

# Основы HTML и CSS

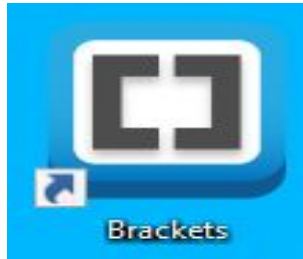
# Что нужно для работы с html/CSS/JS?

## 1. Браузер

**GoogleChrome** – поддерживает все последние фишки

## 2. Текстовые редакторы:

**Brackets**



<https://brackets.ru.uptodown.com/windows>

**CodePen** – on-line редактор

**VSCode** – от Microsoft

<https://code.visualstudio.com/download>



Установить плагины:

**Mithril Emmet** - !+ Tab

**Russian Language Pack for Visual Studio Code** - русский язык

Auto Close Tag- для комфортной работы с тегами

Auto Rename Tag – для комфортного описания HTML

Bracket Pair Colorizer 2 - продвинутая подсветка

Live Server - для отображения в браузере  
open in browser –

Prettier - Code formatter –

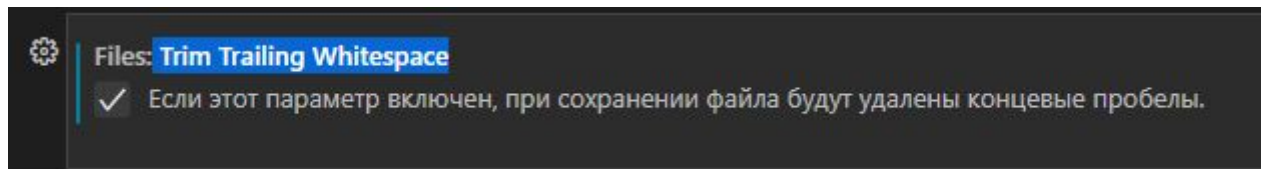
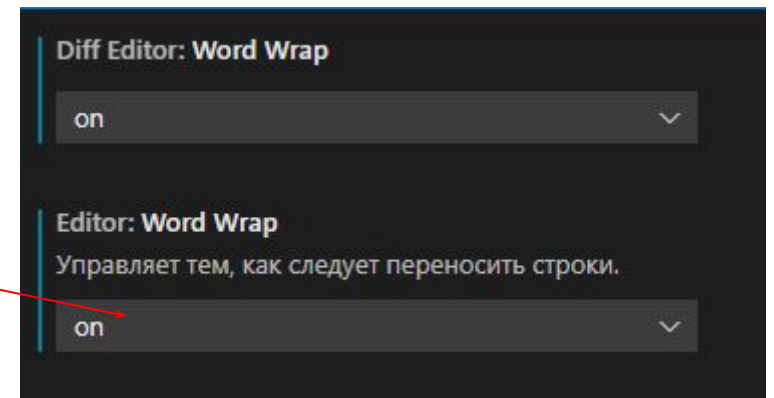
vscode-icons - ИКОНКИ

# Настройка

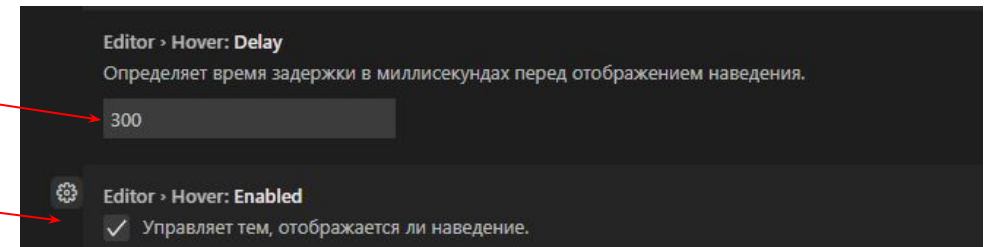
## VS Code - Параметры

1. **Word Wrap** - настройка переноса длинных строк

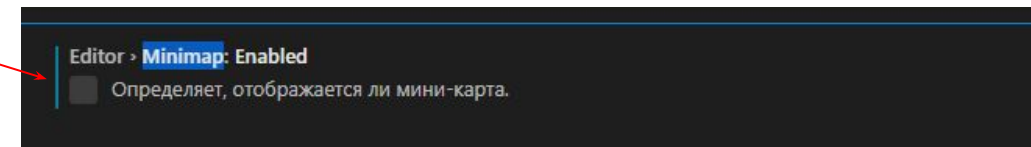
2. **Trim Trailing Whitespace**- настройка переноса длинных строк



