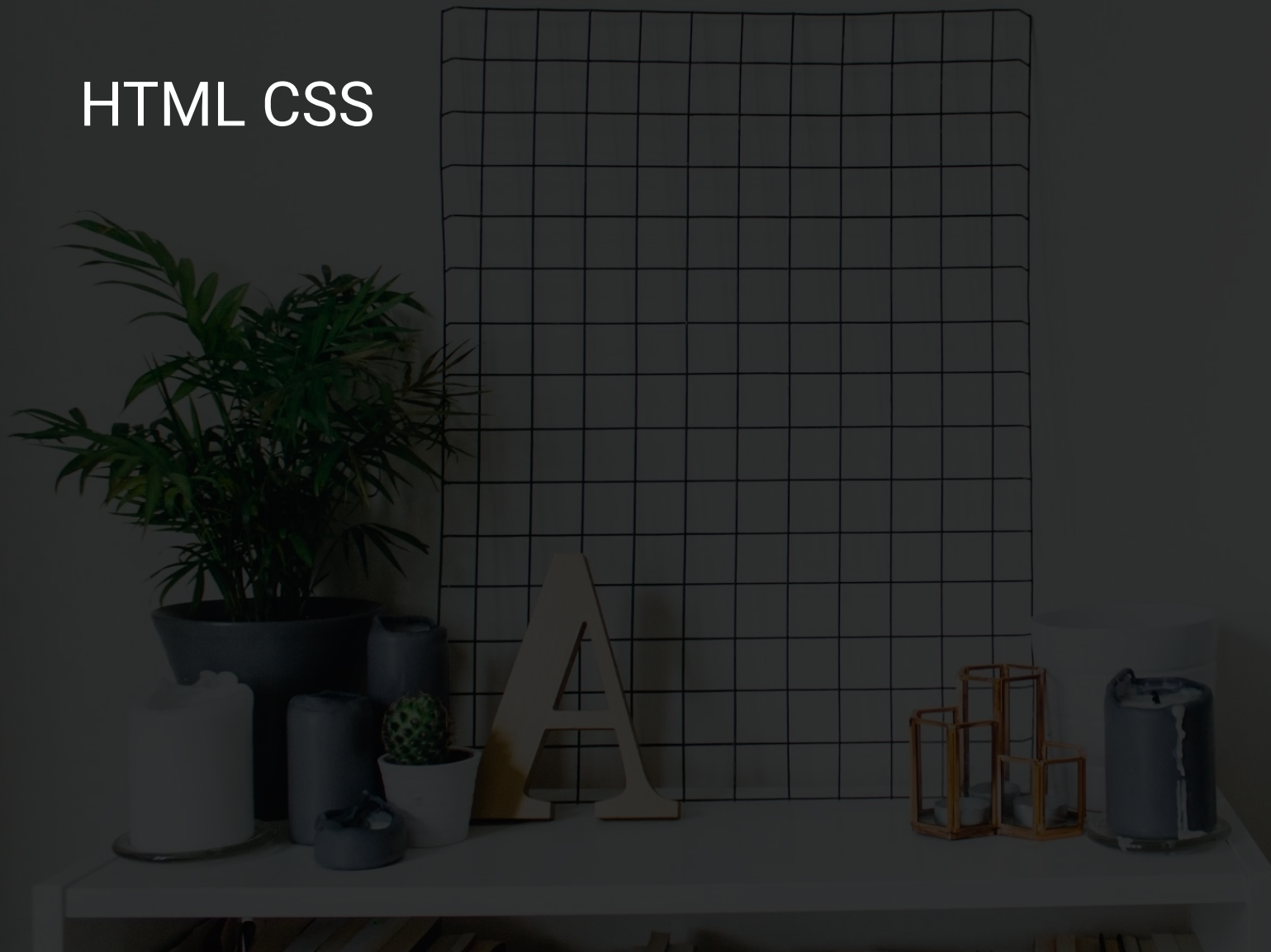




Урок 4

HTML CSS



# Основные теги для верстки (div и span)

JS article.html.js x

```
1 module.exports = (scope) => '<article>
2   <header>
3     <h1><a href="${scope.link}">${scope.title}</a></h1>
4   </header>
5   ${require('./tags.html.js')(scope)}
6   <div>
7     ${scope.body}
8   </div>
9 </article>';
10
```

JS video.html.js x

```
1 module.exports = (scope) => '<article>
2   <header>
3     <h1><a href="${scope.link}">${scope.title}</a></h1>
4   </header>
```

# Внешний вид

`<div>Это блочный элемент</div>`

`<span>Это строчный элемент</span>`

# Особенности блочных элементов

1. Блочные элементы отображаются на веб-странице в виде прямоугольника.
2. Занимают всю доступную ширину.
3. Высота определяется содержимым.
4. Начинаются с новой строки.
5. Допускается вкладывать один блочный элемент внутрь другого.
6. Запрещено добавлять внутрь строчных элементов блочные.

