



Кафедра «Автоматизированные станочные системы»

Dept. of Automated Manufacturing Systems

ИНТЕРАКТИВНЫЕ WEB-СТРАНИЦЫ КАСКАДНЫЕ ТАБЛИЦЫ СТИЛЕЙ (CSS)

Интерактивной называется Web-страница, позволяющая пользователю вводить данные разного типа.

HTML-формой называют конструкцию языка HTML, позволяющую размещать на WWW-странице стандартные диалоговые элементы Windows, такие как кнопки, поля для ввода данных, флажки, списки, а также указывать процедуру обработки данных этих полей.

В форме можно выделить следующие функциональные части:

- заголовок формы;
- отображаемые поля для ввода данных;
- скрытые (управляющие) поля.

Заголовок формы

Определение любой HTML-формы должно начинаться с заголовка `<form..>`, в которой указывается процедура, выполняемая при обработке данных формы.

Заголовок формы для отправки ее содержимого по email должен иметь вид:

```
<form method="post"  
action="mailto: адрес электронной почты">
```

Здесь значение `post` указывает метод взаимодействия с сервером

Для собственно отправки сообщения на форме, как правило, помещается кнопка, описываемая следующим образом:

```
<input type="submit" value="Отправить">
```

Здесь параметр `value` указывает на текст, выводимый на кнопке

Отображаемые поля

Отображаемыми полями условимся называть элементы формы, которые отображаются на WWW-страничке и служат для ввода данных пользователя. Это могут быть кнопки, списки, поля ввода текста и т.п.

Для каждого поля необходимо задать имя (значение параметра name).

Имя поля может начинаться с набора управляющих букв, называемых ключами программы, которые отделяются от остальной части имени знаком "_". Каждый ключ задает то или иное требование к полю.

Допускается использовать ключи, предписывающие полям следующие свойства:

- **r (required)**: данное поле обязательно к заполнению;
- **d (digits)**: в данное поле можно вводить только цифры и десятичную точку (.);
- **e (email)**: вводимые значения должны иметь форму электронного почтового адреса;
- **w (word)**: в данное поле можно вводить только текст, латинские буквы и цифры (A-Z,a-z,0-9);
- **c (currency)**: в данное поле можно вводить только цифры, десятичную точку и знак "\$";
- **m (multiple)**: данное поле может иметь несколько значений (для радио-кнопок с одним и тем же именем);
- **n (\n)**: в данное поле можно вводить символы перехода на следующую строку (вводятся при нажатии клавиши Enter), которые следует удалять при отправке;
- **s (space)**: символы пробелов в начале и в конце значения поля следует удалять.

Например:

```
<input type="text" name="rd_количество">
```

Данное поле обязательно для заполнения, можно вводить только цифры и точку.

```
<input type="text" name="_address">
```

Данное поле может содержать любой текст, его заполнять необязательно.

Помимо тех полей, которые отображаются на WWW-страничке, в форме можно определить **скрытые**, не показываемые поля, необходимые для управления режимом работы программы обработки формы. Все управляющие поля имеют тип hidden (значения параметра type).

Дизайн форм

Важно обеспечить удобный для пользователя дизайн форм.

Для этих целей можно взять на вооружение теги `fieldset`, `legend` и `label` и атрибуты `tabindex` и `accesskey`.

FIELDSET - тег, позволяющий группировать (в том числе, визуально) элементы формы.

LEGEND - тег, который задает заголовок группе элементов (`fieldset`).

LABEL - способен связывать описания элементов формы с самими элементами. Для этого задается атрибут `for`, содержащий `id` соответствующего элемента. Позволяет «кликом» передать фокус связанному элементу.

`tabindex` - данный атрибут определяет порядок перебора элементов формы при нажатии на клавишу `Tab`.

`accesskey` - задает так называемую «горячую клавишу» (`Alt+буква`).

Каскадные таблицы стилей (CSS)

CSS - это язык, содержащий набор свойств для описания внешнего вида любых HTML документов.

Стиль – набор правил оформления и форматирования, который может быть применен к различным элементам страницы.

Существует три вида таблиц стилей: внутренние таблицы стилей, глобальные таблицы стилей и связанные таблицы стилей.

Внутренние таблицы стилей (Inline Style Sheets) при помощи специального атрибута помещаются прямо в HTML теги.

Глобальные таблицы (Global Style Sheets) определяют стиль элементов во всем документе.

Связанные таблицы (Linked Style Sheets) могут быть использованы для нескольких документов сразу (скажем, для всего сайта) и хранятся во внешнем файле с расширением CSS.

Для присвоения какому-либо элементу определенных характеристик вы должны один раз описать их и определить это описание как стиль, а в дальнейшем просто указывать, что элемент, который вы хотите оформить соответствующим образом, должен принять свойства стиля, описанного вами.