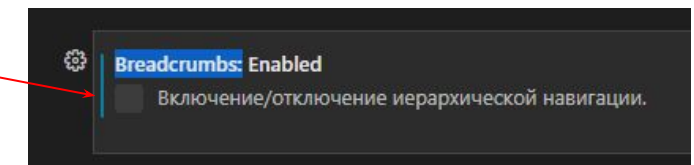
3. **Hover**- задержка подсказки, сделать 3000 или выключить (убрать галочку)



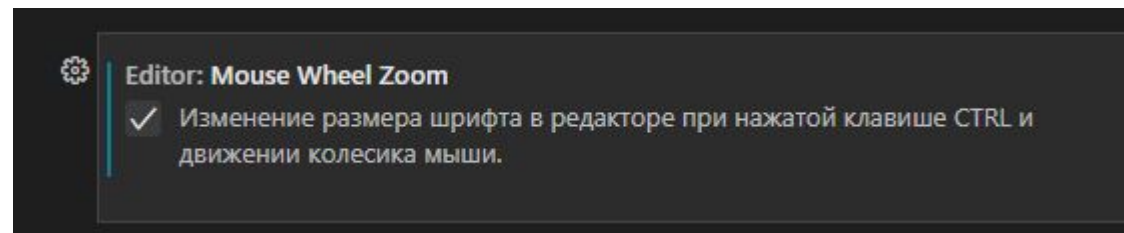
4. **Minimap** - отключить миникарту



5. **Breadcrumbs** - отключить “хлебные крошки” - навигацию



6. **Mouse Wheel Zoom** - включить масштабирование колёсиком мыши



# HTML = *Hypertext Markup Language* (язык разметки гипертекста)

**Тэг** – команда языка HTML

```
<html>
```

открывающий тэг

```
</html>
```

**контейнер**  
(парный тэг)

```
<html>
```

```
<head>
```

```
<title>Первая страница</title>
```

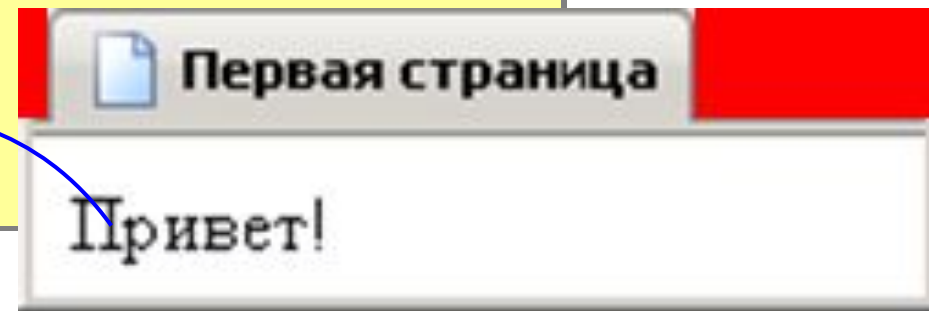
```
</head>
```

```
<body>
```

```
Привет!
```

```
</body>
```

```
</html>
```



# HTML = Hypertext Markup Language (язык разметки гипертекста)

Html-документ

<!DOCTYPE html>

Каждый HTML-документ начинается с декларации типа документа, или «доктайпа». Тип документа необходим, чтобы браузер мог определить версию HTML и правильно отобразить страницу.

<html>

<head>

<title> Пример веб-страницы

Сообщаем браузеру кодировку HTML-страницы  
<meta charset="имя кодировки">  
Самая распространённая современная кодировка — utf-8.  
**Используйте её во всех своих проектах.**

<meta charset="utf-8">

</head>

<body>

<!-- содержимое страницы -->

Комментарий для html – кода  
<!-- комментарий -->

</body>

Для кириллицы в Windows charset часто задавали как windows-1251. Но сейчас это считается плохой практикой.

