

Міністерство освіти і науки України
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
«ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»
Кафедра Обчислювальна техніка та програмування

Альбом звітів з
лабораторних робіт по курсу «Веб-програмування»

Виконав ст.гр. КІТ–36
Надірян Г.О.
Перевірив
Октябрюва О.В.

Лабораторна робота №1

Тема: Подання тексту на веб-сторінці засобами HTML. Подання тексту на веб-сторінці засобами CSS

Мета: Вивчення засобів і можливостей HTML для завдання розміру, кольору, стилю і інших параметрів тексту на Web-сторінці.

Вивчення засобів і можливостей CSS для завдання розміру, кольору, стилю і інших параметрів тексту на веб-сторінці

Індивідуальні завдання

Реферат на довільну тему оформити у вигляді HTML-документа, реалізація якого у вигляді Web-сторінки повинна містити:

- Заголовок;
- текст, що складається з параграфів, окремі слова або пропозиції якого повинні бути:
 - різних кольорів (не менше трьох);
 - різного розміру (не менше трьох);
 - представлені різними типами шрифтів (не менше двох);
 - мати фізичне і логічне форматування;
 - вирівняні;
 - задані у вигляді аббревіатури;
 - прокоментовані.
- горизонтальну лінію.

Колір тексту і фону, вигляд фізичного і логічного форматування, а також вид вирівнювання беруться з табл. 7 відповідно до номером варіанта (номером студента в журналі). Інші параметри вибираються довільно.

№	Цвет		Форматирование		Выравнивание
	текста	фона	физическое	логическое	
12	зеленый	белый	pre	samp	по центру

Довільний текстовий документ (реферат, повідомлення, оголошення та ін.) Подати за допомогою HTML і CSS у вигляді веб-сторінки, яка повинна містити:

- Заголовок;
- текст, розбитий на параграфи;

- окремі фрагменти тексту (пропозиції, слова або символи) повинні бути задані згідно табл. 1:
 - за допомогою зовнішнього завдання CSS;
 - за допомогою внутрішнього завдання CSS;
 - за допомогою вбудованого завдання CSS;
 - з використанням класів (не менше трьох).

У таблиці 3 для кожного варіанту і для кожного виду завдання CSS наведені значення властивостей CSS або назви властивостей CSS. В останньому випадку значення для вказаного властивості вибирається студентом довільно.

№	Свойства CSS		
	Внешнее задание CSS	Внутреннее задание CSS	Встроенное задание CSS
12	line-height, small-caps	color, bold, word-spacing	center, italic

Хід виконання

```
<!DOCTYPE html> <!--
HTML5 использует только одну форму декларации для обозначения типа документа
-->
<html lang="en"> <!--
<html>.. </html> - root element корневой элемент. Задаём англ.язык
-->
<head> <!--
Предназначен для указания HTML-элементов,
содержащих метаданные, т.е. такие данные, которые не являются содержимым веб-страницы,
а представляют собой информацию о веб-странице
-->
<meta charset="window-1251"> <!--
с помощью атрибута charset –
задается кодировка символов текстового содержимого веб-страницы, а для кириллицы указана кодировка "window 1251"
-->
<title>Лабораторная работа 1</title> <!--
задаём текст, который всплывает над данным элементом при наведении на него указателя мышки
-->
</head>
<!-- Определяет отступ от левого края окна браузера до контента веб-страницы. В нашем случае отступ на 50 пикселей -->
<!-- Цвет текста: Зеленый цвет, а цвет фона – белый: -->
<body leftmargin="50" text="green" bgcolor="ffffff">
<h2 title="Выравнен по центру" align="center">Потоки выполнения</h2>
<p align="center">
<font color="black">
<!--ЛИНИИЯ -->
<hr color=#black size="10" width="100%">

<p align="middle">
<font size="5" color="blue" face="Arial Cyr, Arial, Helv D1">
Это, без сомнения, очень важный вопрос. На подобных
понятиях основан весь рынок персональных компьютеров.
Предполагается, что даже новичок понимает, сколько мегов
одной и гигаов другой потребуется его программам. Считается
также, что в самом начале знакомства с компьютером пользователь
постигает концепцию файла и представляет себе, как
файлы загружаются из постоянной в оперативную память, а
затем из оперативной записываются обратно в постоянную.
На вопрос о постоянстве и оперативности часто отвечают,
используя такую аналогию: «Считайте, что оперативная память
— это ваш рабочий стол, а постоянная — шкаф с папками». И
думают, что это прекрасное сравнение! Но я его таковым не
считаю: создается впечатление, что архитектура компьютера
срисована с обычной конторы. Правда же заключается в том,
что различие между постоянной и оперативной памятью искусственно
и существует лишь потому, что нам до сих пор не
удалось создать накопитель данных, который был бы одновременно
```

быстрым, объемным и способным хранить информацию в течение долгого времени. То, что сегодня называется архитектурой Неймана — доминирующая компьютерная архитектура в течение последних 50 лет, — прямое следствие этого технического несовершенства.

<p>

Еще меня как-то спросили: «Почему программы для Macintosh нельзя запускать под Windows?» Я уже было открыл рот, чтобы ответить, но вдруг осознал, что ответ будет содержать гораздо больше технических подробностей, чем рассчитывает услышать просивший.

действие современного компьютера состоит из бесчисленны

<p>Это, <mark>отмеченный</mark> элементом mark, текст.

<abbr title="Объектно-ориентированное программирование">ООП</abbr>

<pre>Текст без предварительного форматирования. </pre>

</body>

</html>

Результати роботи програми

Потоки выполнения

Это, без сомнения, очень важный вопрос. На подобных понятиях основан весь рынок персональных компьютеров. Предполагается, что даже новичок понимает, сколько мегов одной и пиков другой потребуется его программ. Считается также, что в самом начале знакомства с компьютером пользователь постигает концепцию файла и представляет себе, как файлы загружаются из постоянной в оперативную память, а затем из оперативной записываются обратно в постоянную. На вопрос о постоянстве и оперативности часто отвечают, используя такую аналогию: «Считайте, что оперативная память — это ваш рабочий стол, а постоянная — шкаф с папками». И думают, что это прекрасное сравнение! Но я его таковым не считаю: создается впечатление, что архитектура компьютера срисована с обычной канторы. Правда же заключается в том, что различие между постоянной и оперативной памятью искусственно и существует лишь потому, что нам до сих пор не удалось создать накопитель данных, который был бы одновременно быстрым, объемным и способным хранить информацию в течение долгого времени. То, что сегодня называется архитектурой Неймана — доминирующая компьютерная архитектура в течение последних 50 лет, — прямое следствие этого технического несовершенства.

Еще меня как-то спросили: «Почему программы для Macintosh нельзя запускать под Windows?» Я уже было открыл рот, чтобы ответить, но вдруг осознал, что ответ будет содержать гораздо больше технических подробностей, чем рассчитывает услышать просивший. Я хочу, чтобы книга Код помогла вам разобраться в подобных вещах, причем не абстрактно, а глубоко, практически на уровне инженера или программиста. Надеюсь также, что в ней компьютер предстанет перед вами как одно из наиболее выдающихся изобретений XX века и вы поймете, что он прекрасен сам по себе, даже без изысканных метафор и сравнений. Компьютер представляет собой иерархию, в основании которой лежат транзисторы, а на вершине — информация, отображаемая на мониторе. Переходы с одного уровня этой иерархии на следующий — а книга Код построена именно так — далеко не так тяжелы, как думает большинство. Конечно, действие современного компьютера состоит из бесчисленны

Это, **отмеченный** элементом mark, текст.

ООП

Текст без предварительного форматирования.

Рисунок 1 – работа програми

Хід виконання

Файл:2.css

```
h1 {
  text-align:center;
  text-transform:uppercase;
  color:#036;
  font-size: 18px; /* 28px */
  font-family:Geneva;
  font-style:normal; /*обычное начертание текста*/
  font-variant: small-caps; /*small-caps – делает строчные буквы заглавными, но меньшего
размера*/
}
p {
  text-indent:1cm; /*Устанавливает величину отступа первой строки блока текста*/
  color:black;
  font-size:15px;
  line-height: 1.0; /*Устанавливает интерлиньяж (межстрочный интервал) текста*/
}
```

