

A collection of various light blue geometric shapes including triangles, squares, and circles, some containing icons like a gear and a lightbulb, scattered on the left side of the slide.

CSS

JavaScript

Модуль 2. Урок 3.

# CSS

- Введение в CSS
- Позиционирование
- Атрибут z-index
- CSS Box Model
- Атрибуты элемента Display и Visibility
- Частичная видимость: атрибуты overflow и clip
- Управление стилями с помощью JavaScript
- Анимация CSS
- Включение и выключение таблиц стилей

# ВВЕДЕНИЕ В CSS

- Каскадные таблицы стилей (Cascading Style Sheets, CSS) — стандартное средство для определения визуального представления HTML документов.
- Каскадные таблицы стилей (CSS) предназначены для использования графическими дизайнерами: с помощью CSS дизайнер может точно указать шрифты, цвета, поля, отступы, границы и даже расположение элементов в документе.
- CSS также представляет интерес для программистов JavaScript, так как стили CSS могут быть определены в скрипте.
- В визуальном отображении HTML документа существует множество переменных: шрифты, цвета, интервалы и т. д.

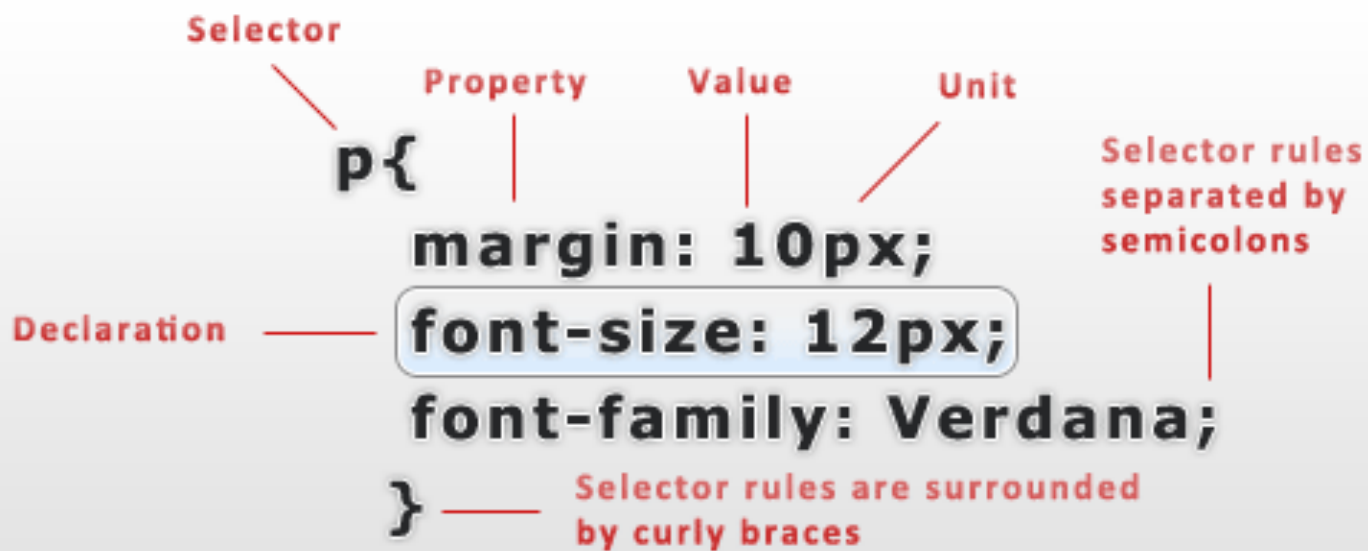
# ВВЕДЕНИЕ В CSS

- В стандарте CSS перечислены эти переменные, называемые свойствами стиля. CSS определяет свойства, которые задают шрифты, цвета, поля, границы, фоновые рисунки, выравнивание текста, размер и позицию элементов.
- Для определения визуального представления элементов HTML необходимо указать значения свойств CSS.
- Существует два способа связать значения свойств CSS с элементами HTML, представление которых они определяют:
  - определить атрибут стиля для каждого отдельного элемента HTML
  - отделить стили CSS от индивидуальных элементов HTML и определить их в таблице стилей

