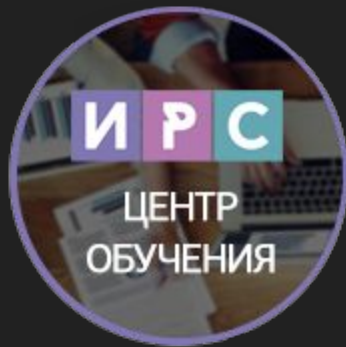


# Курс вёрстки и программирования сайтов

[goo.gl/5WcHvs](https://goo.gl/5WcHvs)



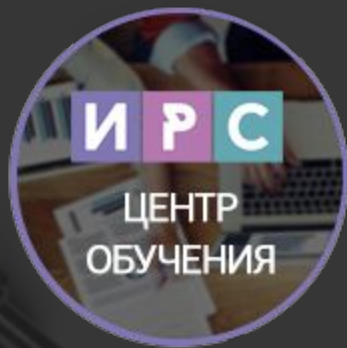
## **Ямасыпов Виталий**

вконтакте: [vk.com/snake\\_yava](https://vk.com/snake_yava)

почта: [snake-yava@mail.ru](mailto:snake-yava@mail.ru)

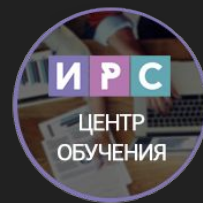
skype: [snake-yava](https://www.skype.com/people/snake-yava)

icq: 366696661



# HTML

```
<!DOCTYPE [html PUBLIC  
<html xmlns="http://www  
<head>  
  <meta name="TITLE"  
  <meta http-equiv="c
```



# Синтаксис HTML

Структура тега:

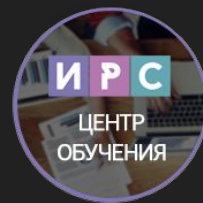
<имя тега атрибут1 атрибут2="значение2" ...>вложенные элементы</имя тега>

Пример: <FONT color="red" face="Arial">Текст</FONT>

**Неправильно:** <b><i>жирный курсив</b></i>

**Правильно:** <b><i>жирный курсив</i></b>

Непарные теги: <br>, <img>



# Структура документа HTML

```
<html>
```

```
  <head>
```

... заголовочная часть документа (*информация для браузера*)

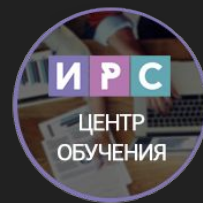
```
</head>
```

```
<body>
```

... тело документа (*видимая для пользователя часть*)

```
</body>
```

```
</html>
```



# Простейший HTML-документ

```
<html>
```

```
  <head>
```

```
    <title>Заголовок</title>
```

```
    <meta charset="UTF-8">
```

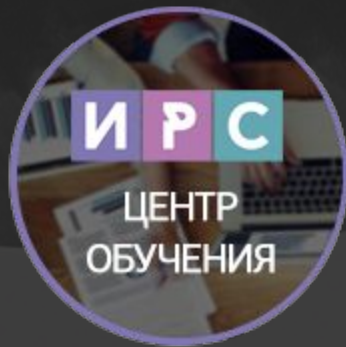
```
  </head>
```

```
  <body>
```

```
    Мой первый <b>HTML-документ!</b><br>(это пример)
```

```
  </body>
```

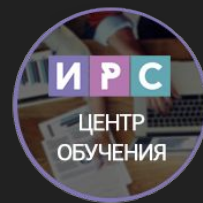
```
</html>
```



HTML5

CSS3  
**CSS**

Javascript



# ОСНОВЫ CSS

CSS (Cascading Style Sheets – каскадные таблицы стилей, произносится «си-эс-эс») – технология управления внешним видом элементов (тегов) веб-страницы. CSS предоставляет гораздо больше возможностей по оформлению страницы, чем HTML.

Например, с помощью стилей CSS можно убрать у ссылок подчеркивание, сделать у таблицы пунктирные границы или даже поменять курсор «мыши».

CSS используется практически на всех сайтах.





HTML



HTML  
+  
CSS

# Синтаксис CSS

Рассмотрим синтаксис CSS. В стилях задается набор правил отображения в парах «свойство – значение», и то, к каким элементам их применять (селектор):

```
Селектор  
{  
    свойство1: значение1;  
    свойство2: значение2;  
    свойство3: значение3 значение4;  
}  
  
p {  
    color: #ff0000;  
    font-size: 14px;  
}  
  
h1 {  
    font-size: 20px;  
}
```

Правила записываются внутри фигурных скобок и отделяются друг от друга точкой с запятой. Между свойствами и их значениями ставится двоеточие.