Информация о стилях может располагаться либо в отдельном файле, либо непосредственно в коде Web-странички.

Для описания стиля в отдельном файле нужно создать обычный текстовый файл, описать с помощью языка CSS необходимые стили, разместить этот файл на Web-сервере, а в коде Web-страниц, которые будут использовать стили из этого файла, нужно будет сделать ссылку на него.

Делается это с помощью тега `<LINK>`, располагающегося внутри тега `<BODY>` страниц:

```
<LINK REL=STYLESHEET TYPE="text/css" HREF="styles.css">
```

Пример

CSS-файл styles.css описывает стиль с именем wrong:

```
.wrong  
{text-decoration: line-through;  
  color:green;  
}
```

HTML-файл содержит ссылку на CSS-файл и стиль wrong применен к абзацу:

```
<html><body>  
<LINK REL=STYLESHEET TYPE="text/css" HREF="styles.css">  
<p class="wrong">errors</p>  
</body>  
</html>
```

При описании стилей непосредственно в коде Web-страницы, описание располагается внутри тега <BODY>, в теге

```
<STYLE type="text/css">
```

...

```
</STYLE>
```

В этом случае можно использовать эти стили для элементов, располагающихся в пределах странички. **Пример:**

```
<html><body>
<STYLE type="text/css">
.wrong
{text-decoration: line-through;
 color:green;
}
</STYLE>
<p class="wrong">errors</p>
</body>
</html>
```

Третий вариант - описание стиля располагается непосредственно внутри тега элемента. Это делается с помощью параметра STYLE, используемого при применении CSS с большинством стандартных тегов HTML.

Присвоение стилей

Самый простой случай присвоения какому-либо элементу определенного стиля выглядит так:

НАЗВАНИЕ_ЭЛЕМЕНТА {свойство: значение;},

где НАЗВАНИЕ_ЭЛЕМЕНТА – это имя HTML тега (H1, P, TD, A и т. д.), а параметры в фигурных скобках – список свойств элемента и присвоенных им значений.

Пример:

```
H1 {font-size: 30pt; color: blue;}
```

Элементы страниц, созданные с использованием CSS, используют механизм наследования.

CSS реализует возможность присваивать стили не всем одинаковым элементам страницы, а избирательно – для этого используется параметр **CLASS** = "имя класса" или идентификатор **ID**=«имя элемента», присваивающиеся любому элементу страницы.

Параметр **CLASS** применяется в случае, если необходимо создать одинаковый стиль для нескольких, но не всех элементов страницы (одинаковых или разных).

Присвоение стилей с помощью идентификаторов (параметр **ID**) применяется в случае, если данному идентификатору соответствует только один элемент на странице. Если элементов, которым необходимо присвоить такой стиль, несколько – это уже класс.

Язык CSS насчитывает довольно большое количество свойств элементов HTML, которыми он может управлять:

- Свойства шрифта, например: указание шрифта или шрифтового семейства (font-family), которым будет отображаться элемент; определение степени жирности шрифта (font-weight); установка размера шрифта (font-size) и др.
- Цвет элемента и цвет фона (color, background-color) .
- Свойства текста, например: эффекты оформления шрифта (text-decoration), выравнивание элемента (text-align), отступы (text-indent) и интервалы (line-height) .
- Свойства границ: отступы вокруг элемента (margin-left/ margin-right/ margin-bottom/ margin-top) .
- Единицы измерения (px/ sm/ mm/ pt/ %).

Описание свойств элементов в CSS состоит из названия свойства с последующим присвоением ему определенного значения. Название свойства и его значение разделены двоеточием. Описание завершается точкой с запятой.

Еще один из интересных вариантов применения CSS скрывается за простой возможностью: можно указывать значения отступов вокруг объектов как отрицательные величины, что позволяет накладывать один слой текста на другой и получать весьма интересные результаты.

Например:

```
<HEAD> <STYLE type="text/css">  
BODY  
{font-family: Verdana; font-size: 70pt; font-weight: bold;}  
.z1 { color: silver; margin-top: 100px; margin-left: 70px;}  
.z2 {color: navy; margin-top: -118px; margin-left: 68px;}  
</STYLE> </HEAD>  
<BODY bgcolor=white>  
<DIV class="z1">Кафедра ACC</DIV>  
<DIV class="z2"> Кафедра ACC </DIV> </BODY>
```

В этом описании мы присвоили тегу <BODY> размер, шрифт и начертание – в таком стиле будут отображаться все элементы страницы. Далее мы описываем два стиля, которые отличаются цветом и размером отступов вокруг них: нижний слой описывается стилем z1, а верхний – z2. Используя отрицательные значения отступов и подбирая нужное значение, мы добиваемся того, что верхний слой как бы напоззает на предыдущий

Кафедра АСС