}

Файл lab2.html

```
<!DOCTYPE html>
<html>
<head>
<title> Шестое вымирание</title>
<link rel="stylesheet" type="text/css" href="2.css">
<style type="text/css">
```

```
p.decor_text_one_par {
color:rgba(89%,0%,59%,1);
word-spacing:1mm;
color:black /* Цвет текста */
font-weight: bold
font:1.5em "Arial Cyr", Arial, sans-serif;
}
```

```
p.decor_text_two_par {
color:rgba(%0%,0%,40%,1);
word-spacing:1mm;
color:red; /* Цвет текста */
font-weight: bold
font:1.5em "Arial Cyr", Arial, sans-serif;
}
```

```
p.decor_text_three_par {
color:rgba(%10%,10%,10%,1);
word-spacing:1mm;
color:green; /* Цвет текста */
font-weight: bold
font:1.5em "Arial Cyr", Arial, sans-serif;
}
```

```
q {
font-style:italic;
text-align: center;
font-size:6.5mm;
quotes: "\201e" "\201c" /* Коды открывающей и закрывающей кавычек */
}
</style>
</head>
<body>
```

```
<h1> Часть первая </h1>
<h1> Черный генезис </h1>
<h1>  $\Sigma$  </h1>
<h1> Глава 1 </h1>
```

```
<p class="decor_text_one_par"> 27 апреля, 18 часов 55 минут по Тихоокеанскому летнему времени
Озеро Моно, штат Калифорния. Похоже на марсианский ландшафт.
Дженна Бек мысленно усмехнулась, услышав это расхожее описание озера Моно еще от одного туриста.
Пока последняя на сегодняшний день группа посетителей делала фотографии, она ждала у своего белого пикапа «Форд Ф-150» '
со звездой службы охраны национальных парков штата Калифорния на передней двери.
<p class = "decor_text_two_par">Потянув за твердые поля форменной шляпы, Дженна надвинула ее ниже на лоб и посмотрела на
солнце.
тянулись вдоль южной оконечности озера подобно окаменевшему лесу и спускались в воду.
```

<p class = "decor_text_three_par">Определенно, этот ландшафт казался неземным – но только никак не марсианским. Дженна хлопнула себя по руке, убивая москита, свидетельствовавшего о том, что среди этой пустынной красоты по-прежнему есть жизнь.

</body>
</html>

Результати роботи програми

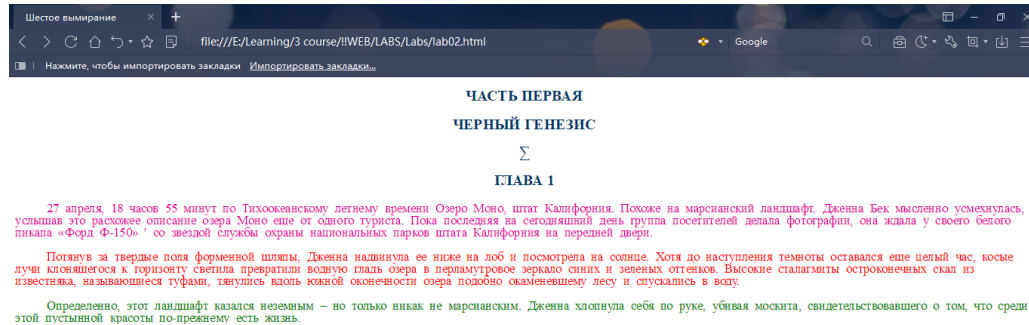


Рисунок 2 – работа програми

Висновки

В результаті виконання лабораторної роботи вивчили засоби і можливості HTML для завдання розміру, кольору, стилю і інших параметрів тексту на Web-сторінці. Вивчили засоби і можливості CSS для завдання розміру, кольору, стилю і інших параметрів тексту на веб-сторінці

ЛАБОРАТОРНАЯ РАБОТА №2

Тема: *Представление элементов на веб-странице. Представление списков на веб-странице.*

Цель: *Изучение и использование CSS-свойств для представления элементов на веб-странице. Изучение средств и возможностей HTML и CSS для представления на веб-странице списков с заданными параметрами*

Разработать Web-страницу, содержащую элементы D1, D2, D3, D4 и D5:

- D1 – блочный элемент прямоугольной формы, не имеющий границ (расположение на Web-странице, размеры, цвет символов и фона – на усмотрение студента);
 - D2 – это элемент D1, который имеет границы с такими параметрами:
 - стиль границы – см. колонку "Граница/style" табл. №5 (значения заданы в пикселях);
 - ширина границы – см. колонку "Граница/width" табл. №5 (значения заданы в пикселях);
 - цвет границы – на усмотрение студента;
 - D3 – это элемент D2, который имеет следующие скругления углов границ – см. колонку "Граница/radius" табл. №5:
 - 1 – 40%;
 - 2 – 20% 30%;
 - 3 – 10% 30% 50%;
 - 4 – 15% 25% 35% 45%;
 - D4 – это элемент D2, который имеет тень (тени) с такими параметрами:
 - смещение тени по горизонтали – колонка "Тень/h" табл. №5 (значения заданы в пикселях);
 - смещение тени по вертикали – колонка "Тень/v" табл. №5 (значения заданы в пикселях);
 - размытость тени – колонка "Тень/blur" табл. №5 (значения заданы в пикселях);
 - размер тени – колонка "Тень/spread" табл. №5 (значения заданы в пикселях);
 - цвет тени – на усмотрение студента;
 - D5 – это элемент D2, который имеет следующие преобразования:
 - смещение элемента – первое значение указывает смещение по горизонтали, второе – по вертикали, значения заданы в пикселях (см. колонку "Трансформация/translate" табл. №5);
 - масштабирование элемента – первое значение указывает во сколько раз нужно изменить размер элемента по горизонтали, второе – по вертикали, если указано значение "--", то по данному направлению изменять размер элемента не надо (см. колонку "Трансформация/scale" табл. №5);
 - вращение элемента задается в градусах (deg) – колонка "Трансформация/rotate" табл. №5);
 - наклон элемента задается в градусах (deg) – значение наклона вдоль оси X предваряется символом "X", значение наклона вдоль оси Y – символом "Y"; (см. колонку "Трансформация/skew" табл. №5);
- Сформировать на веб-странице с использованием элементов `` и `` четыре списка, один из которых является внешним списком, а остальные три – вложенными в него. При этом:

- маркерами для первого и второго списков являются CSS-свойства, указанные в колонках 1 и 2 табл.1;
- в качестве маркера для третьего списка использовать символы, указанные в колонке 3 табл.1;
- в качестве маркера для четвертого списка использовать произвольный рисунок;
- каждый из четырех списков должен содержать не менее пяти элементов;
- номер списка, который для данного варианта является внешним, указан в колонке 4 табл. 1.

Таблица 5 – Перечень заданий

№	Граница			Тень				Трансформация			
	style	width	radius	h	v	blur	spread	translate	scale	rotate	skew
1	dotted	3	1	20	10	5	3	100 100	1.5 0.5	10	X40
2	dashed	5	2	30	-40	0	16	200 0	1.1 1.3	20	Y15
3	solid	7	3	-5	15	10	5	0 250	-- 2.0	30	X30
4	double	9	4	-15	-10	10	5	-50 50	1.4 --	40	Y25
5	groove	11	1	16	20	12	4	300 0	0.6 1.2	50	X20
6	ridge	3	2	25	-8	4	2	0 300	0.4 1.6	60	Y35
7	inset	5	3	-9	5	3	8	400 -50	-- 0.75	70	X10
8	outset	7	4	-20	-12	9	11	350 0	1.7 0.9	-10	Y45
9	dotted	9	1	7	9	6	12	0 120	0.8 1.4	-20	X-10
10	dashed	11	2	12	-15	11	7	380 320	2.0 --	-30	Y-15
11	solid	3	3	-8	7	2	6	300 0	-- 1.3	-40	X-20
12	double	5	4	-18	-13	0	4	0 290	1. 15 1.8	-50	Y=25

Рисунок – 1 таблица заданий

Таблица 1 – Перечень заданий

№	Список №1	Список №2	Список №3	Номер
1	disc	decimal	!	1
2	circle	decimal-loading-zero	@	2
3	square	lower-roman	#	3
4	decimal	upper-roman	\$	4
5	decimal-loading-zero	lower-alpha	%	1
6	lower-roman	upper-alpha	^	2
7	upper-roman	lower-greek	&	3
8	lower-alpha	disc	*	4
9	upper-alpha	circle		1
10	lower-greek	square	<	2
11	disc	decimal	>	3
12	circle	decimal-loading-zero	?	4

Рисунок – 2 таблица заданий


```

<html>

<head>

<title>Переопределение свойства width</title>

<link rel="stylesheet" type="text/css" href="3.css">

<style>

div {

    color:red;

    background-color:blue;

    position:fixed;

        border-style:double; <!--Для границ-->

    border-color:red;

    width:5cm; <!-- фигура-->

    height:1cm; <!-- фигура-->

    margin-left:1;

        font-size:3mm;

    text-align:center;

}

</style>

</head>

<body>

<div style="top:1; padding:10px; border-width:0px;"> D1 – блочный элемент прямоугольной формы,

не имеющий границ (расположение на Web-странице, размеры, цвет символов и фона – на усмотрение студента);

</div>

<div style="top:2px; padding:15px; border-width:10px;border-color:pink; left: 1px; width:5px; height:50px;">D2 </div>

<div style="top:140px; padding:15px; border-width:10px;border-color:pink;border-radius:15% 25% 35% 45%; left: 1px; width:5px;

height:50px;">D3 </div>

<div style="top:270px; padding:15px; border-width:10px;border-color:pink;box-shadow: -18px 13px 0px 4px; left: 50px; width:5px;

height:50px;">D4 </div>

<div style="top:460px; padding:15px; border-width:10px;border-color:pink;border-radius:15% 25% 35% 45%; left:

44px; width:5px; height:50px;transform:translate(55);transform:skewY(25deg);transform:scale(1.15,1.8);transform:rotate(-50);">D5 </div>

</div>

</body>

</html>