*CSS, как и HTML, игнорирует пробелы. Можно добавлять комментарии, заключая их между /\* и \*/*

# Селекторы

Селектор определяет, к каким элементам (тегам) страницы будут применяться правила, заданные парами «свойство – значение».

**В качестве селектора можно использовать:**

**Название тега** – *тогда стиль применится ко всем таким тегам.*

**Несколько тегов через запятую** – *тогда стиль применится для всех перечисленных тегов.*

**ID элемента** - *В стилях уникальный идентификатор указывается после знака # – правила применяются к тегу с атрибутом id="идентификатор".*

**Классы** (например, для некоторых ссылок можно сделать класс `.buttons` и стилизовать в виде кнопки)

# Классы

*Часто нужно, чтобы стиль применялся не ко всем тегам на странице, а только к некоторым элементам (например, не ко всем ссылкам на странице, а только к тем, которые расположены в меню сайта). Для этого используются классы:*

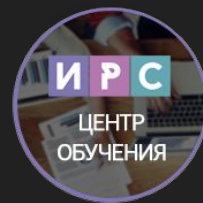
**ТЕГ.имя\_класса { ... }**

**пример: a.button { ... }**

*Правила, указанные после такого селектора, будут действовать только на теги с атрибутом class="имя\_класса":*

**<ТЕГ class="имя\_класса"> ... </ТЕГ>      <a class="button">Кнопка</a>**

*Можно не указывать имя тега, тогда правила будут применяться ко всем тегам с подходящим значением атрибута class.*



# Классы

## Пример:

Для всех тегов с атрибутом `class="class1"` добавим подчеркивание текста и уменьшим размер шрифта, а для тега `<B>` уберем подчеркивание.

```
.class1 {text-decoration: underline; font-size: 80%;}
```

```
A.class1 {text-decoration: none;}
```

## В HTML-коде укажем для тегов имя класса:

```
<h1 class="class1">Мои любимые сайты</h1>
```

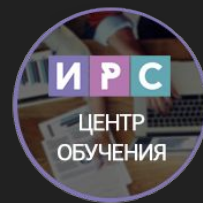
```
<a href="http://yandex.ru" class="class1">
```

```
Яндекс</a><br>
```

```
<a href="http://google.com" class="class1">
```

```
Google</a><br>
```

```
<a href="http://redut.ru" class="class1">Redut.ru</a>
```



\*

Использование CSS позволяет разделить оформление и содержимое документа.

В нашем примере правила оформления содержатся в файле `style.css`, а содержание – в `links.html`. Такое разделение существенно упрощает редактирование сайта в дальнейшем.

Рекомендуется для оформления использовать только средства CSS, отказаться от использования таких тегов, как `<font>`, `<s>`, `<u>`, `<center>`, атрибутов `align`, `border`, `color`, `height`, `width` и т.д.

# Включение стилей

**Стили CSS могут включаться в HTML-документ 3 разными способами:**

## **Внешние стили.**

Хранятся в отдельном файле .css. Подключаются тегом `<link rel="stylesheet" type="text/css" href="адрес_стиля">` в заголовке HTML-документа (`<head>...</head>`).

Основное преимущество: один стиль может использоваться сразу в нескольких документах HTML.

## **Стили уровня документа.**

Применяются ко всему документу, записываются внутри тега `<style>...</style >`, который вкладывается в тег `<head>...</head>` в документе HTML.

Такой способ указания стилей используется, когда нужно применить одинаковые стили сразу к множеству HTML-элементов (тегов) в одном документе.

## **Внутренние стили.**

Используются, когда нужно указать стили конкретного единственного элемента. Внутренний стиль записывается в атрибуте `style` и применяется только к содержимому этого тега. Внутренний стиль имеет более высокий приоритет, чем внешние стили и стиль уровня документа. Предпочтительно не использовать такой способ задания стиля, т.к. он не отвечает принципу разделения содержания и оформления.

# Порядок применения стилей

При работе с CSS необходимо помнить, что более специфичные правила имеют приоритет над менее специфичными, например:

стиль, указанный в атрибуте `style`, перекрывает стиль, указанный в теге `<style>` или внешнем файле CSS:

```
<html>
  <head>
    <style>
      A {color: red; text-decoration: none}
    </style>
  </head>
  <body>
    <a href=http://intuit.ru style="color: green">INTUIT</a>
  </body>
</html>
```

В браузере ссылка будет неподчеркнутой, зеленого цвета.