<meta charset="windows-1251">

# Переход на новую строку

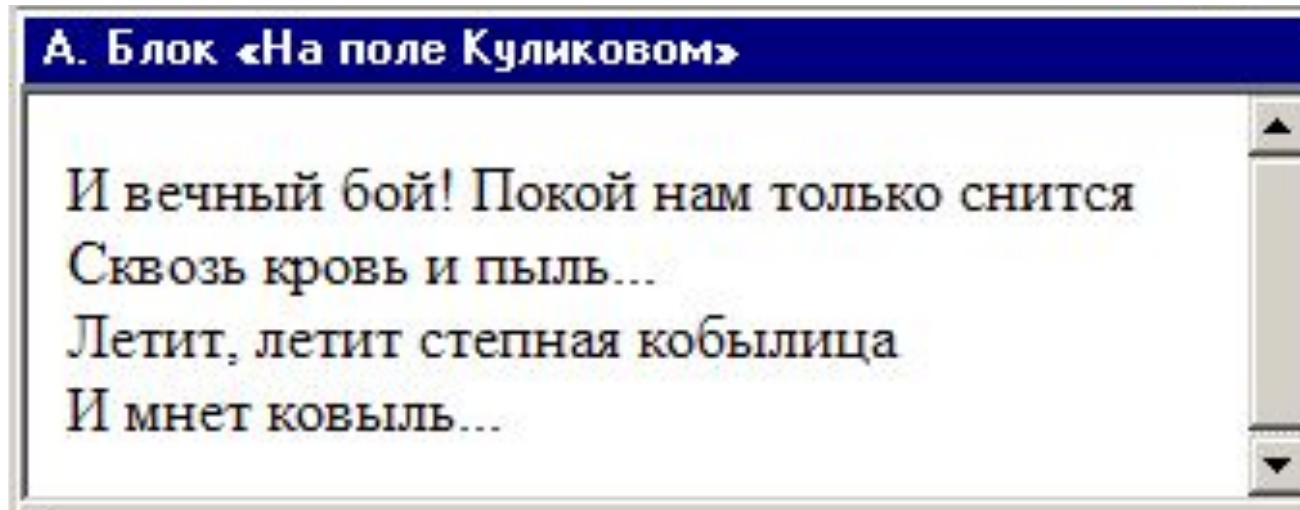
*break* –  
разрыв

И вечный бой! Покой нам только снится

**<br>**Сквозь кровь и пыль...

**<br>**Летит, летит степная кобылица

**<br>**И мнёт ковыль...



[На поле Куликовом — А. Блок](#)

# Абзац

<p>

Молекула воды испарилась из кипящего чайника и, подлетая к потолку, лоб в лоб столкнулась с неизвестно как прокравшейся на кухню молекулой водорода. Кто быстрее отлетел?

</p>

*paragraph* – абзац

<p>И вечный бой! Покой нам только снится</p>

<p>Сквозь кровь и пыль...</p>

<p>Летит, летит степная кобылица </p>

<p>И мнёт ковыль...</p>

интервал

А. Блок «На поле Куликовом»

И вечный бой! Покой нам только снится

Сквозь кровь и пыль...

Летит, летит степная кобылица

И мнет ковыль...

# Одиночный тег

**<HR>**

left, right ,center

толщина  
линии

```
<HR align = "left " size =4 width = "50%" color= " #FF0000 " >
```

ширина линии, может задаваться в пикселах, либо в процентах от ширины окна браузера)

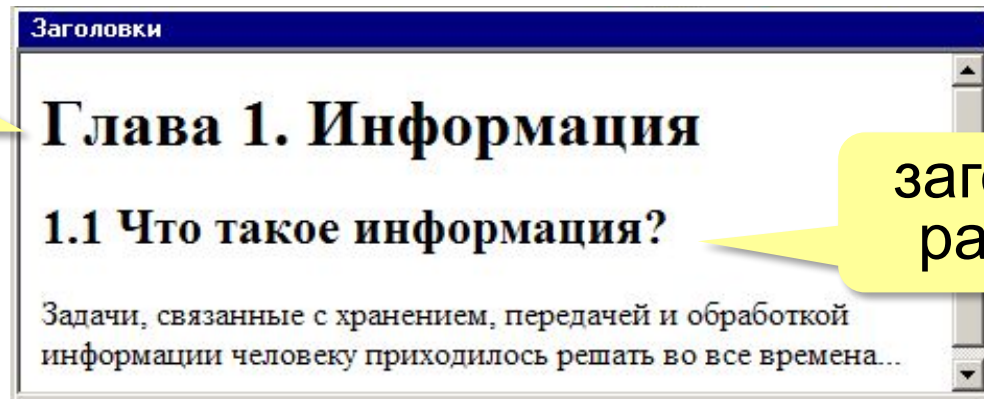


```
<body> <hr noshade size="4" width="75%">  
<hr noshade size="2" width="60%">  
<hr noshade size="1" width="50%">  
<hr noshade size="1" width="38%">  
<hr noshade size="1" width="25%">  
<hr noshade size="1" width="10%"> </body>
```