```

Код программы lab4.html

```
<!DOCTYPE html>

<html>

<head>

<title>Лабораторная работа №4</title>

<link rel="stylesheet" href="my.css">

<style type="text/css">


ul.none li { /* Убираем исходные маркеры */

    list-style-type:none;

}


ul.none li::after { /* Добавляем маркер в виде символа "?" */

    content: "? ";

    color:purple;

    font-size:10mm;

}

ul.im {

    list-style-image: url(windows.jpg);

}


</style>

</head>

<body>


<h1>Представление списков на веб-странице</h1>

<!-- <dl> -->

<!-- <dt>Списки, которые поддерживает HTML: -->

<dd>Цитаты Цицерона;

<ul>

<ul class = "im">

    <li>EX MALIS ELIGERE MINIMA</li>

    <li>HISTORIA EST MAGISTRA VITAE</li>

    <li>ILIAS MALORUM</li>

    <li>LATINE LOQUI</li>

    <li>O TEMPORA, O MORES</li>

</ul>

</ul>
```

```
<pre>Список римских императоров:</pre>
```

```
<ol>
```

```
<li>Октавиан Август;
```

```
<li>Тиберий;
```

```
<li>Клемент;
```

```
<li>Калигула;
```

```
<li>Клавдий;
```

```
<li>Скрибониан;
```

```
<li>Нерон.
```

```
</ol>
```

```
<pre>Список процессоров intel:</pre>
```

```
<dd>
```

```
<ul>
```

```
<li>Intel Celeron
```

```
<li>Intel Pentium.
```

```
<li>Intel Core i3
```

```
<li>Intel Core i5.
```

```
<li>Intel Core i7.
```

```
<li>Intel Core E3
```

```
</ul>
```

```
<pre>Актуальные вопросы, на которые у науки нет ответов:</pre>
```

```
<ul class="none">
```

```
<li>Каков состав Вселенной
```

```
<li> Почему нам снятся сны
```

```
<li>Что случается с простыми числами
```

```
<li>Когда роботы станут нашими собеседниками
```

```
<li>Что представляет собой "чёрная дыра"
```

```
</ul>
```

```
</ul>
```

```
</body>
```

```
</html>
```

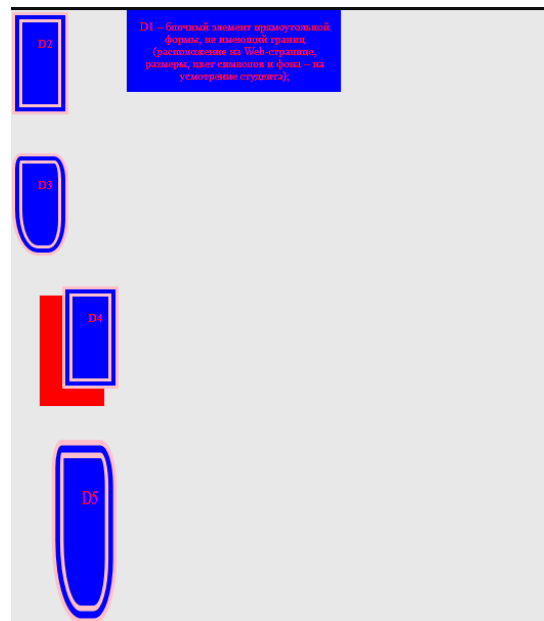







Рисунок -3 результат работы файла lab3.html

Представление списков на веб-странице

Цитаты Цицерона;

-  EX MALIS ELIGERE MINIMA
-  HISTORIA EST MAGISTRA VITAE
-  ILIAS MALORUM
-  LATINE LOQUI
-  O TEMPORA, O MORES

Список римских императоров:

01. Октавиан Август;
02. Тиберий;
03. Клемент;
04. Калигула;
05. Клавдий;
06. Скрибоний;
07. Нерон.

Список процессоров intel:

- o Intel Celeron
- o Intel Pentium.
- o Intel Core i3
- o Intel Core i5.
- o Intel Core i7.
- o Intel Core E3

Актуальные вопросы, на которые у науки нет ответов::

Каков состав Вселенной ?

Рисунок - 4 результат работы файла lab4.html

Вывод

Изучили и использовали CSS-свойств для представления элементов на веб-странице. Изучили средства и возможности HTML и CSS для представления на веб-странице списков с заданными параметрами

ЛАБОРАТОРНАЯ РАБОТА №3

Тема: *Представление таблиц на веб-странице. Представление изображений, видеофрагментов и звука на веб-странице*

Цель: *Изучение средств и возможностей HTML и CSS для представления на веб-странице таблиц с заданными параметрами. Изучение средств и возможностей HTML и CSS для представления на веб-странице изображений, видеофрагментов и звука*

Индивидуальные задания

➤ На Web-странице разработать таблицу, которая должна иметь:

- заголовки столбцов;
- размер, заданный в виде $M \times N$, где M – число строк таблицы; N – число столбцов таблицы (см. колонку "Размер" табл. 3);
- объединенные столбцы, заданные в виде $M(N_1-N_2)$, где M – номер строки, N_1, N_2 – начальный и конечный номера объединенных столбцов (см. колонку "Объединение/столбцов" табл. 3);
- объединенные строки, заданные в виде $N(M_1-M_2)$, где N – номер столбца, M_1, M_2 – начальный и конечный номера объединенных строк (см. колонку "Объединение/строк" табл. 3);
- элемент таблицы, тип которого указан в колонке "Элемент таблицы/Тип" табл. 3), а свойства границ – в колонке "Элемент таблицы/Свойства границ" табл. 3);
- ячейку таблицы, фон которой задан в виде линейного градиента;
- ячейку таблицы, фон которой задан в виде радиального градиента;
- ячейки таблицы, фон которых имеет цвет, который выбирается произвольно.

Таблица 3 – Перечень заданий

№	Размер таблицы	Объединение		Элемент таблицы				
		столбцов	строк	Тип	Свойства границ			
					вид	тип	толщина	цвет
1	3x7	1(3-5)	6(2-3)	th	border-bottom	dashed	10	красный
2	5x5	4(2-4)	5(1-5)	td	border	dotted	7	синий
3	6x4	5(1-2)	2(1-5)	th	border-left	double	5	желтый
4	2x8	2(3-7)	3(1-2)	table	border-right	outset	11	зеленый
5	4x5	3(4-5)	1(1-3)	th	border-top	inset	4	бирюзовый
6	7x4	6(1-4)	2(2-6)	td	border-bottom	double	thick	лиловый
7	4x8	3(3-7)	2(2-4)	td	border	dashed	12	темно-бирюзовый
8	6x6	3(1-6)	5(4-5)	table	border-left	dotted	9	темно-красный
9	4x7	1(2-5)	5(1-3)	th	border-right	inset	6	темно-синий
10	6x3	5(2-3)	3(2-6)	td	border-top	dashed	10	темно-зеленый
11	8x3	7(1-2)	3(1-7)	table	border-bottom	dotted	thick	темно-лиловый
12	7x5	5(3-5)	2(1-3)	table	border	outset	medium	светло-бирюзовый
13	7x7	2(4-7)	7(2-5)	th	border-left	inset	11	светло-красный
14	3x9	3(3-9)	8(2-3)	td	border-right	dotted	8	светло-синий
15	9x3	3(1-3)	2(5-9)	th	border-top	double	3	светло-зеленый
16	4x8	4(3-6)	7(2-4)	table	border-bottom	dashed	thick	светло-бирюзовый
17	6x5	2(3-4)	4(2-5)	th	border	dotted	9	красный

Рисунок – 1 перечень заданий

➤ Разработать HTML-документ, задающий на веб-странице:

- фон в виде изображения, который при прокрутке страницы остается неподвижным;
- область с прозрачным фоном и область с непрозрачным фоном;
- изображение в трех модификациях;
 - в рамке;
 - с изменением размера;
 - урезанное до одной четвертой части с абсолютными координатами;
- видеофрагмент с использованием элемента *video*;
- воспроизведение звука с использованием элемента *audio*.