# Порядок применения стилей

селектор ID (#) имеет больший приоритет, чем селектор класса (.), а тот, в свою очередь, – больший, чем обычный селектор тега:

```
<html>
  <head>
    <style>
      A {color: red; text-decoration: none; font-size: 120%}
      .links {color: blue; text-decoration: underline}
      #greenlink {color: green}
    </style>
  </head>
  <body>
    <a href="http://htmlbook.ru" class="links" id="greenlink">htmlbook.ru</a>
  </body>
</html>
```

В браузере ссылка будет зеленой и подчеркнутой, размер шрифта увеличен на 20%.

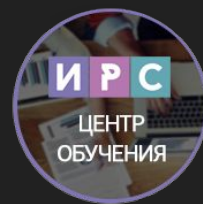
# Наследование

Другой важной особенностью CSS является то, что некоторые атрибуты наследуются от родительского элемента к дочернему.

Например, если атрибут `font-size` задан для тега `<div>`, то он наследуется всеми элементами на странице.

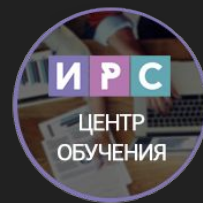
Когда свойство размера задается в процентах, оно будет вычислено исходя из значения для родительского элемента.

Узнать, является ли атрибут наследуемым, можно в справочнике по атрибутам CSS (например, <http://htmlbook.ru>).



# CSS. Задание 1

- а) Создайте внешний CSS файл `style.css` в поддиректории `css`. Подключите его ко всем страницам вашего сайта. Увеличьте размер шрифта, задайте для тега `BODY` фон свойством `background-color` и границу толщиной `5px`.
- б) На главной странице измените цвет фона на отличный от цвета на других страницах.
- в) Создайте 2 различных класса стилей для ссылок на внутренние страницы (в навигационном меню) и внешних ссылок. Добавьте атрибут `class` в теги `<a>` на страницах.



# Полезные ссылки

Google ;) [google.com](http://google.com)

ОСНОВЫ CSS [http://css.manual.ru/articles/css\\_basics](http://css.manual.ru/articles/css_basics)

ОСНОВЫ CSS <http://www.intuit.ru/department/internet/operawebst/27/>

CSS по шагам <http://stepbystep.htmlbook.ru/?pid=5>

# CSS-свойства: размеры, цвета, шрифты, текст

## РАЗМЕРЫ

Размеры в CSS можно задавать в различных единицах измерения:

em – текущая высота шрифта

pt – пункты (типографская единица измерения шрифта)

px – пиксель

% – процент

Единица измерения записывается сразу за значением без пробела:

```
TABLE {font-size: 12pt}
```

# CSS-свойства: размеры, цвета, шрифты, текст

## Типы шрифтов:

**serif** – шрифты с засечками (антиквенные), например: Times New Roman, Georgia.

**sans-serif** – рубленые шрифты (шрифты без засечек или гротески), типичные представители – Arial, Impact, Tahoma, Verdana;

**cursive** – курсивные шрифты: Comic Sans MS;

**fantasy** – декоративные шрифты, например: Curlz MT.

**monospace** – моноширинные шрифты, ширина каждого символа одинакова.

Примеры: Courier New, Lucida Console.

# Текст

CSS позволяет управлять свойствами шрифта и текста.

**font-family** – задает начертание шрифта. Можно указать несколько значений через запятую.

Браузер проверит первый шрифт из списка: если шрифт установлен на компьютере пользователя, то браузер применит его, если нет - перейдет ко второму шрифту и т.д. Последним в списке обычно указывается общий тип шрифта serif, sans-serif, cursive, fantasy или monospace.

Пример:

```
font-family: Georgia, 'Times New Roman', serif
```

Если на компьютере пользователя установлен шрифт Georgia, то будет использоваться он, если нет – то Times New Roman. Если же и Times New

Roman отсутствует, то браузер будет использовать шрифт с засечками, который установлен на компьютере.

**font-style** – задает начертание текста: normal (обычное), italic (курсивное).

**color** – задает цвет текста (см. пункт «Цвета» этой лекции). Например, зададим красный цвет для всех заголовков:

```
H1, H2, H3, H4, H5, H6 {color: #ff0000}
```

или

```
H1, H2, H3, H4, H5, H6 {color: red}
```

**line-height** – межстрочный интервал (интерлиньяж), указывает расстояние между строками текста. Может задаваться числом как множитель от текущего размера шрифта, в процентах, а также в пунктах (pt), пикселях (px) и других единицах измерения CSS.