# ВВЕДЕНИЕ В CSS

- Буква С в аббревиатуре CSS относится к слову Cascading — каскадный. Оно указывает на то, что правила стиля, применяемые к данному элементу в документе, могут браться из «каскада» различных источников:
  - из стандартной таблицы стилей веб-браузера
  - из таблиц стилей документа
  - из атрибутов стиля для каждого отдельного элемента HTML
- Стили из атрибута стиля переопределяют стили из таблиц стилей.
- Основным элементов таблицы стилей CSS является правило стиля, которое состоит из селектора и ряда свойств CSS со значениями, заключенными в фигурные скобки.

# СТРУКТУРА



# СЕЛЕКТОРЫ

- Селекторы используются для объявления применяемой части разметки стиля посредством соотнесения тэгов и атрибутов в самой разметке.
- Селекторы могут применяться:
  - ко всем элементам конкретного типа, например к заголовкам второго уровня h2
  - к элементам, определенным атрибутом, в частности:
    - id: уникальный идентификатор документа
    - class: идентификатор, который группирует различные элементы в документе
  - к элементам в зависимости от того, как они располагаются по отношению к другим элементам в дереве документа

# СЕЛЕКТОРЫ

Паттерн	Описание	Пример
E	элемент типа E	h2
E F	элемент F — потомок элемента E	div p
*	любой элемент	*
E#myid	элемент E с ID равным "myid"	p#myP
E.info	элемент E, имеющий класс "info"	div.Container
E > F	элемент F — дочерний элемент элемента E	ul > li
E + F	элемент F, которому непосредственно предшествует элемент E	image + p
E[foo]	элемент E с атрибутом "foo"	p[align]
E[foo="bar"]	элемент E, у которого значение атрибута "foo" точно равно "bar"	p[align="left"]



## ПРИМЕР

```
<!DOCTYPE html>
<html>
<head lang="en">
  <meta charset="UTF-8">
  <title>Test Document</title>
  <style>
    body {
      margin-left: 30px;
      margin-right: 15px;
      background-color: #ffffff
    }

    p {
      font-size: 24px;
    }
  </style>
</head>
<body>
<p>Testing, testing</p>
</body>
</html>
```

# СВОЙСТВА

Свойство	Описание
position	Указывает тип позиционирования, применимого к элементу
top, left	Указывают позицию верхнего и левого краев элемента
bottom, right	Указывают позицию нижнего и правого краев элемента
width, height	Указывают размер элемента
z-index	Указывает «порядок размещения» элемента относительно любых накладывающихся элементов; определяет третье измерение позиционирования элемента
display	Указывает, отображается ли элемент и каким образом
visibility	Указывает, является ли элемент видимым
clip	Определяет «область отсечения» для элемента; отображаются только части элемента в этой области
overflow	Указывает, что делать, если элемент больше, чем выделенное ему пространство

# СВОЙСТВА

Свойство	Описание
margin, border, padding	Указывают интервалы и границы для элемента
background	Указывает фоновый цвет или рисунок для элемента
opacity	Указывает, насколько непрозрачным (или прозрачным) должен быть элемент.

