Лабораторная работа</title>

<meta charset=utf8 />

<style>

input {

margin: 10px;

}

html, body {

width: 100%;

padding: 0;

margin: 0;

font-family: Tahoma;

}

.left-block {

float: left;

width: 50%;

}

.right-block {

float: right;

width: 50%;

}

.clear {

clear: both;

}

#header p {

text-align: center;

color: red;

font-size: 24px;

font-family: fantasy, sans-serif;

}

p {

background: yellow;

}

</style>

</head>

<body>

<div id="header">

<p>Назарьев Сергей Сергеевич, гр. 2125, вариант 2509</p>

<p>{{ERROR}}</p>

</div>

<form method=POST>

<div class="left-block">

<input type="text" name="x" placeholder="x">

<input type="text" name="y" placeholder="y">

<input type="submit" name="submit" value="Send">

</div>

<div class="right-block">

<input checked type="radio" name="r" id="r1" value="1">

<label for="r1">R = 1</label>

<input type="radio" name="r" id="r2" value="2">

<label for="r2">R = 2</label>

<input type="radio" name="r" id="r3" value="3">

<label for="r3">R = 3</label>

<input type="radio" name="r" id="r4" value="4">

<label for="r4">R = 4</label>

<input type="radio" name="r" id="r5" value="5">

<label for="r5">R = 5</label>

</div>

<div class="clear"></div>

</form>

</body>

</html>

<!doctype html>

<html>

<head>

<title>Лабораторная работа</title>

<meta charset=utf8 />

<style>

html, body {

width: 100%;

padding: 0;

margin: 0;

font-family: Tahoma;

}

.cell {

float: left;

width: 100px;

height: 100px;

border: 1px solid black;

}

.cell:nth-child(even) {

background: red;

}

.cell:nth-child(odd) {

background: yellow;

}

.clear {

clear: both;

}

#header p {

text-align: center;

color: red;

font-size: 24px;

font-family: fantasy, sans-serif;

}

p {

background: yellow;

}

</style>

</head>

<body>

<div id="header">

<p>Назарьев Сергей Сергеевич, гр. 2125, вариант 2509</p>

</div>

<div><div class="cell">

X

</div>

<div class="cell">

Y

</div>

<div class="cell">

R

</div>

<div class="cell">

Текущее время

</div>

<div class="cell">

Время загрузки

</div>

<div class="cell">

Результат

</div></div>

<div class="clear"></div>

<div><div class="cell">

{{X}}

</div>

<div class="cell">

{{Y}}

</div>

<div class="cell">

{{R}}

</div>

<div class="cell">

{{CURRENT\_TIME}}

</div>

<div class="cell">

{{LOAD\_TIME}}

</div>

<div class="cell">

{{RESULT}}

</div></div>

<div class="clear"></div>

</body>

</html>

**Вывод:**

В ходе данной лабораторной работы мне посчастливилось приобщиться к программированию на PHP и к вёрстке на HTML /CSS, были изучены основные принципы взаимодействия CSS и HTML, работа HTTP-запросов и способ обработки HTTP-запросов на стороне сервера.