

Министерство сельского хозяйства Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
Иркутский государственный аграрный университет им. А.А. Ежевского  
Институт экономики, управления и прикладной информатики  
Кафедра информатики и математического моделирования

Асалханов П.Г., Бендик Н.В.

## *Создание Web-страниц с использованием HTML*

Методические указания для выполнения лабораторных работ  
по дисциплине «Информационно-коммуникационные технологии»  
для студентов направления 06.03.01 Биология



**Иркутск – 2020**

Печатается по решению методической комиссии института экономики, управления и прикладной информатики Иркутского государственного аграрного университета им. А.А. Ежевского.

Протокол №3 от 26 ноября 2020 г.

**Рецензенты:** к.т.н., доцент, директор института экономики, управления и прикладной информатики Федурин Н.И.; доцент кафедры информатики и математического моделирования Беляков А.Ю.

Бендик Н.В. Учебно-методические указания к выполнению лабораторных работ по курсу «Информационно-коммуникационные технологии» студентов направления подготовки 35.03.01 Лесное дело [Текст] / Н.В. Бендик, П.Г. Асалханов – Иркутск: Изд-во Иркутского ГАУ, 2020. – 17 с.

Данные методические указания разработано для поддержки компьютерных лабораторных занятий и самостоятельной работы по курсу «Информационно-коммуникационные технологии» для студентов и магистрантов, обучающихся по направлению «Лесное дело».

## Введение в язык HTML

Язык разметки гипертекста HTML (Hypertext Markup Language) используют для создания Web-документов. Гипертекст, то есть расширенный текст, включает дополнительные элементы: иллюстрации, ссылки, вставные объекты. Под разметкой понимается использование специальных кодов, легко отделяемых от смыслового содержания документа и используемых для реализации гипертекста. Применение этих кодов подчиняется строгим правилам, определяемым спецификацией языка HTML.

HTML-документ - это файл, содержащий обыкновенный текст со специальными командами. Такой файл имеет расширение **.htm** или **.html** и может быть подготовлен в произвольном текстовом редакторе (существуют, однако, специальные программы-конверторы и HTML-редакторы). Создание Web-страниц непосредственно в HTML-кодах надежнее всего в простых текстовых редакторах, например, в приложении Блокнот. В качестве альтернативы можно использовать более функциональный редактор Notepad++, который поддерживает подсветку синтаксиса большого количества языков программирования и разметки, в том числе и HTML.

HTML-документ состоит из содержимого, то есть собственно полезной информации, и команд, задающих структуру. Каждая команда (управляющая конструкция) HTML-документа (тег) должна заключаться в угловые скобки - вот так: <тег>.

Выделяют 2 группы тегов: одиночные, не имеющие закрывающего тега и парные теги (теги-контейнеры), требующие закрывающего тега.

Чаще всего в документе встречаются парные теги (открывающий и соответствующий ему закрывающий), так как браузеру необходимо знать область действия тега. Одиночные теги используются только там, где область действия очевидна и дополнительной информации не требуется (ясно, например, что если мы встретили тег "начало абзаца" (<P>), то предыдущий абзац уже закончился). В сомнительном же случае лучше перестраховаться и поставить закрывающий парный тег, иначе документ может оказаться нечитаемым. Открывающий и закрывающий теги называются одинаково и отличаются друг от друга только символом "наклонная черта" или "слэш" - "/", который ставится сразу после открывающей угловой скобки закрывающего тега. Закрытие парных тегов выполняется так, чтобы соблюдались правила вложения.

<b><i>На этот текст воздействуют два тега</i></b>

Кроме того, тег может включать атрибут, дающий дополнительную информацию браузеру. Например, при помощи атрибута можно попросить браузер изменить величину шрифта, ориентацию изображения по отношению к строке следующего за ним текста, поменять цвет фона документа и т.д. В парных тегах атрибуты добавляются только к открывающему тегу. Атрибуты представляют собой дополнительные ключевые слова, отделяемые от ключевого слова, определяющего тег, и от других атрибутов пробелами и размещаемые до завершающего тег символа ">". Способ применения некоторых атрибутов требует указания значения атрибута. Значение атрибута отделяется от ключевого слова атрибута символом "=" (знак равенства) и заключается в кавычки.

```
<h1 align="left">
```

Язык HTML в большинстве случаев совершенно равнодушен к регистру, в котором набираются теги. Скажем, браузеру совершенно все равно, наберете вы тег, служащий для рисования горизонтальной линии, как **<HR>** или **<hr>** - эффект будет один и тот же.