# ПРИМЕР

```
<!DOCTYPE html>
<html>
<head lang="en">
  <meta charset="UTF-8">
  <title>Test Document</title>
<link rel="stylesheet" href="mystyles.css" type="text/css">
</head>
<body>
<p>Testing, testing</p>
</body>
</html>
```

## ПРИМЕР

```
p {                                /* the selector "p" matches all <p> elements */
  text-indent: .5in;              /* indent the first line by .5 inches */
}

.warning {                         /* Any element with class="warning" */
  background-color: yellow;        /* gets a yellow background */
  border: solid black 5px;         /* and a big black border */
}
```

# ПОЗИЦИОНИРОВАНИЕ И ВИДИМОСТЬ

- Свойство `position` указывает тип позиционирования, примененного к элементу. Для этого свойства возможны четыре значения:
  - `static`
  - `absolute`
  - `fixed`
  - `relative`

# ПОЗИЦИЯ

## static

По умолчанию. Это значение используется, если необходимо явно отключить позиционирование.

## absolute

Позиция элемента является фиксированной по отношению к родительскому элементу. Применимо только в случае, когда свойство позиционирования родительского элемента имеет значение `relative`, `fixed` или `absolute`. Можно использовать любой родительский элемент, указав для него значение `position: relative`, без каких-либо сдвигов.

## fixed

Позиция элемента является фиксированной. Необходимо указать позицию элемента относительно окна документа. Даже если остальная часть документа прокручивается, элемент остается в фиксированном положении.

# ПОЗИЦИЯ

## relative

Позиция элемента сдвигается относительно его обычной позиции. Используется для сдвига элемента на указанную величину. Иногда для достижения того же эффекта можно использовать свойство элемента `margin`.