Таблица 8 – Перечень заданий

№	Изображение 1		Изображение 2		Изображение 3	
	Выравнивание	Рамка	Выравнивание	Изменение/число раз	Расположение рисунка	Четверть рисунка
1	left	outset	right	больше/3	левый/верхний	правая/нижняя
2	middle	inset	bottom	меньше/2	правый/верхний	левая/нижняя
3	right	ridge	top	длиннее/4	левый/нижний	правая/верхняя
4	bottom	groove	left	короче/5	правый/нижний	левая/верхняя
5	top	outset	middle	больше/5	середина/верх	правая/нижняя
6	left	inset	right	меньше/3	середина/низ	левая/нижняя
7	middle	ridge	bottom	длиннее/6	левый/верхний	правая/верхняя
8	right	groove	top	короче/2	правый/верхний	левая/верхняя
9	bottom	outset	left	больше/5	левый/нижний	правая/нижняя
10	top	inset	middle	меньше/4	правый/нижний	левая/нижняя
11	left	ridge	right	длиннее/8	середина/верх	правая/верхняя
12	middle	groove	bottom	короче/4	середина/низ	левая/верхняя

Рисунок -2 перечень заданий

Код программы lab5.html

```

<!DOCTYPE html>

<html>

  <head>

    <title>colspan rowspan</title>

    <link rel="stylesheet" href="file:///C:/Users/blac_order/my.css">

    <style type="text/css">

      table

```

```

        {
        border: medium solid turquoise;
        border-style:outset
        border-width:medium
        text-align:center;

        };
</style>

</head>
<body>
    <h4>Таблица с объединенными колонками:</h4>

    <table border="1">
    <tr>
        <th>Name</th>
        <th rowspan = "3">Telephone</th>
        <th>Telephone</th>
        <th>Telephone</th>
        <th>Telephone</th>
    </tr>
    <tr>
        <td>Hamlet</td>
        <td>073 022 40 71</td>
        <td>073 022 40 71</td>
        <td>073 022 40 71</td>

    </tr>
    <tr>
        <td>Hamlet</td>

```


<td>073 022 40 71</td>
<td>073 022 40 71</td>
<td>073 022 40 71</td>
</tr>
<tr>
<td>Hamlet</td>
<td>073 022 40 71</td>
<td>073 022 40 71</td>
<td>073 022 40 71</td>
<td>073 022 40 71</td>
</tr>
<tr>
<td>Hamlet</td>
<td>073 022 40 71</td>
<td colspan="2" style="background-color:#e0e0ff;text-align:center;"> 073 022 40 71</td>
<td>073 022 40 71</td>
</tr>
<tr>
<td>Hamlet</td>
<td>073 022 40 71</td>
<td>073 022 40 71</td>
<td>073 022 40 71</td>
<td style="text-align:right; background:repeating-linear-gradient(to right, lightblue, yellow 50%);">073 022 40 71 </td>
</tr>
<tr>

```

        <td>Hamlet</td>
        <td>073 022 40 71</td>
        <td>073 022 40 71</td>
        <td>073 022 40 71</td>
        <td style="background: radial-gradient(lightgreen,
lightblue);">073 022 40 71</td>
    </tr>
</table>
</body>
</html>

```

Код программы lab6.html

```

<!DOCTYPE html>
<html>
<head>
<title>Видеофрагменты и звук на Web-странице</title>

<style type="text/css">

body {
    color:white;

    background:url(b.jpg) fixed;
}
p {
    text-indent:0;
    color:white;
    font-style:italic;

```

```
font-weight:bold;
font-size:7mm;
}
```

```
td {
    color:white;

    font-size:5.5mm;
    font-weight:bold;
    width:200px;
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<h1 align="center">Lab06</h1>
```

```
<table width="80%" height="100" align="center" style="text-align:center; border:groove 500
white">
```

```
<tr>
```

```
<td style="background-color:rgb(160, 111, 255)">Непрозрачный фон
```

```
<td style="background-color:transparent">Прозрачный фон
```

```
<td style="background-color:rgba(160, 111, 255, 0.5)">Полупрозрачный фон </tr>
```

```
</table><br><br>
```

```
<p class="clear">
```

```

```

```
<p align="bottom "><p align = "center"> Фрагмент из фильма "Love story 1970 "
```

```

```

```
<p class="clear">
```

```
<p align="center ">
```

```
<video width="200" height="200" align=center controls ">
```

```
<source src="story.webm">
```

```
</video>
```

```
<p>Музыка из фильма:
```

```
<BR>
```

```
<audio controls>
```

```
<source src="story.mp3" type="audio/mpeg" codecs="mp3"/>
```

```
</audio>
```

```
<p align = "center">
```

```

```

```
</body>
```

```
</html>
```

Работа програми

Таблица с объединенными колонками:

Name		Telephone	Telephone	Telephone
Hamlet	Telephone	073 022 40 71	073 022 40 71	073 022 40 71
Hamlet		073 022 40 71	073 022 40 71	073 022 40 71
Hamlet	073 022 40 71	073 022 40 71	073 022 40 71	073 022 40 71
Hamlet	073 022 40 71	073 022 40 71		073 022 40 71
Hamlet	073 022 40 71	073 022 40 71	073 022 40 71	073 022 40 71
Hamlet	073 022 40 71	073 022 40 71	073 022 40 71	073 022 40 71

Рисунок – 3 работа програми lab5.html

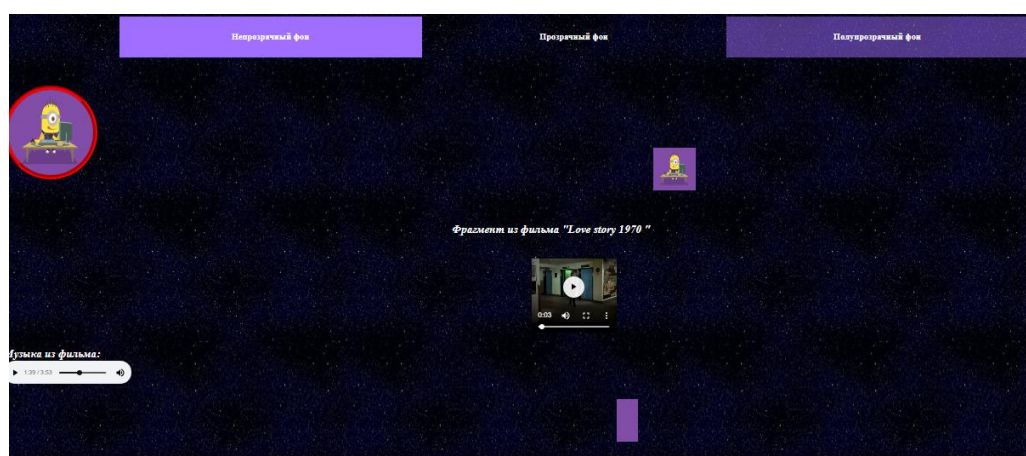


Рисунок - 4 работа програми lab6.html

Вывод

Изучили средства и возможности HTML и CSS для представления на веб-странице таблиц с заданными параметрами. Изучили средства и возможности HTML и CSS для представления на веб-странице изображений, видеофрагментов и звука

ЛАБОРАТОРНАЯ РАБОТА №4

Тема: Работа со ссылками на веб-странице. Задание форм на веб-странице

Цель: Изучение средств и возможностей HTML и CSS для представления на веб-странице ссылок и сегментированной графики. Изучение средств и возможностей HTML для создания форм на веб-странице.

Индивидуальные задания

В соответствии с технологией "клиент-сервер" в данной лабораторной работе разработать клиентскую часть, осуществляющую прием от клиента данных и передачу их на сервер (серверная часть, выполняющая обработку переданных данных, разрабатывается в лаб. раб. №17).

Клиентскую часть реализовать в виде веб-страницы с HTML-формой, которая имеет следующие значения атрибутов:

- *method* – указывается в соответствии с полем "Метод" табл. 1 индивидуального задания;
- *action* – указывается путь к одному из файлов с HTML-документом (или URL-адрес серверной CGI-программы)

и содержит следующие элементы:

- текстовое поле для ввода наименования студенческой группы;
- текстовое поле для ввода фамилии и инициалов студента;
- текстовые поля для ввода пяти чисел (целых или с плавающей точкой), подлежащих обработке;
- переключатель, имеющий значения:
 - "Полный";
 - "Сжатый",

для задания одного из двух возможных вариантов вывода результатов обработки данных, выполняемого в лаб. раб. №17). Переключатель реализовать в виде тега, указанного в поле "Тег1" табл. 1;

- переключатель для задания нескольких задач обработки введенных данных. Реализовать в виде тега, указанного в поле "Тег2" табл. 1. Возможные значения переключателя приведены в поле "Задачи" табл. 1, при этом:
 - вместо "S" необходимо указать "Сумма" (задача: определить сумму переданных чисел);
 - вместо "P" необходимо указать "Произведение" (задача: определить произведение переданных чисел);
 - вместо "M" необходимо указать "Среднее значение" (задача: определить среднее арифметическое значение переданных чисел);
 - вместо "MAX" необходимо указать "Максимум" (задача: определить максимальное число);
 - вместо "MIN" необходимо указать "Минимум" (задача: определить минимальное число);
 - вместо "RU" необходимо указать "По возрастанию" (задача: отсортировать переданные числа по возрастанию значений);

- вместо "RD" необходимо указать "По убыванию" (задача: отсортировать переданные числа по убыванию значений).

Значение переключателя будет использоваться при выполнении лаб. раб. №17 для выбора вычислительных задач;

- кнопка "СБРОС";
- элемент формы для передачи данных на сервер указывается в соответствии с полем "Передача" табл. 1.

