

2-й семестр (весна)

Лабораторная работа #6

Внимание! У разных вариантов разный текст задания!

Разработать PHP-скрипт, определяющий попадание точки на координатной плоскости в заданную область, и создать HTML-страницу, которая формирует данные для отправки их на обработку этому скрипту.

Параметр R и координаты 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 непадения 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 табличную верстку.
- Данные формы должны передаваться на обработку посредством GET-запроса.
- Таблицы стилей должны располагаться в самом веб-документе.
- При работе с CSS должно быть продемонстрировано использование селекторов атрибутов, селекторов элементов, селекторов идентификаторов, селекторов классов а также такие свойства стилей CSS, как наследование и каскадирование.
- HTML-страница должна иметь "шапку", содержащую ФИО студента, номер группы и номер варианта. При оформлении шапки необходимо явным образом задать шрифт (cursive), его цвет и размер в каскадной таблице стилей.
- Отступы элементов ввода должны задаваться в процентах.

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.

Вариант 819