Пример:

```
line-height: 1.5; /* полуторный интервал */
```

**text-decoration** – задает оформление текста. Варианты: `line-through` (перечеркнутый), `overline` (линия над текстом), `underline` (подчеркивание), `none` (отключение эффектов).

Например, отключим подчеркивание у ссылок:

```
A {text-decoration: none}
```

**text-align** – выравнивание текста в блоке: `left` (по левому краю), `center` (по центру), `right` (по правому краю) или `justify` (по ширине).

Пример:

```
P {text-align: justify}
```



## CSS. Задание 2

С помощью CSS задайте следующие параметры для заголовков:

размер шрифта – 16 pt, полужирный, выравнивание по центру.

Для каждого абзаца – размер шрифта 12 pt,

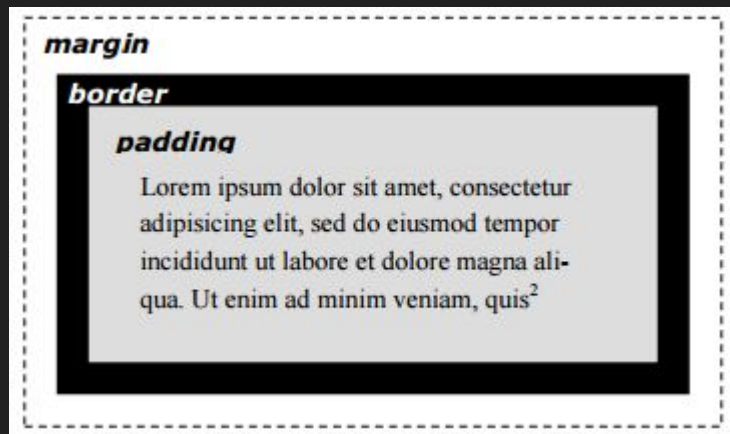
межстрочный интервал – полуторный, (line-height)

красная строка – 1,5 см. (text-indent)

Подберите подходящий шрифт для заголовка и текста.

# CSS-свойства: поля, заполнение, границы

В CSS каждый элемент располагается в блоке, которому можно задать значения полей (*margin*), заполнения (*padding*) и границы (*border*). Поле является отступом элемента от соседних, а заполнение – пустой областью между границей и содержимым.



Ширина полей и заполнения задается следующими CSS свойствами: `margin-top`, `margin-right`, `margin-bottom`, `margin-left` – для верхней, правой, нижней, левой стороны поля.

`margin` – сокращенная запись. Задает значение сразу для всех сторон.

Пример:

```
P {margin: 10px}
```

аналогично записи

```
P {  
    margin-top: 10px;  
    margin-right: 10px;  
    margin-bottom: 10px;  
    margin-left: 10px;  
}
```

# Границы

`border-width` – толщина границы;

`border-color` – цвет границы (по умолчанию – черный);

`border-style` – стиль границы. Может принимать значения `solid` (по умолчанию), `dotted`, `dashed`, `double`, `groove`, `ridge`, `inset` или `outset`.

Пример:

```
<P style="border: solid 1px green">Текст</P>
```

# Размеры блоков

Многие HTML элементы по умолчанию занимают 100% ширины элемента, в которые они вложены. Для определения размера в CSS существуют свойства `width` и `height`. Чаще всего ширину и высоту задают в пикселях (px) или в процентах (%) от ширины родительского элемента.

Пример:

```
#text1 { border: 1px solid black; }
```

```
#text2 { border: 1px solid black; width: 300px; }
```

```
#text3 { border: 1px solid black; width: 50%; }
```

# Задание

Добавить границы блоков DIV.header, DIV.middle, DIV.footer

Установите поля и отступы для тегов BODY, H1 и P. Для каждого абзаца установите разный вид и толщину границ.

Для блока.left-block установите ширину 200px.

# CSS-свойства: фон, оформление таблиц

Как и в языке HTML, в CSS фоном служит заливка цветом или изображение. Фоновое изображение может быть повторяющимся.

**background-color** – устанавливает цвет фона.

Пример:

```
TD.head {background-color: #ffff00}
```

**background-image** – устанавливает в качестве фона изображение:

```
BODY {background-image: url(images/bg.jpg)}
```

**background-position** – начальное положение фонового изображения по горизонтали (left, center, right) и вертикали (top, center, bottom). Вместо ключевых слов можно указывать расстояние в пикселях или процентах.

**background-repeat** – указывает, в каком направлении должно размножаться фоновое изображение:

repeat – по горизонтали и вертикали (по умолчанию);

repeat-x – только по горизонтали;

repeat-y – только по вертикали;

no-repeat – отключить повторение.