# Особенности строчных элементов

1. Используются для изменения вида текста и логического выделения.
2. Являются частью строки.
3. Ширина равна содержимому плюс значения отступов.
4. Внутри строчных элементов допустимо помещать текст или другие строчные элементы.
5. Свойства, связанные с размерами, неприменимы.
6. Элементы, идущие подряд, не переносятся на другую строку, располагаются на одной строке.

# План урока

1. Таблицы в html
2. Создание таблиц
3. Стилизовое оформление таблиц

# Для чего нужны таблицы?

Для представления табличных  
данных

~~Для веретки страниц~~

# Плюсы

1. Таблицы не перекрываются друг с другом при маленьких разрешениях
2. Легко сделать кроссбраузерный дизайн
3. Создание табличных элементов.



# Минусы

1. Очень много лишнего кода
2. Громоздкий код – не каждый дизайн можно создать с помощью таблиц
3. Совсем непопулярная
4. Долгая загрузка

# Структура таблицы в HTML

```
4  <table>
5      <tr>
6          <td>Столбец 1</td>
7          <td>Столбец 2</td>
8          <td>Столбец 3</td>
9      </tr>
10     <tr>
11         <td>Столбец 4</td>
12         <td>Столбец 5</td>
13         <td>Столбец 6</td>
14     </tr>
15 </table>
```

Внешний вид:

Столбец 1	Столбец 2	Столбец 3
Столбец 4	Столбец 5	Столбец 6

# Название таблицы

Внешний вид:

Таблица №1

Столбец 1	Столбец 2	Столбец 3
-----------	-----------	-----------

```
2  <table>
3  <caption>Таблица №1</caption>
4      <tr>
5          <td>Столбец 1</td>
6          <td>Столбец 2</td>
7          <td>Столбец 3</td>
8      </tr>
9  </table>
```

# Атрибуты для объединения

<b>rowspan</b> – объединение по вертикали (строк)	<b>colspan</b> – объединение по горизонтали (столбцов)	

# Размеры таблицы

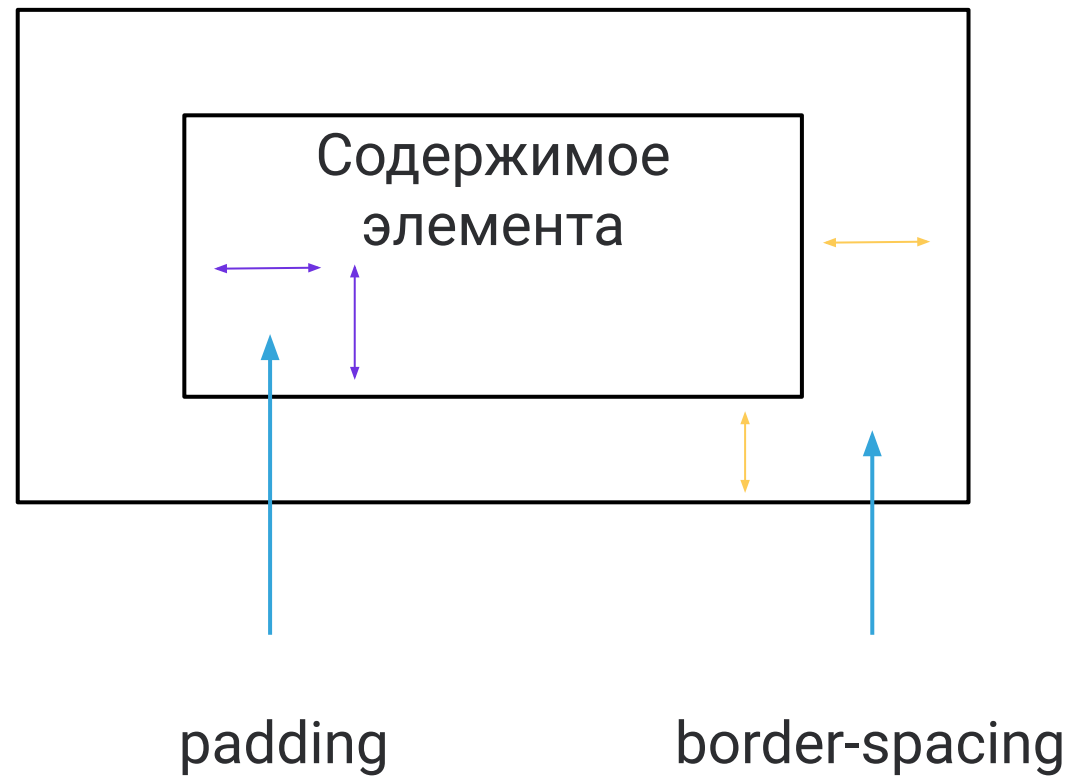
По умолчанию ширина и высота таблицы определяются содержимым ее ячеек. Если не задать ширину, она будет равна ширине самого широкого ряда.

```
2  table {  
3      width: 400px;  
4      height: 200px;  
5  }
```