# Заголовки

заголовок  
документа



заголовок  
раздела

<h1> заголовок 1 уровня <h1>

<h2> заголовок 2 уровня <h2>

<h3> заголовок 3 уровня <h3>

<h4> заголовок 4 уровня <h4>

<h5> заголовок 5 уровня <h5>

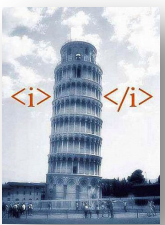
<h6> заголовок 6 уровня <h6>

```
<h1>Глава 1. Информация</h1>
```

Не используйте заголовки чтобы сделать текст **БОЛЬШИМ** или **жирным**.

Поисковые системы используют ваши заголовки для индексации структуры и содержания ваших веб страниц.

# Форматирование текста (HTML)



Курсив (*italic*):

```
<i>Вася</i>
```

Вася

Курсив (*emphasize*):

```
<em>Вася</em>
```

акцентированный

Жирный (*bold*):

```
<b>Вася</b>
```

Вася

Сильное выделение (*strong*):

```
<strong>Вася</strong>
```

Подчёркивание (*underline*):

```
<u>Вася</u>
```

Вася

Зачёркивание (*strikeout*):

```
<s>Вася</s>
```

~~Вася~~

Верхний индекс (*superscript*):

```
Вася<sup>2</sup>
```

Вася<sup>2</sup>

Нижний индекс (*subscript*):

```
Вася<sub>2</sub>
```

Вася<sub>2</sub>

Семантический смысл тегов – это значит, что каждый тег не просто задает тот или иной внешний вид интерфейса на странице, но и еще сообщает о том, зачем этот тег, поясняет смысл и назначение тега.

Всевозможные поисковые системы, сервисы используют **семантические правила** для извлечения необходимой информации, а это значит, что если страница правильно спроектирована с использованием подходящих тегов, то она будет гораздо ценней для этих поисковиков и сервисов. Это немаловажный фактор при продвижении сайта.

При использовании тегов `<em>` и `<strong>`

текст становится более важным на странице.

- особый семантический смысл
- немаловажный фактор при продвижении сайта.

```
<big> Это большой текст </big>
```

```
<small> Это маленький текст </small>
```

# Форматирование текста (HTML)

Для отображения программных кодов

`<code>a:=b+c;</code>`

`<kbd> Ввод с клавиатуры </kbd>`

`<tt> Текст телетайпа  
</tt>`

`<var> Компьютерная переменная </var>`

`<samp> Текст примера  
</samp>`

Определение (*definition*):

`<dfn>Информация</dfn> &mdash; это...`

Цитата (*citation*):

`<cite>Карету мне, карету!</cite>`

Сокращение (*abbreviation*):

`<abbr>НИИЧАВО</abbr>`

# Специальные символы (*HTML entities*)

Символ	HTML-код	Название
	<code>&amp;nbsp;</code>	неразрывный пробел
—	<code>&amp;ndash;</code>	короткое тире
—	<code>&amp;mdash;</code>	(длинное) тире
§	<code>&amp;sect;</code>	параграф
«	<code>&amp;laquo;</code>	левая русская кавычка
»	<code>&amp;raquo;</code>	правая русская кавычка
<	<code>&amp;lt;</code>	левая угловая скобка
>	<code>&amp;gt;</code>	правая угловая скобка
©	<code>&amp;copy;</code>	символ авторского права
®	<code>&amp;reg;</code>	зарегистрированная торговая марка
°	<code>&amp;deg;</code>	градус
²	<code>&amp;sup2;</code>	квадрат
³	<code>&amp;sup3;</code>	куб
×	<code>&amp;times;</code>	знак умножения
÷	<code>&amp;divide;</code>	знак деления

Неразрывный пробел (`&nbsp;`)

Дом сдали в 2011&nbsp;году.