Таблица 1 – Перечень заданий

№	Метод	Ter №1	Ter №2	Задачи	Передача
1	get	input	input	S, MAX, RU	submit
2	post	input	select	P, MAX, RU	image
3	get	select	input	S, MIN, RU	submit
4	post	input	input	P, MIN, RU	image
5	get	input	select	S, MAX, RD	submit
6	post	select	input	P, MAX, RD	image
7	get	input	input	S, MIN, RD	submit
8	post	input	select	P, MIN, RD	image
9	get	select	input	MAX, RU, M	submit
10	post	input	input	MIN, RU, M	image
11	get	input	select	MAX, RD, M	submit
12	post	select	input	MIN, RD, M	image

Рисунок -1 перечень заданий

Индивидуальные задания

Разработать HTML-документ, задающий на веб-странице:

- ✗ обычную ссылку;
- ✗ две локальные ссылки;
- ✗ ссылку в виде сегментированной графики с несколькими областями чувствительности;
- ✗ курсор с требуемым изображением (параметры всех трех ссылок и курсора приведены в таблице 1).

Таблица 1 – Перечень заданий

№	Ссылка в виде	Локальная ссылка в виде	Локальная ссылка в виде	Области чувствительности сегментированной графики	Вид курсора
1	изображения	кнопки	текста	3-угольник, квадрат	text
2	кнопки	текста	изображения	квадрат, окружность	help
3	кнопки с изобр.	кнопки	текста	окружность, 5-угольник	wait
4	текста	кнопки	изображения	4-угольник, окружность	e-resize
5	изображения	текста	кнопки	5-угольник, квадрат	n-resize
6	кнопки	изображения	текста	3-угольник, 6-угольник	ne-resize
7	кнопки с изобр.	кнопки	текста	квадрат, 5-угольник	nw-resize
8	текста	кнопки с изобр.	изображения	окружность, квадрат	url
9	изображения	кнопки с изобр.	текста	4-угольник, 5-угольник	help
10	кнопки	кнопки с изобр.	текста	5-угольник, 4-угольник	wait
11	кнопки с изобр.	текста	изображения	3-угольник, 5-угольник	e-resize
12	текста	изображения	кнопки	квадрат, 6-угольник	n-resize
13	изображения	кнопки	текста	окружность, квадрат	ne-resize

Рисунок -2 перечень заданий

Код программы Lab7.html

```
<!DOCTYPE html>

<html>

<head>

<title>—сылки на веб-странице</title>

<link rel="stylesheet" href="my.css">

<link rel="stylesheet" href="/my.css">

<style type="text/css">

table {

    color:#0000C0;

    font-size:6mm

}

td { height:40px; }

iframe {

    border-width:0px;

    margin-left:3cm;

    height:100px;

    width:140px;

}

</style>

</head>

<body>

<a id="beg"></a><br>

<p align="center">—сылки на веб-странице</p>

<p align = "center">ҒјЅќ–ј“ќ–Ќјя –јЅќ“ј €7 <br>

“ема: –абота со ссылками на веб-странице <br>

÷ель: »зучение средств и возможностей HTML и CSS длѧ представлениѧ на веб-странице

ссылок и сегментированной графики</p>

<iframe name="ff" src=""> </iframe>

<table width="60%" height="111%" align="center">
```


<tr><td>—ссылка со стандартными параметрами:

```
<td><a style = "cursor:n-resize"
href="https://www.youtube.com/watch?v=eap0oA46HPM">ссылка на ютуб</a>
```

<tr><td>—ссылка в виде рисунка:

```
<td><a style = "cursor:n-resize" href="lab.htm" title="»спользование графики в ссылках">
</a>
```

<p>—ссылка в виде кнопки</p>

```
<div style="margin:5px;">
<a style = "cursor:n-resize" href="palette.htm" class="LinkButton"> нопка</a>
</div>
```

<tr><td>—ссылка в виде сегментированной графики:

```
<td>
</table>
<map name="my_map">
<area shape="circ" coords="88,49,5 noref">
<area shape="rect" coords="10,6,50,30" style = "cursor:n-resize" href="prim2.htm"
alt="«адание параметров шрифтов">
<area shape="circ" coords="88,49,20" style = "cursor:n-resize" href="prim1.htm"
alt="»спользование CSS длѐ форматированиѐ текста">
</map>
<a id="text"></A><BR>
```

</body>

</html>

Код программы Lab8.html

```
<!DOCTYPE html>
```

```
<html>
```

```

<head>
<title>Формы на веб-странице </title>
<style type="text/css">

input, textarea, option {
    background-color:white;
    color:#004000;
    font-size:6mm;
}
.kn {
    background:white;
    border:outset 3px blue;
    font-size:4.5mm;
}
td {
    vertical-align:middle;
}
.newStyle1 {
}
.auto-style4 {
    font-family: Arial, Helvetica, sans-serif;
    margin-left: 160px;
}
</style>
</head>
<body>
    <h1 class="auto-style4">ЛАБОРАТОРНАЯ РАБОТА №8. Задание форм на веб-
    странице</h1>
    <form method="get" action="lab01.html">
    <table align="center" style="color:#000080; font-size:6.5mm">

```

```
<tr> <td> Текстовое поле для ввода наименования студенческой группы  
<td> <input type="text" size="15" name="txt1" value="Строка текста">
```

```
<tr> <td> Текстовое поле для ввода фамилии и инициалов студента
```

```
<td> <input type="text" size="15" name="txt1" value="Строка текста">
```

```
<tr> <td> Текстовые поля для ввода пяти чисел;
```

```
<td> <input type="text" size="15" name="txt1" value="Строка текста">
```

```
<tr><td> Выбор из списка (select)
```

```
<td><select name="sel" size="2" multiple>
```

```
<option value="first">Полный
```

```
<option selected value="second">Сжатый
```

```
</select>
```

```
<tr> <td> Переключатели
```

```
<td><input type="radio" name="rad" value="on1"> Минимум<br>
```

```
<input type="radio" name="rad" value="on2" checked>По убыванию<br>
```

```
<input type="radio" name="rad" value="on3"> Среднее <br>
```

```
</table>
```

```
<p style="margin-left:5cm; color:white; background-color:white">
```

```
<input type="reset" src ="button.png">
```

```
<input type="image" src="button.png" >
```

```
</p>
```

```
</form>
```

</body>

</html>

Робота програми на рис.3 (файл lab7)

Ссылки на веб-странице

ЛАБОРАТОРНАЯ РАБОТА №7

Тема: Работа со ссылками на веб-странице

Цель: Изучение средств и возможностей HTML и CSS для представления на веб-странице ссылок и сегментированной графики

Ссылка со стандартными параметрами:

[ссылка на ютуб](#)

Ссылка в виде рисунка:



Ссылка в виде кнопки

Кнопка

Ссылка в виде сегментированной графики:



Рисунок 3

Робота програми на рис.4 (файл lab8)

ЛАБОРАТОРНАЯ РАБОТА №8. Задание форм на веб-странице

Текстовое поле для ввода наименования студенческой группы	<input type="text" value="123"/>
Текстовое поле для ввода фамилии и инициалов студента	<input type="text" value="321"/>
Текстовые поля для ввода пяти чисел;	<input type="text" value="21312321"/>
Выбор из списка (select)	<div>Полный ^ Сжатый v</div>
Переключатели	<div><input type="radio"/> Минимум <input checked="" type="radio"/> По убыванию <input type="radio"/> Среднее</div>
<div><input type="button" value="Сбросить"/> <input type="button" value="Передача"/></div>	

Рисунок 4

Вывод

Изучили средства и возможности HTML и CSS для представления на веб-странице ссылок и сегментированной графики. Изучили средства и возможности HTML для создания форм на веб-странице.

ЛАБОРАТОРНАЯ РАБОТА №5

Тема: Основы языка JavaScript

Цель: Рассмотрение способов включения фрагментов JavaScript в HTML-документ, типов данных, методов работы с экраном и средств отладки программы

Индивидуальные задания

Разработать на JavaScript с использованием методов *alert()*, *confirm()* и *prompt()* программу тестирования знаний студентов по HTML и CSS, которая должна выполнить следующие действия:

- *обеспечить ввод студентами названия группы и своей фамилии;*
- *проверить с помощью операторов *try/catch* и *throw* правильность введенных данных;*
- *вывести на экран не менее четырех вопросов с двумя вариантами ответов каждый ("да" и "нет") по теме из таблицы 9 согласно варианту (номеру студента в журнале);*
- *определить на основании ответов студента количество полученных им баллов;*
- *вывести результаты тестирования на экран с указанием фамилии студента, его группы и полученной оценки.*

№	Тема проверки знаний
1	Форматирование текста средствами HTML
2	Форматирование текста средствами CSS
3	Вывод списков на веб-странице средствами HTML и CSS
4	Вывод таблиц на веб-странице
5	Вывод изображений на веб-странице
6	Создание фона на веб-странице
7	Вывод видеофрагментов на веб-странице
8	Использование ссылок на веб-странице
9	Использование фреймов на веб-странице
10	Использование плавающих фреймов на веб-странице
11	Создание формы на веб-странице. Поле ввода текста.
12	Создание формы на веб-странице. Зависимые переключатели.
13	Создание формы на веб-странице. Независимые переключатели.
14	Создание формы на веб-странице. Тег select.
15	Использование интерфейса CGI для обработки формы.
16	Выравнивание текста.
17	Задание размера символов.
18	Формат HTML-документа.
19	Структура URL-адреса.
20	Задание цвета на веб-странице