HTML не признает никакого дополнительного форматирования текста, кроме как с помощью тегов. В результате текст, превосходно смотрящийся в текстовом редакторе, в окне браузера сольется в единую нечитаемую массу. Так, на месте нескольких пробелов будет лишь один пробел. Исчезнут все заголовки, пустые строки, деление текста на абзацы. Без HTML-тегов браузер просто игнорирует все элементы форматирования.

Определение HTML как языка разметки основывается на том, что при удалении из документа всех тегов получается текстовый документ, совершенно эквивалентный по содержанию исходному гипертекстовому документу. Таким образом, при отображении документа HTML сами теги не отображаются, но влияют на способ отображения остальной части документа.

## Лабораторная работа №1.

### Создание простейшего HTML-документа. Форматирование текста

Рассмотрим последовательность создания и редактирования простейшего HTML-документа.

#### Задание 1

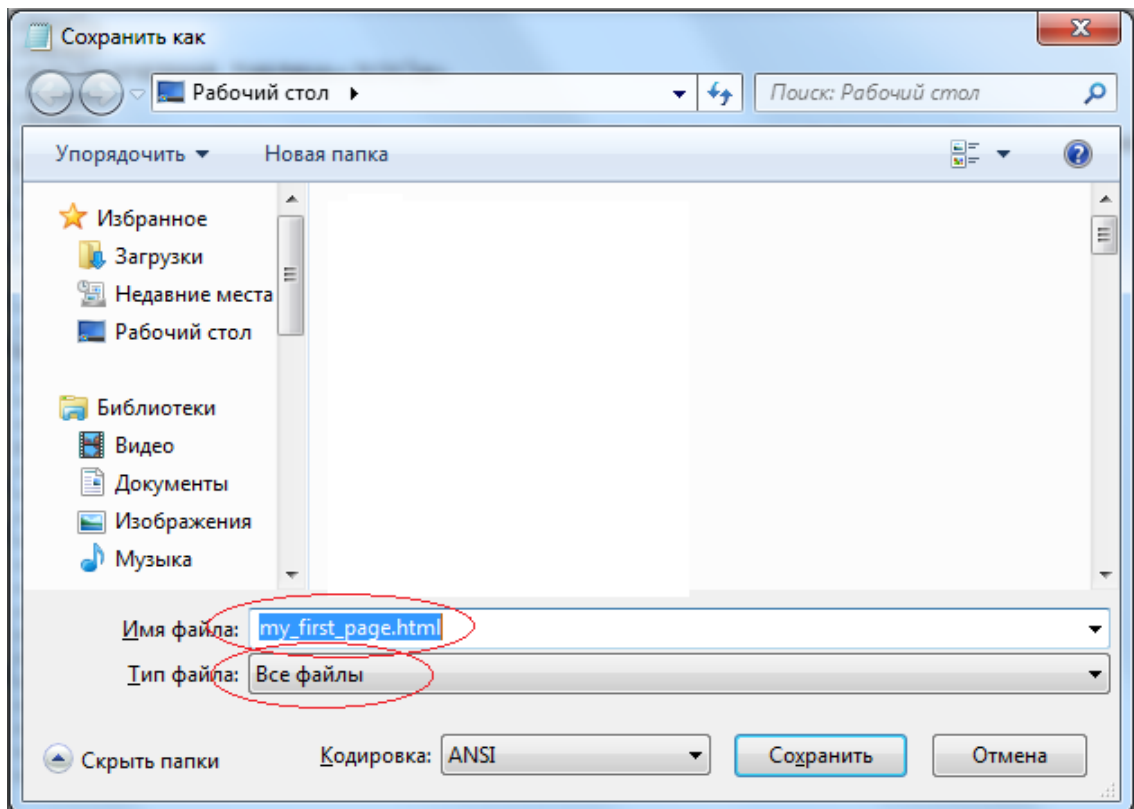
1. Откройте текстовый редактор Блокнот (меню Пуск→Все программы→Стандартные).
2. В окне текстового редактора введите следующий текст:

```
<html>
<head>
<title>Моя первая страница</title>
</head>
<body>
<p>Привет!
</body>
</html>
```