год

Из дома вышел А.С.  
Пушкин – конце  
русской презид

не  
отрывать:  
инициалы

Пёс весил 12&nbsp;кг

единицы  
измерения

Угловые скобки (`&lt;`

`<p>Верно ли, что X < Y?</p>`

X &lt; Y

# Тег предварительного форматирования <PRE> ...

</PRE>

<pre>  
Текст  
разбит  
на строки  
</pre>

Отображение  
Текст  
разбит  
на строки

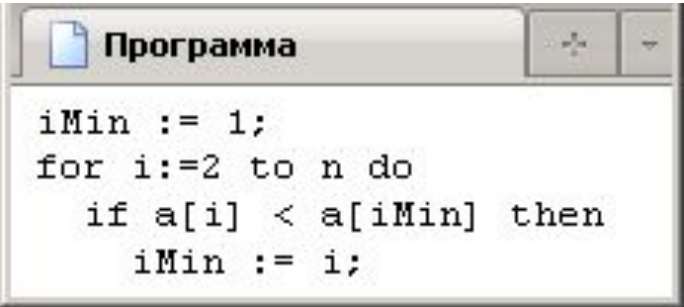
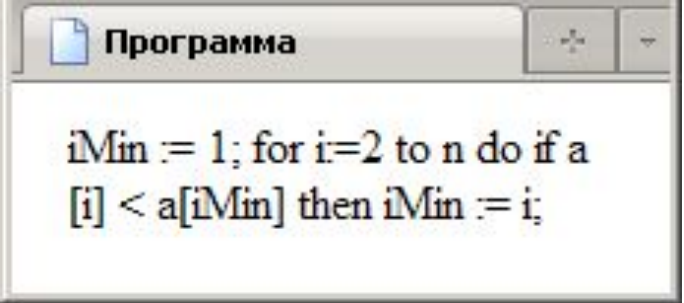
Внутри контейнера <pre> не допустимо применять теги:  
<big>, <img>, <object>, <small>, <sub> и <sup>.

Тег <PRE> ... </PRE> применяется при отображении программного кода

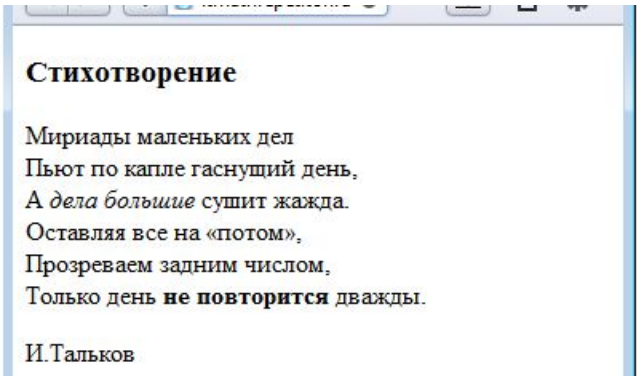
```
iMin := 1;  
for i:=2 to n do  
  if a[i] < a[iMin] then  
    iMin := i;
```

*preformatted* (уже отформатированный)

```
<pre>  
iMin := 1;  
for i:=2 to n do  
  if a[i] < a[iMin] then  
    iMin := i;  
</pre>
```



**Задание 1.** Создайте html-файл (кодировка utf-8) с заголовком "Задача 1", результат которого показан на рисунке. Используйте подходящий тег заголовков (<h1>-<h6>), теги <br>, <p>, <i>, <em>, <strong>, <b> (необязательно все перечисленные). В html-код добавьте комментарии: дату решения данной задачи и ФИО.



образе  
ц

**Стихотворение**  
Мириады маленьких дел  
Пьют по капле гаснувший день,  
А дела большие сушит жажда.  
Оставляя все на «потом»,  
Прозреваем задним числом,  
Только день **не повторится** дважды.  
И.Тальков

копироват  
ь

**Задание 2.** Создайте html-файл (кодировка utf-8) с заголовком "Задача 2", результат которого показан на рисунке. Используйте подходящий тег заголовков (h1-h6), теги <br>, <p>, <i>, <em>, <strong>, <b>, <hr>, <small> (необязательно все перечисленные). Обратите внимание, что шрифт фамилии автора меньшего размера. Заметим, что тег <hr> в разных браузерах может отображаться по-разному. В html-код добавьте условие данной задачи в виде комментариев.