Код программы

```
<!DOCTYPE html>
```

```
<script type="text/javascript">
```

```
    var count=0;
```

```
        function create_counter() { // Подсчет вызовов функции
```

```
        return function() {
```

```
        return ++count;
    }
}
```

```
var user_name=prompt("Введите имя","");
```

```
if(!user_name) ;//alert("Введено имя: "+user_name);
else
{
throw alert("Имя не введено");
}
```

```
var user_group=prompt("Введите группу","");
```

```
if(!user_group) alert("Введено группы: "+user_group + "\nВведено имя: " +user_name);
//if(!user_name) alert("\nВведено имя: "+user_name);
else
{
throw alert("Группу не введено");
}
```

```
var fn=create_counter();
```

```
if(confirm("Назначение и атрибуты тега <optgroup> ?\n Ок-Элемент <optgroup> используется  
для группирования в ниспадающем списке элементов.\n Отмена-Элемент <optgroup>  
используется для группирования в форме связанных по смыслу элементов.")){
```

```
    alert("О,нет..");
```

```
}
```

```
else
```

```
{
```



```

        alert(fn());
        alert("Отлично!");
    }

//-----

    if(confirm("Атрибут wrap :\n Ок-казывает тип переноса слов при передаче формы.\n Отмена-указывает одну или несколько форм, которым принадлежит текстовая область ")){
        alert("О,нет..");
    }
else
    {
        alert(fn());
        alert("Отлично!");
    }

//-----

    if(confirm("Тег <optgroup> имеет следующие атрибуты:\n Ок-Элемент disabled и label.\n Отмена-form и name")){
        alert("Отлично!");
        alert(fn()); // 4
    }

    else alert("О,нет..");

    if(confirm("Атрибут get:\n Ок-добавляет данные формы\n Отмена-передает данные формы как HTTP post транзакция;")){
        alert("Отлично!");
        alert(fn()); // 4
    }

    else alert("О,нет..");

```

```
alert("Введено группу: "+user_group + "\nВведено имя:" +user_name +"\nВсего правильных  
ответов:" +count);  
  
//alert("Name: " + user_name +"\nВсего правильных ответов:" + count);  
</script>
```

Работа программы на рис.1 и рис.2

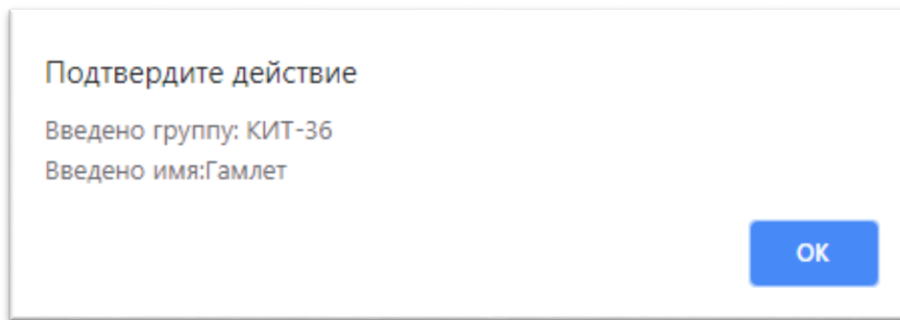


Рисунок 1

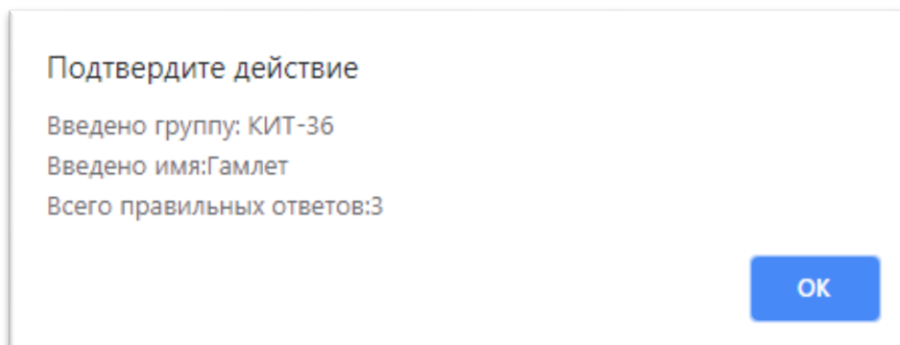


Рисунок -2

Вывод

На лабораторной работе рассмотрено способы включения фрагментов JavaScript в HTML-документ, типов данных, методов работы с экраном и средств отладки программы

ЛАБОРАТОРНАЯ РАБОТА № 6

Тема: Работа с массивами на JavaScript

Цель: Изучение способов и функций работы с индексными и ассоциативными массивами

Индивидуальные задания

На языке JavaScript разработать программные средства, выполняющие согласно указанным в таблице 1 индивидуальным параметрам, следующие действия:

- *создать индексный массив, содержащий не менее 10 элементов, из которых: 5 чисел (целых и вещественных) и 5 строк, с помощью способа, заданного в колонке 2 таблицы;*
- *над элементами созданного массива выполнить указанный в колонке 3 метод;*
- *определить тип элементов модифицированного массива и выполнить для чисел или для строк (в зависимости от задания) операцию, заданную в колонке 4;*
- *создать ассоциативный массив из не менее 5 элементов;*
- *над ключами или значениями выполнить указанную в колонке 5 операцию;*
- *после выполнения каждого из вышеперечисленных пунктов выводить содержимое массива на экран;*
- *создать многомерный ассоциативный массив и вывести на экран его значения;*
- *определить наименование и версию текущего браузера.*

№	Создание массива	Метод	Операция	
			индексный массив	ассоциативный массив
1	2	3	4	5
1	поэлементно	join()	найти максимальное число	найти ключ, первый по алфавиту
2	при объявлении	pop()	найти минимальное число	найти ключ, последний по алфавиту
3	литеральное	push()	найти среднее арифметическое чисел	найти среднюю длину ключа
4	поэлементно	reverse()	найти среднюю длину строк	найти ключ минимальной длины
5	при объявлении	shift()	найти строку, последнюю по алфавиту	найти ключ максимальной длины
6	литеральное	slice()	найти строку, максимальной длины	найти ключ, первый по алфавиту
7	поэлементно	splice()	найти строку, минимальной длины	найти ключ, последний по алфавиту
8	при объявлении	onshift()	найти максимальное число	найти среднюю длину ключа
9	литеральное	join()	найти минимальное число	найти ключ максимальной длины
10	поэлементно	pop()	найти среднее арифметическое чисел	найти ключ минимальной длины
11	при объявлении	push()	найти среднюю длину строк	найти ключ, первый по алфавиту
12	литеральное	reverse()	найти строку, последнюю по алфавиту	найти среднюю длину ключа
13	поэлементно	shift()	найти строку, максимальной длины	найти ключ, последний по алфавиту
14	при объявлении	slice()	найти строку, минимальной длины	найти ключ, первый по алфавиту
15	литеральное	splice()	найти максимальное число	найти ключ, последний по алфавиту
16	поэлементно	onshift()	найти минимальное число	найти среднюю длину ключа
17	при объявлении	join()	найти среднее арифметическое чисел	найти ключ, первый по алфавиту
18	литеральное	pop()	найти среднюю длину строк	найти ключ, последний по алфавиту
19	поэлементно	push()	найти строку, последнюю по алфавиту	найти ключ максимальной длины
20	при объявлении	reverse()	найти строку, максимальной длины	найти ключ, последний по алфавиту

Код программы

```
<!DOCTYPE html>
```

```
<script type="text/javascript">
```

```
arr=[1,2,2.8e-1,4,5,"Если у тебя есть сад и библиотека, то у тебя есть все, что тебе нужно.",
"А Нет ничего более изобретательного, чем природа.",
"Оживи речь юмором.",
"Дом, в котором нет книги, подобен телу, лишённому души.",
"А Рукописи не горят."];
```

```
a="";
```

```
function min(arr1){
```

```
(arr1.sort())
```

```
_min = arr1[5];
```

```
return _min;
```

```
}
```

```
alert(arr.reverse());
```

```
alert(typeof arr);
```

```
alert("Min:" +min(arr));
```

```
Arr={}
```

```
Arr['color_one'] = 'red';
```

```
Arr['color_two'] = 'blue';
```

```
Arr['color_three'] = 'black';
```

```
Arr['color_four'] = 'green';
```

```

Arr['color_five'] = 'white';
count_length=0;
for( var ArrVal in Arr ) {
//alert(Arr[ArrVal].length);
count_length+= Arr[ArrVal].length;
}
countavg = count_length/Arr[ArrVal].length;
alert(countavg);
var smartphone={ 'Модель': { 'фирма':'Xiaomi','Версия андроид':'7.0 NRD90M' },
'Память': { 'Память ОЗУ':'3ГБ','Постоянная':'32 ГБ' } };
a="Модель:\n";
for( i in smartphone['Модель']) a+=i+' - '+smartphone['Модель'][i]+"\n";
a+='Память: '+smartphone['Память']['Тип'];
alert(a);
alert(navigator.userAgent)
</script>