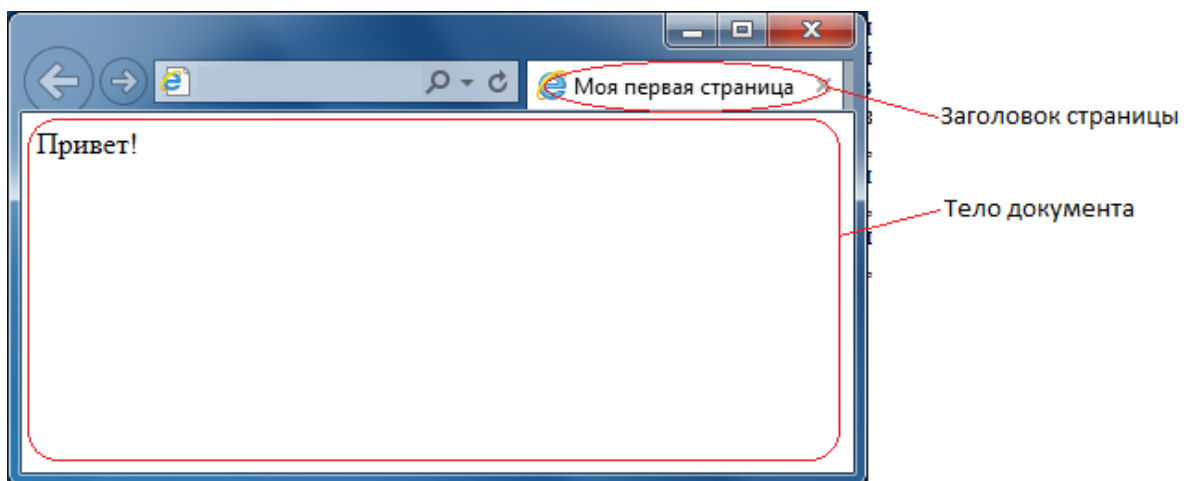
Это пример простейшей структуры html-документа.

Первый тег в документе - **<HTML>**, сообщает браузеру о том, что он имеет дело именно с документом в формате HTML. Тег **<HTML>** и парный ему закрывающий тег **</HTML>** можно считать, как бы "конвертом", в который помещается весь документ. Любой HTML-документ состоит из заголовка, который задается при помощи тега **<TITLE>**, и тела документа, который определяется тегом **<BODY>**. В заголовке документа размещается служебная информация, комментарии автора и заголовок страницы, заключаемый в теги **<TITLE>**. Заголовок, вписанный между тегами **<TITLE>**, в основное текстовое поле браузеру не попадает, а, как правило, размещается в заголовке окна браузера. Тег **<P>** обозначает абзац в тексте.

3. Сохраните документ, используя пункт "Сохранить как..." в меню "Файл". Укажите тип файла – "Все файлы" и дайте ему название, например, my\_first\_page.html. Обратите внимание, что имя файла обязательно должно заканчиваться расширением .html или .htm.



4. Откройте полученный HTML-документ в любом интернет-браузере, например, Internet Explorer. Документ при этом будет иметь приблизительно следующий вид:



5. Не закрывая браузер, отредактируйте документ в Блокноте, внося в нем следующие изменения:

- поменяйте заголовок страницы:

```
<title>Страница приветствия</title>
```

- после строки

```
<p>Привет!
```

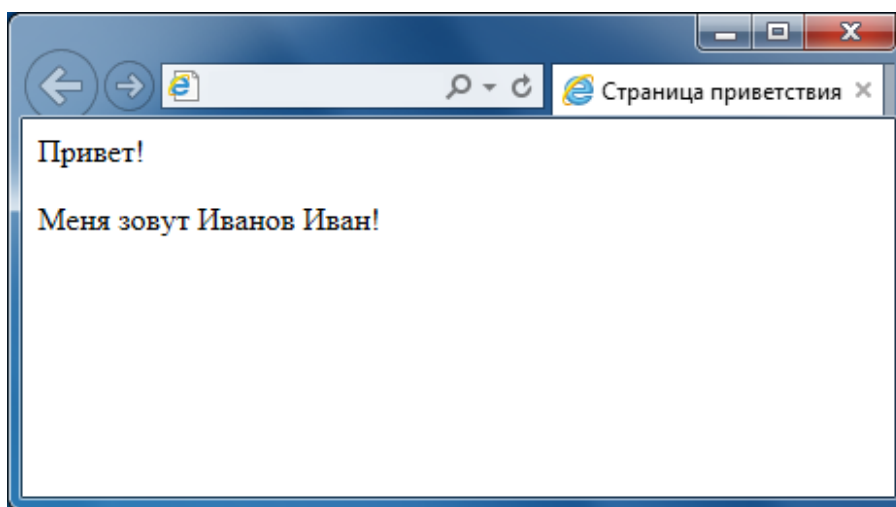
добавьте строку:

```
<p>Меня зовут Иванов Иван!
```

при заполнении используйте свои имя и фамилию.

6. Сохраните документ (Файл→Сохранить).

7. Для отображения в браузере изменений документа, нажмите кнопку F5 на клавиатуре.



При необходимости можно просмотреть HTML-код в самом браузере.

Большинство элементов языка HTML описывает части содержания документа и помещается между тегами **<BODY>** и **</BODY>**, то есть, внутри структурного элемента BODY. Такие элементы делят на блочные и текстовые. Блочные элементы относятся к частям текста уровня абзаца. Текстовые элементы описывают свойства отдельных фраз и еще более мелких частей текста. Теперь можно сформулировать правила вложения элементов:

- элементы не должны пересекаться (т.е. если открывающий тег располагается внутри элемента, то и соответствующий закрывающий тег должен располагаться внутри этого же элемента);
- блочные элементы могут содержать вложенные блочные и текстовые элементы;
- текстовые элементы могут содержать вложенные текстовые элементы;
- текстовые элементы не могут содержать вложенные блочные элементы.

Строго говоря, все правила языка HTML можно рассматривать исключительно как "пожелания". Средство, используемое для отображения

Web-документа, сделает все возможное, чтобы истолковать разметку наиболее разумным образом. Тем не менее, гарантию правильного воспроизведения документа дает только неукоснительное следование требованиям спецификации языка.

### Функциональные блочные элементы

В большинстве документов основными функциональными элементами являются заголовки и абзацы. Язык HTML поддерживает шесть уровней заголовков. Они задаются при помощи парных тегов от **<H1>** до **<H6>**. При отображении Web-документа на экране компьютера эти элементы показываются при помощи шрифтов разного размера.

Обычные абзацы задаются с помощью парного тега **<P>**. Язык HTML не содержит средств для создания абзацного отступа ("красной строки"), поэтому при отображении на экране компьютера абзацы разделяются пустой строкой. Закрывающий тег **</P>** рассматривается как необязательный. Подразумевается, что он стоит перед тегом, который задает начало очередного абзаца документа. Например:

```
<h1>Заголовок</h1>  
<p>Первый абзац  
<p>Второй абзац  
<h2>Заголовок второго уровня</h2>
```

## Заголовок

Первый абзац

Второй абзац

## Заголовок второго уровня

Следствием наличия специального тега, определяющего абзац, является тот факт, что обычного символа конца строки, вводимого по нажатию клавиши ENTER, для создания абзацного отступа недостаточно. Язык HTML рассматривает символы конца строки и пробелы особым образом. Любая последовательность, состоящая только из пробелов и символов конца строки, при отображении документа рассматривается как одиночный пробел. Это, в частности, означает, что символ конца строки даже не осуществляет перехода на новую строку (для этой цели используется текстовый элемент, задаваемый непарным тегом **<BR>**).