образе

ц  
Не позволяй душе лениться

Не позволяй душе лениться!  
Чтоб в ступе воду не толочь,  
Душа обязана трудиться  
И день и ночь, и день и ночь!

Гони ее от дома к дому,  
Тащи с этапа на этап,  
По пустырю, по бурелому  
Через сугроб, через ухаб!

Не разрешай ей спать в постели  
При свете утренней звезды,  
Держи лентяйку в черном теле  
И не снимай с нее узды!

В.Заболоцкий

копироват

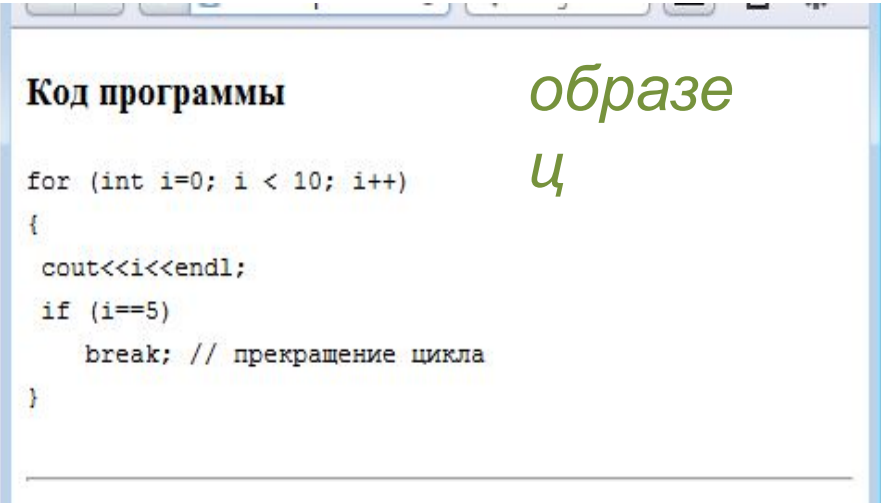
ь  
Не позволяй душе лениться

Не позволяй душе лениться!  
Чтоб в ступе воду не толочь,  
Душа обязана трудиться  
И день и ночь, и день и ночь!

Гони ее от дома к дому,  
Тащи с этапа на этап,  
По пустырю, по бурелому  
Через сугроб, через ухаб!

Не разрешай ей спать в постели  
При свете утренней звезды,  
Держи лентяйку в черном теле  
И не снимай с нее узды!  
В.Заболоцкий

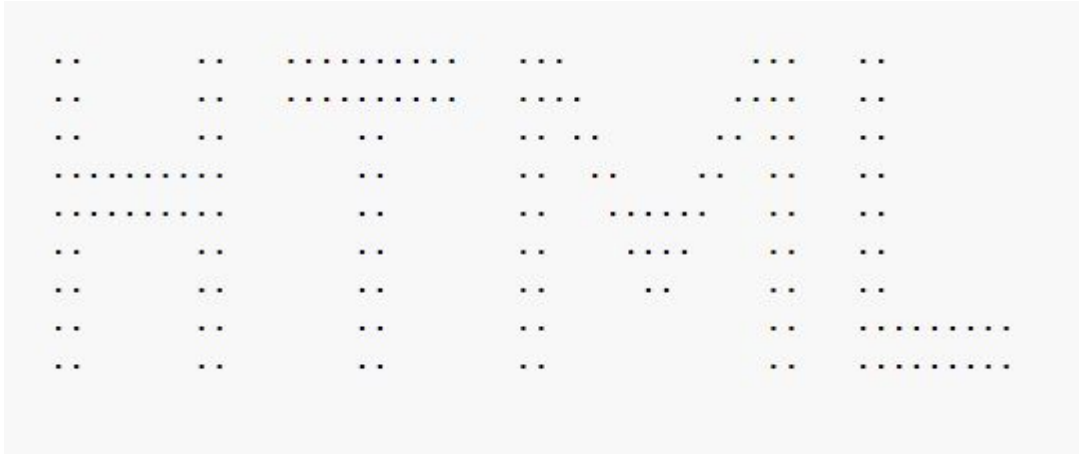
**Задание 3.** Создайте html-файл, результат которого показан на рисунке. Используйте тег <code> и &nbsp;



Код программы  
for (int i=0; i<10; i++)  
{  
  cout<<i<<endl;  
  if (i==5)  
    break; // прекращение цикла  
}

копировать

**Задание 4.** Воссоздайте код представленной html-страницы. Используйте тег <pre>, пробелы и обычные точки или звездочки.