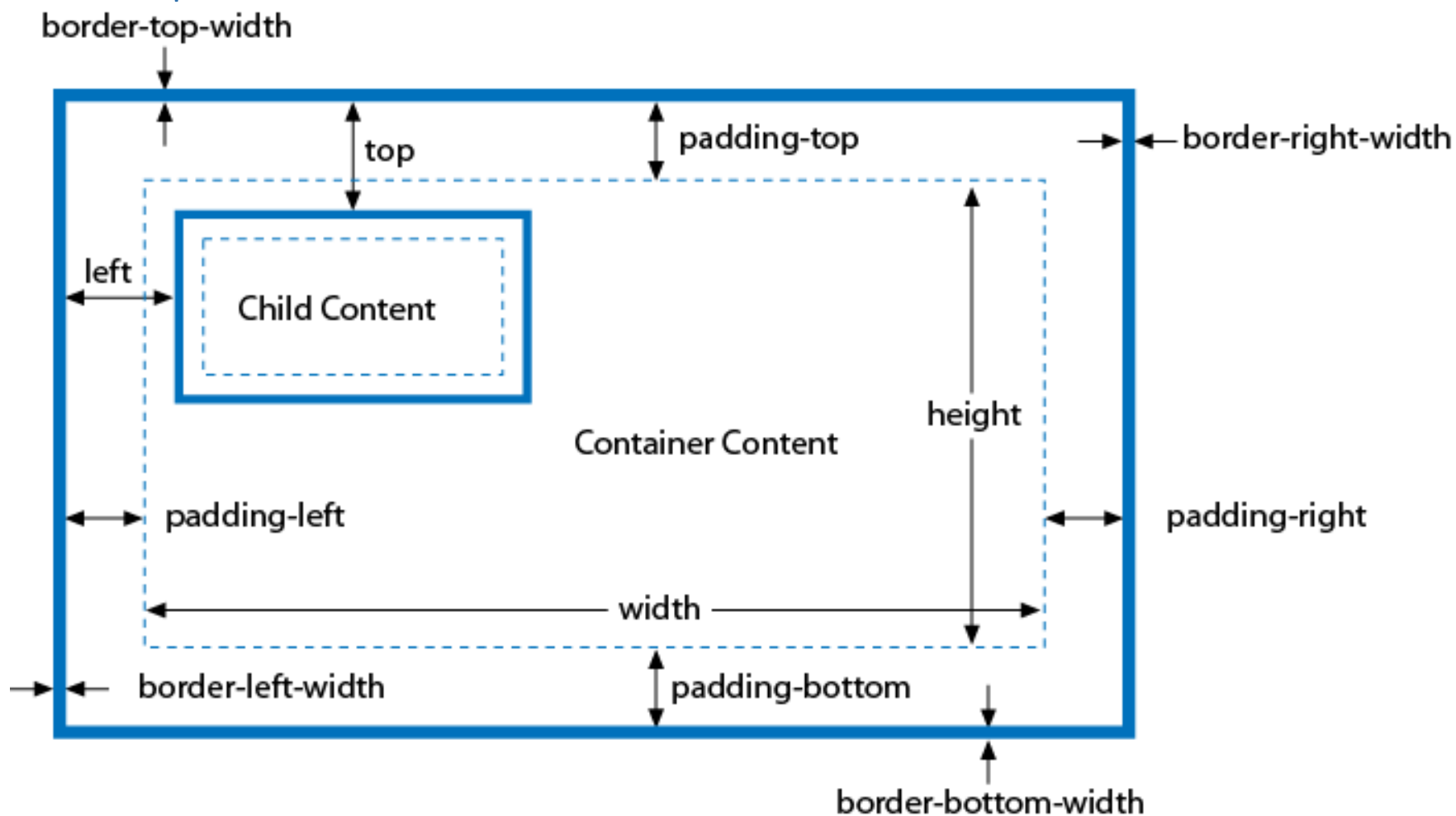


## СВОЙСТВО Z-INDEX

- Свойство z-index определяет своего рода третье измерение: с его помощью можно определить порядок размещения элементов и указать, какой из двух или более накладывающихся элементов располагается выше других.
- Свойство z-index должно быть целым числом. По умолчанию его значение ноль, но можно указать положительные или отрицательные значения.
- Когда два или более элементов перекрывают друг друга, они выводятся в порядке увеличения значения z-index; элемент с самым большим значением z-index появляется поверх всех остальных элементов.
- Если перекрывающиеся элементы имеют одинаковое значение z-index, они выводятся в том порядке, в каком они появляются в документе: последний перекрывающийся элемент появляется поверх остальных.

# CSS BOX MODEL

Абсолютно позиционированный элемент, встроенный в позиционированный элемент-контейнер



## АТТРИБУТЫ ЭЛЕМЕНТА DISPLAY И VISIBILITY

- На видимость элементов документа влияют два свойства CSS: `visibility` и `display`
- Если свойство `visibility` имеет значение `hidden`, элемент не отображается; если оно имеет значение `visible`, элемент отображается.
- Свойство `display` имеет более общий характер и используется для указания типа отображения элемента. Оно определяет, является ли элемент блоком, строковым элементом, элементом списка и т. д.
- Однако, если свойство `display` имеет значение `none`, соответствующий элемент не отображается или вообще исключается.

## АТТРИБУТЫ ЭЛЕМЕНТА DISPLAY И VISIBILITY

- Различие между свойствами стиля `visibility` и `display` заключается в их влиянии на элементы, имеющие статическое или относительное позиционирование.
- Когда значение свойства `visibility` установлено на `hidden`, элемент становится невидимым, но в разметке документа сохраняется пространство для него.
- Однако, если свойство `display` имеет значение `none`, пространство для элемента в разметке документа не сохраняется.

## ЧАСТИЧНАЯ ВИДИМОСТЬ: СВОЙСТВА OVERFLOW И CLIP

- Свойства `overflow` и `clip` позволяют отображать только часть элемента.
- Свойство `overflow` определяет, что произойдет, когда содержимое элемента превысит указанный для него размер (например, таким параметрам стиля, как ширина и высота).
- Допустимы следующие значения этого свойства:
  - `visible`
  - `hidden`
  - `scroll`
  - `auto`

## ЧАСТИЧНАЯ ВИДИМОСТЬ: СВОЙСТВА OVERFLOW И CLIP

- Значение свойства clip определяет область отсечения для документа.