### Текстовое оформление страниц

Для изменения параметров шрифта можно использовать тег **<font>**. Для тега используются следующие атрибуты: **face**, **size** и **color**.



Атрибут **Face** служит для задания гарнитуры шрифтов, использующихся для текста. Названий шрифтов можно указать несколько, через запятую. В этом случае, если первый указанный шрифт не будет найден, будет использоваться следующий по списку.

**Пример 1. Использование атрибута face**

```
<font face="Arial, Helvetica, sans-serif">Текст будет написан  
шрифтом  
Arial.</font>
```

**Size** задает размер шрифта в условных единицах от 1 до 7. Средний размер, используемый по умолчанию принят 3. Размер шрифта можно указывать как абсолютной величиной (например, size=4), так и относительной (например, size=+1, size=-1). В последнем случае размер изменяется относительно базового.

**Пример 2. Задание размера шрифта**

```
<font size=1>Шрифт размера 1</font><br>  
<font size=2>Шрифт размера 2</font><br>  
<font size=3>Шрифт размера 3</font><br>  
<font size=4>Шрифт размера 4</font><br>  
<font size=5>Шрифт размера 5</font><br>  
<font size=6>Шрифт размера 6</font><br>  
<font size=7>Шрифт размера 7</font><br>
```

Шрифт размера 1  
Шрифт размера 2  
Шрифт размера 3  
Шрифт размера 4  
Шрифт размера 5  
Шрифт размера 6  
Шрифт размера 7

**Color** определяет цвет текста, который можно задавать с помощью названий цветов или цифровыми значениями в шестнадцатеричном формате. Существует 16 основных цветов, представленных в таблице.

Цвет	Символическое название	Шестнадцатеричное значение
------	------------------------	----------------------------

	Морской волны	Aqua	#00FFFF
	Черный	Black	#000000
	Голубой	Blue	#0000FF
	Малиновый	Fuchsia	#FF00FF
	Серый	Gray	#808080
	Зеленый	Green	#008000
	Ярко-зеленый	Lime	#00FF00
	Темно-красный	Maroon	#800000
	Темно-синий	Navy	#000080
	Оливковый	Olive	#808000
	Пурпурный	Purple	#800080
	Красный	Red	#FF0000
	Серебряный	Silver	#C0C0C0
	Темной морской волны	Teal	#008080
	Белый	White	#FFFFFF
	Желтый	Yellow	#FFFF00

Кроме того, на сайте [http://www.puzzleweb.ru/html/colors\\_html.php](http://www.puzzleweb.ru/html/colors_html.php) (и других подобных) имеется расширенная таблица из 140 цветов и удобное средство для их подбора в шестнадцатеричном формате.

### Пример 3. Изменение цвета текста

```
<font size=5 color=red face=Arial>П</font>ервая буква этого предложения будет написана шрифтом Arial, красным цветом и увеличенной.</font>
```

**П**ервая буква этого предложения будет написана шрифтом Arial, красным цветом и увеличенной.

Видоизменение текста - средства его форматирования, такие как выбор начертания шрифта и использование эффектов, позволяющих менять вид текста. В таблице перечислены основные теги, которые применяются для изменения оформления текста.

КодHTML	Описание	Пример
<code>&lt;b&gt;Текст&lt;/b&gt;</code>	Жирный текст	<b>Текст</b>
<code>&lt;i&gt;Текст&lt;/i&gt;</code>	Курсивное начертание текста	<i>Текст</i>
<code>&lt;u&gt;Текст&lt;/u&gt;</code>	Подчеркнутый текст	<u>Текст</u>
<code>&lt;sup&gt;Текст&lt;/sup&gt;</code>	Верхний индекс	$e=mc^2$

<code>&lt;sub&gt;Текст&lt;/sub&gt;</code>	Нижний индекс	H <sub>2</sub> O
<code>&lt;strike&gt;Текст&lt;/strike&gt;</code>	Зачеркнутый текст	<del>Текст</del>
<code>&lt;pre&gt;Т е к с т&lt;/pre&gt;</code>	Текст пишется как есть, включая все пробелы	Т е к с т
<code>&lt;em&gt;Текст&lt;/em&gt;</code>	Курсивный текст	<i>Текст</i>
<code>&lt;strong&gt;Текст&lt;/strong&gt;</code>	Жирный текст	<b>Текст</b>

Обычно для создания верхнего или нижнего индекса используется тег **small**, делающий индекс меньше по размеру основного шрифта.

