# Каскадная таблица стилей

Скорее всего вы уже слышали о CSS. Давайте просто быстро суммируем информацию.

Во-первых, есть три способа изменения стиля элемента:

1. С помощью атрибута style напрямую после тега элемента. Например <div style = “color: red”>
2. В head документа, внутри тега <style>
3. Подключая дополнительную CSS страницу

Во-вторых, что очень важно, и о чем зачастую забывают все.

Элементы делятся на блочные и строчные. Div – блочный элемент. a, p – строчный. Атрибуты, применимые к блочным элементам не всегда применимы к строчным. Например: vertical-align не применим к тегу div. (Вертикальное выравнивание). Подробнее про блочные и строчные элементы можно почитать тут: <https://html5book.ru/block-inline-elements/#width>

В-третьих, таблица стилей зачастую занимает намного больше строк кода, чем сам html код. Писать его чисто и понятно хотя бы для самих себя – это не только правило хорошего тона, а также экономия будущего времени и энергии. Поэтому, старайтесь правильно подбирать названия классов, id, имена полей и тд. Хорошая статья по чистоте кода: [https://webref.ru/layout/learn-html-css/writing-your-best-](https://webref.ru/layout/learn-html-css/writing-your-best-code) [code](https://webref.ru/layout/learn-html-css/writing-your-best-code)

# Работа с формами HTML

Начало работы с формами на шаг приближает нас к реальному программированию. О том, как взаимодействовать с сервером при помощи формы мы поговорим позже, когда начнем изучать непосредственно backend. А для начала давайте поговорим об элементах форм.

1. Input – бОльшая часть элементов допустима при использовании тега input.

Тег <input> является одним из разносторонних элементов формы и позволяет создавать разные элементы интерфейса и обеспечить взаимодействие с пользователем. Главным образом <input> предназначен для создания текстовых полей, различных кнопок, переключателей и флажков. Хотя элемент <input> не требуется помещать внутрь контейнера <form>, определяющего форму, но если введенные пользователем данные должны быть отправлены на сервер, где их обрабатывает серверная программа, то указывать <form> обязательно. То же

самое обстоит и в случае обработки данных с помощью клиентских приложений, например, скриптов на языке JavaScript.

Основной атрибут тега <input>, определяющий вид элемента — type. Он позволяет задавать следующие элементы формы: текстовое поле (text), поле с паролем (password), переключатель (radio), флажок (checkbox), скрытое поле (hidden), кнопка (button), кнопка для отправки формы (submit), кнопка для очистки формы (reset), поле для отправки файла (file) и кнопка с изображением (image). Для каждого элемента существует свой список атрибутов, которые определяют его вид и характеристики. Кроме того, в HTML5 добавлено еще более десятка новых элементов. (Подробнее об атрибутах тут: <http://htmlbook.ru/html/input>)

1. Textarea – многострочное поле ввода. Требует закрывающий тег.
2. Radiobutton и CheckBox Возможность выбора одного или нескольких вариантов из списка соответственно

При использовании элементов формы есть одна очень весомая проблема – многие из них практически не стилизуются.

# Вопросы:

1. В начале мы говорили о том, что существуют три способа задания стиля элемента. Допустим к элементу div были применены все три способа (информация о его стиле содержится в отдельном файле, внутри head, а также внутри атрибута style). Какой из способов будет преобладать над другими.
2. Как изменить размер textarea
3. Иногда нужно вертикально центрировать текст внутри div. Как это возможно сделать (ответ в файле logib.html)
4. Как добавить всплывающую подсказку при наведении на поле ввода
5. Приведите пример хорошего и плохого кода, написанного на CSS.