`clip: rect(top right bottom left);`

- Значения top, right, bottom и left определяют границы отсекаемого прямоугольника относительно верхнего левого угла блока элемента.

## УПРАВЛЕНИЕ СТИЛЯМИ С ПОМОЩЬЮ JAVASCRIPT

- Самый простой способ прописать сценарий CSS — изменить атрибуты стиля для отдельных элементов документа.
- Как и большинство других атрибутов HTML, style является свойством объекта Element, и поэтому им можно манипулировать в JavaScript.
- При этом свойство style не совсем обычное: его значение является не строкой, а объектом CSSStyleDeclaration.
- Свойства этого объекта style в JavaScript представляют свойства CSS, определенные атрибутом HTML style.

```
var e = document.getElementById("id");  
e.style.fontSize = "24pt";  
e.style.fontWeight = "bold";  
e.style.color = "blue";
```

## УПРАВЛЕНИЕ СТИЛЯМИ С ПОМОЩЬЮ JAVASCRIPT

- Многие свойства стилей CSS, например font-size, в своих именах содержат дефисы. В JavaScript дефис трактуется как знак минуса.
- Если в имени свойства CSS есть один или более дефисов, имя свойства CSSStyleDeclaration образуется путем удаления дефисов и капитализацией буквы, идущей сразу после каждого дефиса.
- Обратите внимание, что все значения должны быть указаны как строки.
- Все свойства позиционирования указываются с единицами измерения.

```
e.style.left = 300; // Incorrect: this is a number, not a string  
e.style.left = "300"; // Incorrect: the units are missing  
e.style.left = "300px"; // Correct
```



## УПРАВЛЕНИЕ СТИЛЯМИ С ПОМОЩЬЮ JAVASCRIPT

- Альтернативой написания сценария для отдельных стилей CSS через встроенное свойство `style` может быть прописывание значения атрибута HTML класса.
- Изменение класса элемента меняет набор селекторов таблицы стилей, которые применяются к элементу и могут при этом изменять различные свойства CSS.

```
function grabAttention(e) { e.className = "attention"; }  
function releaseAttention(e) { e.className = ""; }
```

# АНИМАЦИЯ CSS

- Одним из наиболее распространенных вариантов использования CSS является создание анимационных визуальных эффектов.
- Эффекты создаются с помощью методов `setTimeout()` или `setInterval()`, которые регулярно вызывают функцию, меняющую встроенный стиль элемента.
- Модуль CSS3 Transitions определяет способ указания анимационных эффектов в таблицах стилей, исключая тем самым необходимость создания сценария.

```
.fadeable { transition: opacity .5s ease-in }
```

## ВКЛЮЧЕНИЕ И ВЫКЛЮЧЕНИЕ ТАБЛИЦ СТИЛЕЙ

- Самый простой метод создания сценария с помощью таблиц стилей также является и самым экономным и надежным.
- Элементы `<style>`, элементы `<link>` и объекты `CSSStyleSheet` определяют свойство `disabled`, которое можно запрашивать и устанавливать в JavaScript.
- Если свойство `disabled` имеет значение `true`, то таблица стилей будет отключена и проигнорирована браузером.

Если передать функции строку, она будет интерпретировать ее как селектор CSS, передаст ее методу `document.querySelectorAll()` и установит в значение `true` свойство `disabled` всех полученных элементов:

```
function disableStylesheet(ss) {  
  if (typeof ss === "number") {  
    document.styleSheets[ss].disabled = true;  
  } else {  
    var sheets = document.querySelectorAll(ss);  
    for(var i = 0; i < sheets.length; i++)  
      sheets[i].disabled = true;  
  }  
}
```

# CSS

- Введение в CSS
- Позиционирование
- Свойство z-index
- CSS Box Model
- Атрибуты элемента Display и Visibility
- Частичная видимость: свойства overflow и clip
- Управление стилями с помощью JavaScript
- Анимация CSS
- Включение и выключение таблиц стилей