**Пример 4. Создание нижнего индекса**

```
<b>Формула серной кислоты:</b>
<i>H<sub><small>2</small></sub>SO<sub><small>4</small></sub></i>
```

**Формула серной кислоты:**

*H<sub>2</sub>SO<sub>4</sub>*

**Выравнивание текста**

Выравнивание текста определяет его внешний вид и ориентацию краев абзаца и может выполняться по левому, правому краю, по центру или по ширине.

КодHTML	Описание	Пример
<code>&lt;p&gt;Текст&lt;/p&gt;</code>	Добавляет новый параграф, по умолчанию выровненный по левому краю. Перед параграфом автоматически добавляется пустая строка.	Текст
<code>&lt;p align=left&gt;Текст&lt;/p&gt;</code>	Выравнивание по левому краю.	Текст
<code>&lt;p align=right&gt;Текст&lt;/p&gt;</code>	Выравнивание по правому краю.	Текст
<code>&lt;p align=center&gt;Текст&lt;/p&gt;</code>	Выравнивание по центру.	Текст
<code>&lt;p align=justify&gt;Текст&lt;/p&gt;</code>	Выравнивание по ширине.	Текст по ширине

**Задание 2**

1. Поместите на HTML-страницу следующий код:

```
<html>
<body>
```

```
<font size="3" face="Times, Palatino, serif" color="#FF0000">Это пример  
№ 1</font> <br>  
<font size="2" face="Arial, Helvetica, nonserif" color="#008000">Это  
пример № 1</font> <br>  
<font size="5" face="Verdana, Arial, Courier, Times"  
color="#FF8000">Это пример№ 1</font>  
</body>  
</html>
```

2. Загрузите страницу в браузер и посмотрите, что получилось.

Это пример № 1  
Это пример № 1  
Это пример № 1

3. Как правило, шрифты типа serif используются для текстов, а шрифты типа nonserif используются для заголовков. Однако, если размер шрифта небольшой тексты serif могут оказаться неразборчивыми. Поэтому старайтесь не использовать минимальные размеры шрифтов. Для упражнения поместите на страницу текст с разными размерами шрифта serif и оцените возможность их легкого прочтения.

4. Иногда для заглавных букв используют размер шрифта, отличный от размера шрифта для остального текста. Поместите на страницу следующую конструкцию и посмотрите в браузере, что получилось:

```
<font size="5">Д</font>обро пожаловать на мою страницу!
```

Добро пожаловать на мою страницу!

5. Для заголовков часто используют специальные тэги заголовков: <h1>, <h2>, ...<h6>.

Используйте следующие примеры для отображения заголовков:

```
<h1>Заголовок 1</h1>  
<h2> Заголовок 2</h2>  
<h6> Заголовок 6</h6>
```