# Границы в таблице

```
2
3  /* Внешняя граница таблицы */
4  table {
5      border: 1px solid black;
6  }
7
8      /* Границы для ячеек таблицы */
9      td {
10         border: 1px solid black;
11     }
12
13     /* Убрать в таблице двойные линии, которые образовались в месте стыка ячеек */
14     table {
15         border-collapse: collapse;
16     }
```

# Отступы в таблице



# Выравнивание по вертикали

```
2  
3  td {  
4    |   vertical-align: top; /* bottom | baseline | middle */  
5    |  
6    | }
```



# Псевдоэлементы

:after

добавляет контент после содержимого элемента

:before

добавляет контент до содержимого элемента

:first-letter

определяет стиль первого символа в тексте

:first-line

определяет стиль первой строки блочного текста

# Псевдоклассы

**Псевдоклассы** – это атрибуты, назначаемые строго к селекторам с намерением определить реакцию или состояние для данного селектора.



# Определяющие состояние

:hover

курсор мыши в пределах элемента

:active

при активации элемента

:focus

при получении фокуса элемента

:link

используются для непосещенных ссылок

:visited

используется для ссылки на страницу, которую уже посетили

При наведении на  
ссылки цвет изменится  
на белый, а шрифт  
будет жирным.

```
3
4  a:hover {
5      color:  □#FFF;
6      font-weight: bold;
7  }
8
```

# Выбор необходимого дочернего элемента

:first-child

первый дочерний элемент

:last-child

последний дочерний элемент

:nth-child(n)

n-й по счету дочерний элемент

# Добавление стилей к дочерним элементам

## HTML:

```
4 <div class="parent">
5   <div class="child"></div>
6 </div>
```

## CSS:

```
4 .parent:hover .child{
5   background-color: blue;
6 }
```

# Добавление стилей к дочерним элементам

При наведении на блок с классом «parent» у дочернего элемента с классом «child» цвет фона изменится на синий.

# Комбинирование псевдоклассов

При наведении на не посещенную ссылку цвет текста будет зеленым.

```
2
3  a:link:hover {
4      color: #0F0;
5  }
```