# Форматы рисунков

## **GIF** (*Graphic Interchange Format*)

- сжатие без потерь (LZW)
- прозрачные области
- анимация
- **только с палитрой** (2...256 цветов)

рисунки с четкими границами, мелкие рисунки



**SVG** (*Scalable Vector Graphics*,  
масштабируемые  
векторные изображения)



## **JPEG** (*Joint Photographer Expert Group*)

- сжатие с потерями
- только *True Color* (16,7 млн. цветов)
- **нет анимации и прозрачности**

рисунки с размытыми границами, фото



## **PNG** (*Portable Network Graphic*)

- сжатие без потерь
- с палитрой (PNG-8) и *True Color* (PNG-24)
- прозрачность и полупрозрачность (альфа-канал)
- нет анимации
- **плохо сжимает мелкие рисунки**







# Форматы рисунков

**WebP** — новый формат, созданный и развиваемый с 2010 года компанией Google.

Главная цель этого проекта — **ещё больше уменьшить вес при сохранении такого же качества.**

Формат использует **новый алгоритм сжатия**, в котором искажения отличаются от искажений других форматов. **Ухудшается детализация и структура, в то время как края остаются чёткими.**

## Особенности WebP:

- сжимает изображения без потерь лучше, чем PNG (на 26% по данным Google);
- сжимает изображения с потерями лучше, чем JPEG (на 25–34% по данным Google);
- поддерживает прозрачность (альфа-канал).

Иногда WebP сжимает изображение даже лучше, чем заявляет Google.



JPEG:

44 килобайт



WebP:

26 килобайт.

## **Avif** – уникальный формат

**изображения** лучше на 20%, чем формата WebP;

- На 50% выше, чем формат JPG.

Правда, в некоторых случаях JPG позволяет сжимать с большим качеством изображения.

# Форматы в зависимости от цели



**Под фотографическими изображениями** понимаются полноцветные фотографии, чёрно-белые фотографии, полноцветные изображения, рисунки с большим количеством разноцветных деталей.

Под **графикой, логотипами, иконками** — графика со множеством сплошных цветов, с небольшим количеством цветов (до 256 цветов), с текстом или линиями, с прозрачностью.

	Фотографические изображения	Графика, логотипы, иконки
Лучший выбор	WebP; JPEG (с оптимальной степенью сжатия).	SVG; PNG; WebP.
Худший выбор	GIF; SVG.	JPEG (сжатие добавляет артефакты, смазывается текст, края линий, пропадает прозрачность).
Лучшее качество	JPEG (минимальное сжатие); PNG; WebP.	PNG; SVG; WebP.
Наименьший размер файла	JPEG (максимальное сжатие); WebP.	SVG; GIF.

# Изображения в документе

из той же папки:

*image*  
(изображение)

*source*  
(источник)

```

```

```

```

из другой папки:

```

```

с другого сервера:

```

```

# Атрибуты для картинок

**alt** – альтернативный текст, который отображается на месте изображения, если по каким-либо причинам само изображение не может быть показано;

```

```

# Как загрузить картинку от `show.html`?



# Гиперссылки (покальные)

страница в той же папке:

Переход на

```
<a href="newpage.html">новую страницу</a>
```

anchor  
(якорь)

hyper reference  
(гиперссылка)



во вложенной папке:

```
<a href="news/info.htm">Информация </a>
```

в родительской папке:

```
<a href=" ../info.htm">Информация </a>
```

в соседней папке:

```
<a href=" ../news/info.htm">Информация </a>
```

для скачивания:

```
<a href="file.pdf" download>Скачать </a>
```

# Гиперссылки (внешние)

на URL:

```
<a href="http://example.net/news/info.htm"> Информация  
</a>
```

Загружает страницу в  
новое окно браузера

```
<a href="new.html" target="_blank"> Открыть в новом  
окне</a>
```

для запуска почтовой программы:

```
<a href="mailto:vasya@mail.ru">  
Напишите мне!</a>
```