# Заголовок 1

## Заголовок 2

Заголовок 6

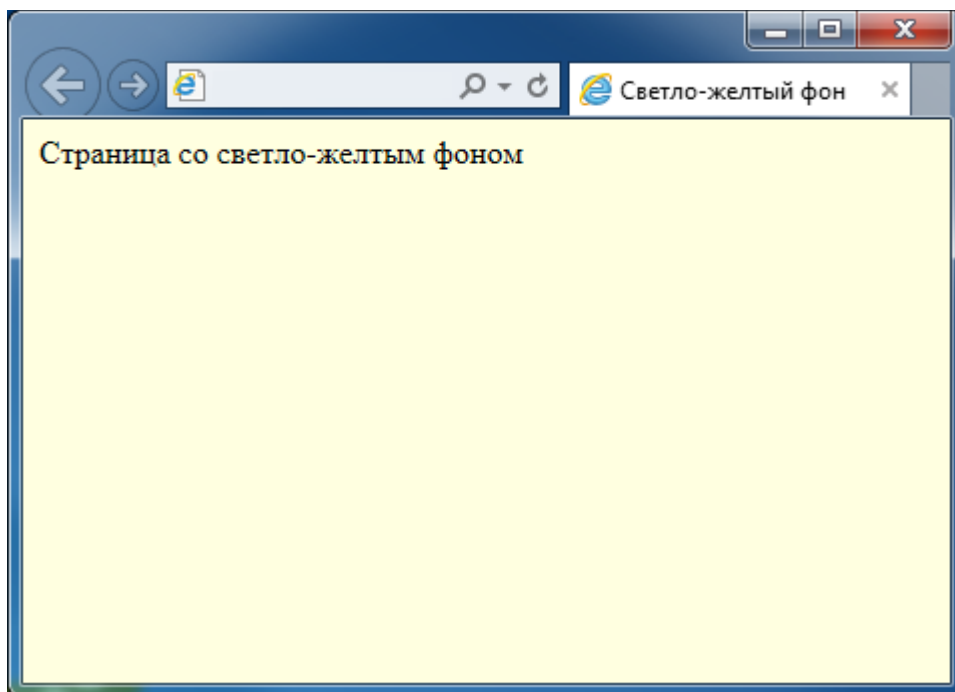
### Изменения фонового оформления страницы

У тега **<BODY>** есть несколько атрибутов, позволяющих поменять фон на странице, цвет текста, а также задать отступы от границ окна.

Цвет фона страницы можно задать с помощью атрибута **bgcolor**.

#### Пример 5. Страница со светло-желтым фоном

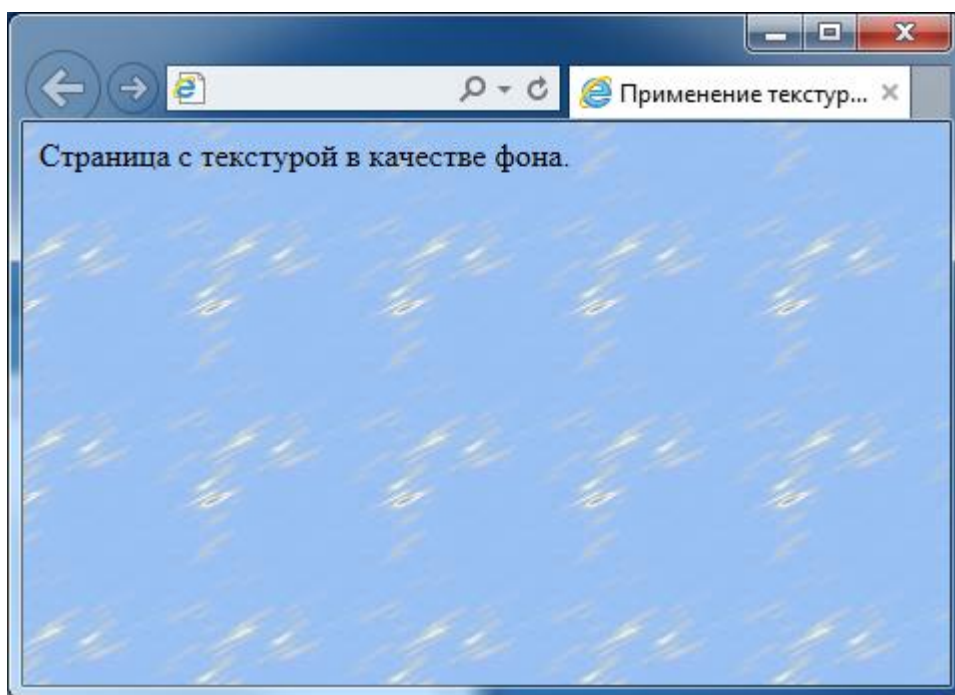
```
<html>
<head>
  <title>Светло-желтый фон</title>
</head>
<body bgcolor="lightyellow">
  Страница со светло-желтым фоном
</body>
</html>
```



Атрибут **bgcolor** устанавливает одноцветный фон на странице. Но в качестве фона может быть использована и повторяющаяся картинка, или так называемая текстура. Для этого служит следующий атрибут **background**, позволяющий установить текстурное изображение в качестве фона Web-страницы.