# Float

**float:left;** (выравнивание по левой стороне содержащего элемента, а весь контент выравнивается по правой).

**float:right;** (выравнивание по правой стороне содержащего элемента, а весь контент выравнивается по левой).

**float:none;** (значение по умолчанию).

# Особенности float

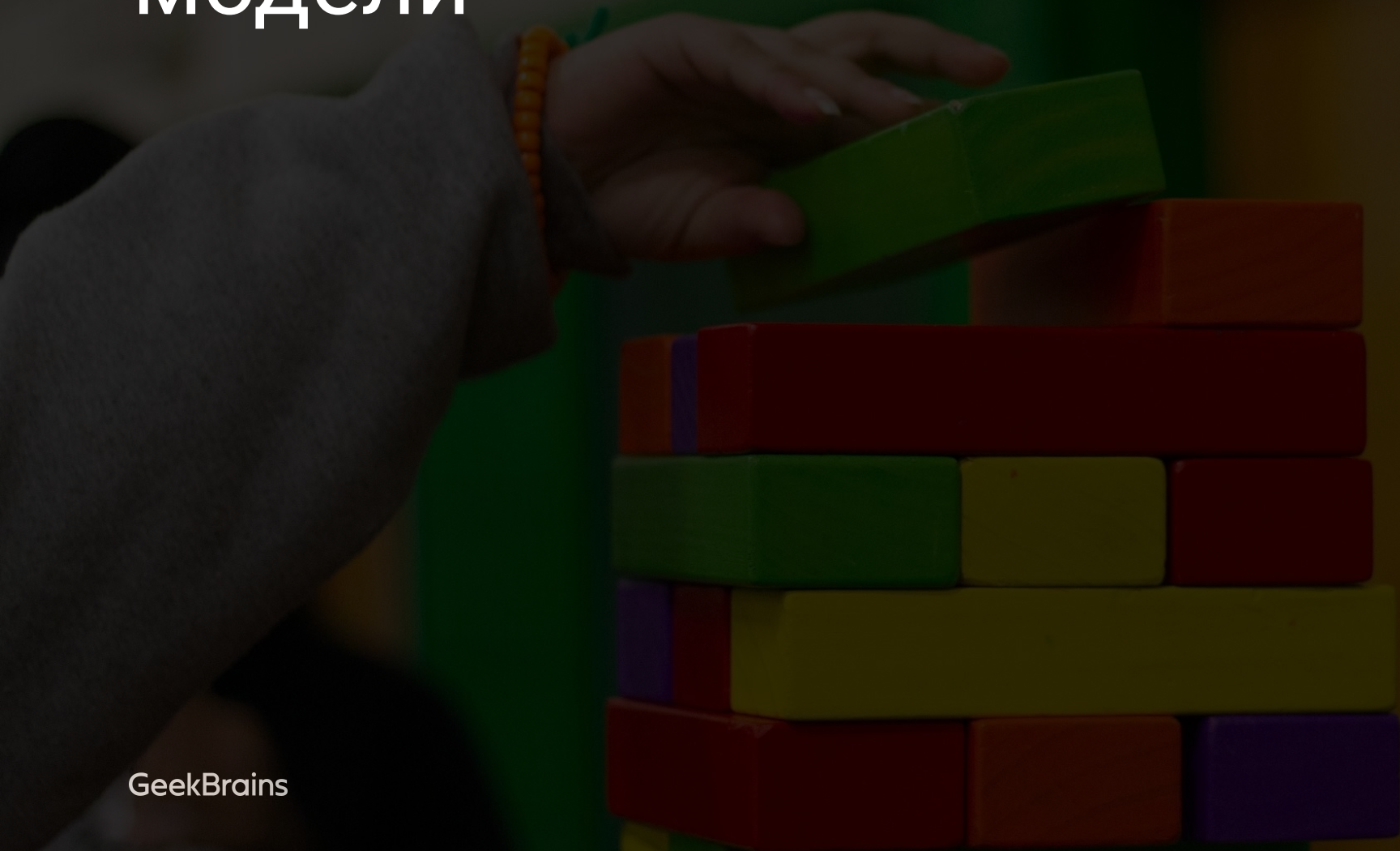
1. Элемент сдвигается влево или вправо.
2. Блочные элементы ведут себя, как будто элемента с float нет.
3. Строчные элементы обтекают блок с float.
4. Элемент при наличии float получает display:block.
5. Вертикальные отступы не сливаются.

# clear & overflow

clear:both

overflow:hidden

# Формирование блочной модели

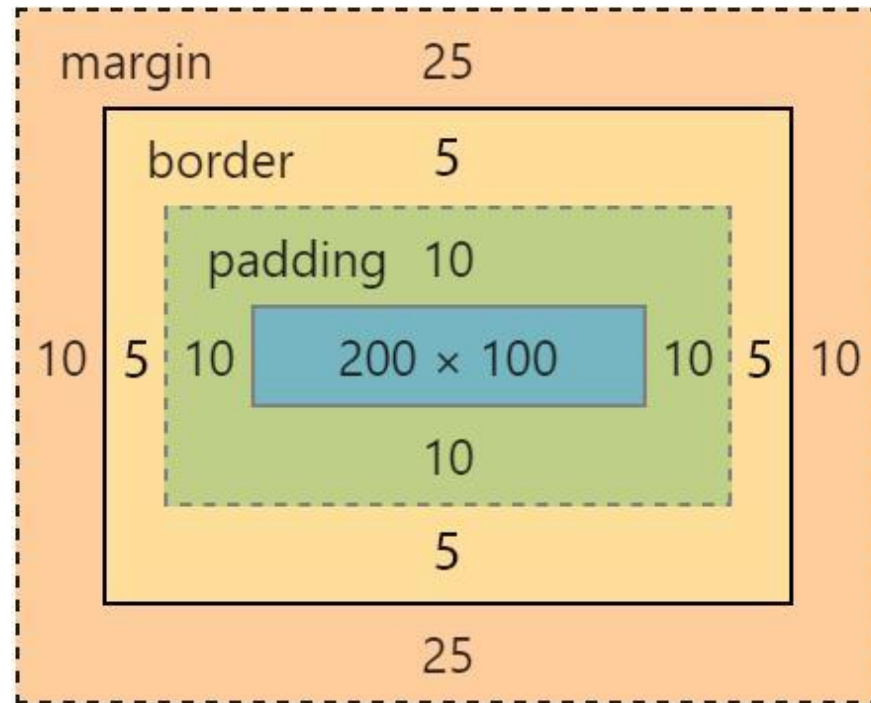


# Ширина и высота элемента

1. Содержимое (width, height).
2. Внутренний отступ (padding).
3. Рамка (border).
4. Внешний отступ (margin).

# Блочная модель

margin: 25px 10px;  
border: 5px solid black;  
padding: 10px;  
width: 200px;  
height: 100px;



# Схлопывание

- Применяется только для вертикальных отступов.
- Ширина общего отступа равна ширине большего из исходных.