```

Работа программы

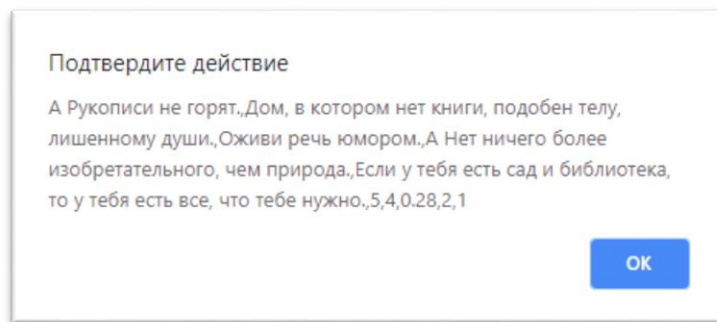


Рисунок 1

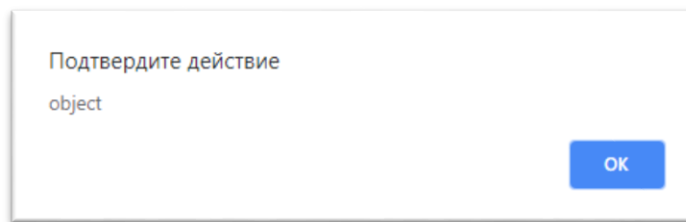


Рисунок 1.2

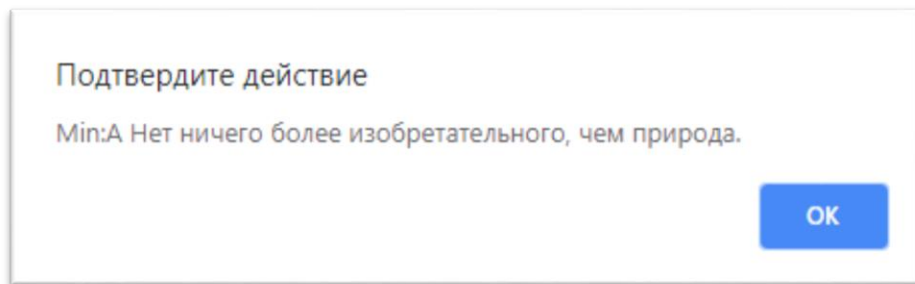


Рисунок 1.3

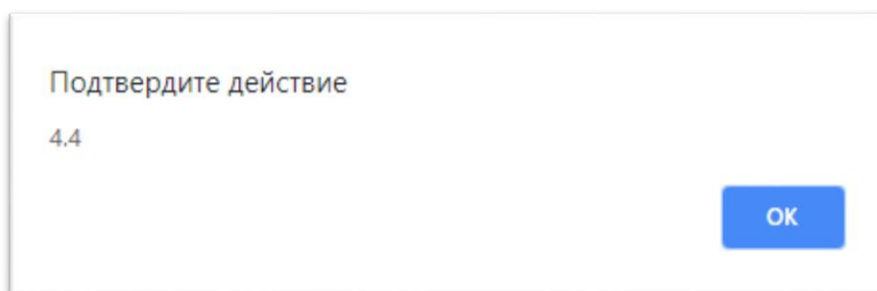


Рисунок 1.4

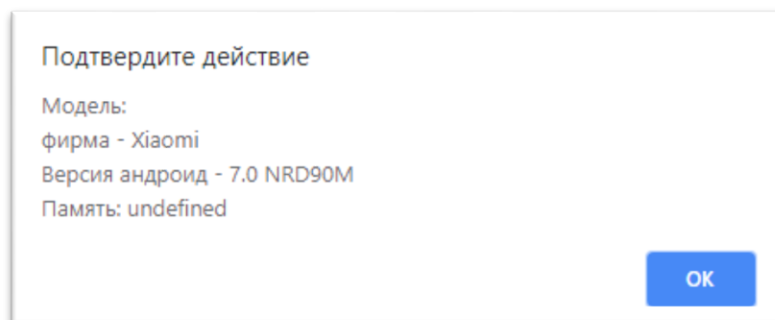


Рисунок 1.5

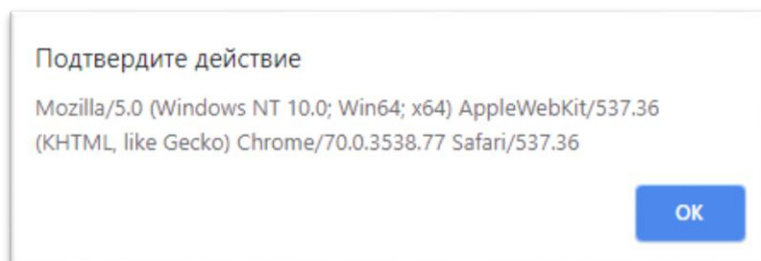


Рисунок 1.6

Вывод

На лабораторной работе изучили способов и функций работы с индексными и ассоциативными массивами.

Индивидуальные задания

На языке JavaScript разработать программные средства, выполняющие согласно указанным в таблице 1 индивидуальным параметрам, следующие действия:

- по событию 1 с помощью указанного способа доступа к тегу изменить его свойства (свойства должны быть указаны: а) в виде HTML-атрибута; б) с помощью атрибута style; в) с помощью тега <style>);
- по событию 2 восстановить исходные свойства тега. При этом использовать альтернативный способ доступа к тегу (если при изменении свойств применялся доступ по отметке, то теперь – по идентификатору, и наоборот);
- по событию 3 с помощью коллекции изменить одно свойство всех имеющихся на веб-странице тегов (не менее трех) указанного наименования;
- обработчики событий зарегистрировать, используя:
 - для события 1 – встроенную модель;
 - для события 2 – традиционную модель;
 - для события 3 – усовершенствованную модель;
- разработать обработчик события error и проверить его работу.

Код программы

```
<!DOCTYPE HTML>

<html>

<head>

<meta charset="utf-8">

<title>Oncontextmenu</title>

<style>

div {

height: 200px; /* Высота слоя */

background: #f0f0f0; /* */

padding: 5px; /* Поля вокруг текста */

border: 1px solid #333; /* Параметры рамки */

}

</style>

<ol id="myOl">

<li>One</li>
```



```
<li>Two</li>
<li>Three</li>
</ol>
<button onclick="myFunction()">Try it</button>
```

```
<script >
```

```
function myFunction() {
    var x = document.getElementById("myOl");
    x.start = "314";
}
```

```
var color = 0;
function colorDiv() {
    if (color) {
        document.getElementById("layer").style.background = "#f0f0f0";
        color = 0;
    }
    else {
```

layer.onmouseout = restoreTag;//onmouseout – возникает при снятии указателя мышки с элемента или с одного из его дочерних элементов;

```
    }
}

function restoreTag() { //Восстановление свойств элементов Web-страницы
document.getElementById("layer").style.background = "blue";
    color = 1;
```

```

}

</script>

</head>

<body>

<p>Click right click.</p>

<div id="layer" oncontextmenu="colorDiv()">

<a href="https://www.nasa.gov/"></a>

<p id="myOl" style="color:blue; font-size:8mm; position:absolute; left:50px;
top:100px">TEXT</p>

</div>

</body>

</html>

```

Работа программы

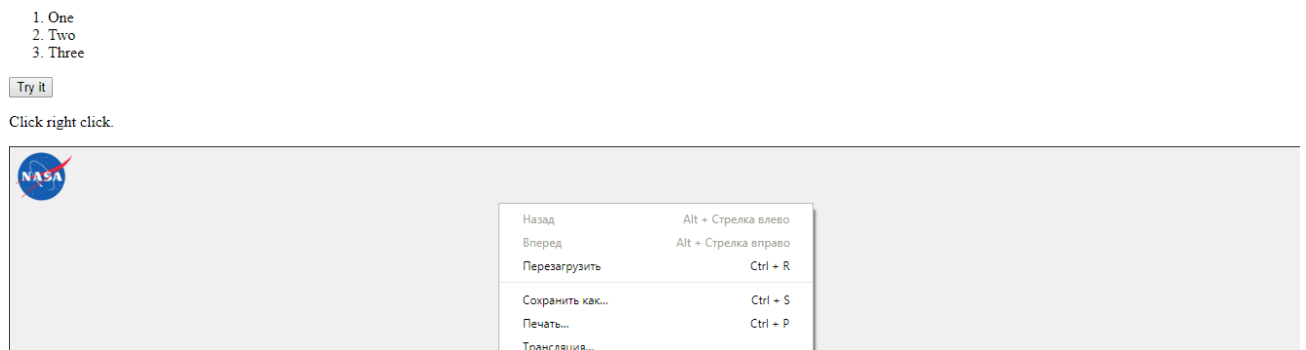


Рисунок 1



Рисунок 2

314. One
315. Two
316. Three

[Try it](#)

Click right click.