### Пример 6. Страница с изображением в качестве текстуры

```
<html>
<head>
  <title>Применение текстурного фона</title>
</head>
<body background="texture.jpg">
  Страница с текстурой в качестве фона.
</body>
</html>
```



В данном примере в качестве текстуры фона использовалось изображение в файле `texture.jpg`. Чтобы браузер смог найти изображение-источник, оно должно быть расположено в одной папке с HTML-файлом, для которого устанавливается текстура. В противном случае необходимо прописывать полный адрес изображения источника.

### Задания для самостоятельной работы

1. Создайте HTML-документ и оформите в нем текст в соответствии с образцом вашего варианта.
2. С использованием таблицы цветов подберите удобный для восприятия цвет фона страницы.
3. Сохраните как html-страницу и прикрепите в ЭИОС к заданию.

## Распределение вариантов

ФИО	№ варианта
Боролдоева Валентина Валерьевна	Вариант №1
Будлянский Дмитрий Александрович	Вариант №2
Исманов Никита Борисович	Вариант №3
Кучинская Кристина Владимировна	Вариант №4
Макарова Ольга Александровна	Вариант №5
Мартынов Алексей Дмитриевич	Вариант №6
Мельников Владислав Игоревич	Вариант №7
Пурбуев Чингис Эрдыниевиич	Вариант №8
Серебренникова Алина Юрьевна	Вариант №9
Титова Александра Павловна	Вариант №10
Файзулин Рустам Рашидович	Вариант №8
Чернышев Константин Витальевич	Вариант №7
Бобровникова Ксения Алексеевна	Вариант №6
Трусов Сергей Константинович	Вариант №1

## ВАРИАНТЫ

### Вариант 1

**Пусть вечно спорят  
дверь с ключом  
о первенстве и важности  
Главнее нету**

**В споре том  
Глазка замочной скважины.**

**Вариант 2**

Есть в близости  
людей заветная черта,  
Ее не перейти влюбленности и страсти, —  
*Пусть в жуткой*  
ТИШИНЕ сливаются уста,  
*И сердце рвется от любви на части.*

**Вариант 3**

Все, что в сердце  
твоим туманится,  
Станет ясно в моей тишине,  
*И когда он с тобой расстанется,*  
Ты признаешься только мне.

**Вариант 4**

Любовь глубокой  
нежности полна,  
В соблазнах, *горестях закалена,*  
Крепка в разлуке, вдалеке — горда,  
*Все та же — чудо, долгие года.*

**Вариант 5**

Жизнь не состязание в борьбе.  
*С юмором живи,*  
*но без усмешки.*  
Силы пробуждать учись в себе  
И живи *обдуманно*, без спешки.



**Вариант 6**

**О**ктябрь уж наступил,  
Уж роща **отряхает** последние листы  
С нагих своих ветвей.  
**Д**охнул осенний **хлад**,  
Дорога промерзает,  
Журча, <sup>еще</sup> бежит  
**З**а мельницу ручей

**Вариант 7**

И в ночи январской,  
**б**еззвездной,  
Сам *дивясь* **небывалой** судьбе,  
**В**озвращенный из смертной **бездны**,  
Ленинград *салютует* себе.

**Вариант 8**

**А** тебе еще мало по-русски,  
**И** ты хочешь на всех языках  
**З**нать, как **круты подъемы** и спуски  
И ~~почем~~ у нас  
**совесть и страх.**

**Вариант 9**

Что войны, что чума? - конец им **виден** **скорый**,  
Им приговор почти произнесен.  
**Но** кто нас **защитит** от ужаса, **который**  
**Был бегом** времени  
когда-то наречен?

**Вариант 10**

*Дорогою ценой*

и нежданной

Я узнала, что **помнишь** и **ждешь**.

А быть может,

**и место найдешь**

**Ты - могилы моей безымянной.**