Рисунок 3

Вывод

Изучили средства и возможности языка JavaScript для обработки событий на веб-странице с целью определения и изменения свойств ее элементов

ЛАБОРАТОРНАЯ РАБОТА №12

Тема: Движение элементов на веб-странице

Цель: Изучение средств и возможностей языка JavaScript для организации движения элементов на веб-странице

Индивидуальные задания

- На языке JavaScript разработать программные средства, выполняющие следующие действия:
 - по событию 1 (табл. 1) отметить указанный в таблице объект, изменив одно или несколько его свойств;
 - по событию 2 осуществить движение этого объекта из заданной точки (x,y) экрана по траектории, заданной функцией $f(x)$ и направлением ("куда") в соответствии с указанным режимом:
 - 1 – до достижения границ окна браузера;
 - 2 – до достижения границ окна браузера после однократного отражения;
 - 2 – непрерывное движение с многократными отражениями от границ окна браузера;
 - амплитуду и скорость движения выбирать из соображения наглядности. Обратить внимание на выбор диапазона значений аргумента функции (переменной x);
 - выполненное движение элемента в окне браузера отобразить в виде графика движения (см. пример №3).

Код программы<!DOCTYPE html>

```
<html>

  <head>

    <meta charset="utf-8">

    <title></title>

    <style>

      #myli{

        position:absolute;

      }

      .dot {

        position:absolute;

        width:7px;

        height:7px;

        background-color:#0000bf;

        z-index:99;
```

```

        }
    </style>
</head>
<body>

    <ul id="myli" oncontextmenu="ev1();" onmouseover="ev2();">
        <li>Click me</li>

    </ul>

    <script>
isMoving=false;
mode = 1;
framecount=0;
xpos=0;
firstMove=true;
function f(x){
    return (Math.log(50)*(x+5)-50) ;
}
frameproc = function(){
    ++framecount;
    xpos += f(framecount)*mode;
    if(mode == 1){
        if(xpos < window.innerWidth-myli.clientWidth){
            setTimeout(frameproc,16);
        }
    }
    else{
        xpos = window.innerWidth-myli.clientWidth;
        mode = -1;
        setTimeout(frameproc,16);
    }
}
    </script>

```

```

        }
    }
    else{
        if(xpos >0){
            setTimeout(frameproc,16);
        }
        else{
            xpos = 0
            isMoving=false;
            firstMove=false;
        }
    }
    //myli.style.left = xpos+"px";
    if(firstMove){
        var dv=document.createElement("DIV");
        dv.className="dot";
        var xpos2 = xpos/2 + window.innerWidth/2
        dv.style.left=xpos2+"px";
        dv.style.top=framecount+"px";
        document.body.appendChild(dv);
    }

}

ev1=function(){
    myli.style.backgroundColor="yellow";

}

ev2=function(){
    if(!isMoving){
        xpos = 75;
    }
}

```

```

        myli.style.top = 175+"px";

        framecount=0;

        isMoving=true;

        mode=1;

        setTimeout(frameproc,16);

    }

}

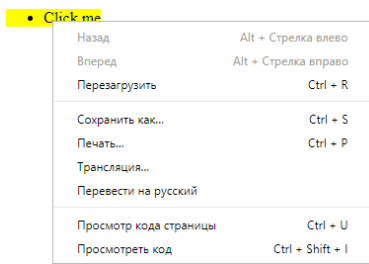
</script>

</body>

</html>

```

Работа программы



Вывод

Изучили средства и возможности языка JavaScript для организации движения элементов на веб-странице

ЛАБОРАТОРНАЯ РАБОТА №13

Тема: Скрытие элементов веб-страницы. Работа с клавиатурой

Цель: Изучение средств и возможностей языка JavaScript для скрытия элементов веб-страницы и обработки событий, связанных с работой пользователя на клавиатуре

Индивидуальные задания

- Разработать веб-страницу, содержащую трехслойную структуру, каждый слой которой состоит из тегов, указанных в колонке "Теги слоев" таблицы 1.
- По событиям 1, 2, 3 и 4, заключающимся в одновременном нажатии двух клавиш (первая указана в колонке "Буква", вторая – одна из клавиш "1", "2", "3" и "4"), выполнить следующие действия:
 - по событию 1 показать только первый слой;
 - по событию 2 показать только второй слой;
 - по событию 3 показать только третий слой;
 - по событию 4 показать одновременно два слоя, номера которых указаны в колонке "Слой".
- Используя свойство CSS display:
 - по событию 5 – поместить в окно браузера комментарий или пояснение, касающееся одного из элементов веб-страницы;
 - по событию 6 – удалить выведенные комментарий или пояснение.

Примечание: если событие 6 совпадает с событием 5, это означает, что оно заключается в повторном выполнении события 5.
- По нажатию клавиши "F1" вывести на экран подсказку (help) о назначении используемых клавиш.

Таблица 1 – Перечень заданий

№	Теги слоев			Слой	Буква	События	
	1	2	3			5	6
12	P, A	IMG	IFRAME	1, 2	I	Alt	Shift

Код программы <!DOCTYPE html><html>

<html>

<head>


```
<style>
```

```
#div1 {
```

```
width: 100%;
```

```
padding: 50px 0;
```

```
text-align: center;
```

```
background-color: lightblue;
```

```
margin-top: 20px;
```

```
p {
```

```
color: red;
```

```
font-family: courier;
```

```
font-size: 160%;
```

```
}
```

```
}
```

```
</style>
```

```
<p id="demo" style="color:black; font-size:11.0mm;margin-left:1% position:absolute ;"></p>
```

```
<a id="myLink"style="color:black; font-size:11.0mm;margin-left:1% position:absolute ; title="Click  
to do something"
```

```
href="PleaseEnableJavascript.html" onclick="MyFunction();return false;"></a>
```

```
<a id="iframeCover" class="om-trigger-conversion"></a>
```

```
<!-- <div id="formInstructions">Double Click Below to Get Started</div> -->
```

```
<iframe id="fred" hidden = "none" width="200" height="200" ></iframe>
```

<p id="p1" style="text-align:center " ;></p>

<script>

var left_, top_, width, height, img_;

window.onkeydown=function(e) { // Обработчик события keydown

if('onhelp' in window) {

window.onhelp=function() { // Обработчик события help для браузера IE

help();

return false;

}

}

else if(e.which==112) {

help();

return false;

}

<!-- if(event.altKey) -->

<!-- document.getElementById("p1").innerHTML = "New text!"; -->

<!-- if(event.shiftKey) -->

<!-- document.getElementById("p1").hidden = true; -->

if((e.shiftKey && e.which==50)||(e.shiftKey && e.which==52)) {

left_=500, top_=500, width=1280, height=960, img_;

```

img_=document.createElement("IMG");
document.body.appendChild(img_);
img_.style.position="absolute";
img_.style.left=left_+"px";
img_.style.top=top_+"px";
img_.style.width=width+"px";
img_.style.height=height+"px";
img_.style.zIndex="-1";

}

switch (e.which) { // Анализ Unicode клавиш
case 37: { // "Трелка влево"
if(left_>10) {
left_-=10;
img_.style.left=left_+"px";
};
break;
}
case 39: { // Стрелка вправо"
if(left_<window.innerWidth-170) {
left_+=10;
img_.style.left=left_+"px";
};
break;
}
case 38: { // "Стрелка вверх"
if(top_>10) {
top_-=10;
img_.style.top=top_+"px";
}
}

```

```
break;
}
case 40: { // "Стрелка вниз"
if(top_<window.innerHeight-170) {
top_+=10;
img_.style.top=top_+"px";
} break;
}
case e.shiftKey && 49: { // "1"
document.getElementById("demo").innerHTML =
"Hello Dolly!";
document.getElementById("myLink").innerHTML="Link";
break;
}
case e.shiftKey && 50: { // "2"

img_.src="begin.jpg";
break;
}
case e.shiftKey && 51: { // "3"
document.getElementById("fred").style.display = "block"; //shows the frame

break;
}
case e.shiftKey && 52: { // "4"
document.getElementById("demo").innerHTML =
"Hello Dolly!";
document.getElementById("myLink").innerHTML="Link";
img_.src="begin.jpg";
break;
```

```
}
```

```
case 18:{
```

```
document.getElementById("p1").innerHTML = "block – элемент показывается как блочный, занимает всю доступную ширину окна и перед которым и после которого браузер устанавливает обрывы строк (переносы строк), т.е. строки, завершаются символами новой строки. Для блочного элемента можно явно указать ширину и высоту. Блочными элементами являются, например, заголовки и параграфы веб-страницы;!";
```

```
var x = document.getElementById("p1");
```

```
if (e.shiftKey) {
```

```
x.innerHTML = "";
```

```
} else {
```

```
x.innerHTML = "block – элемент показывается как блочный, занимает всю доступную ширину окна и перед которым и после которого браузер устанавливает обрывы строк (переносы строк), т.е. строки, завершаются символами новой строки. Для блочного элемента можно явно указать ширину и высоту. Блочными элементами являются, например, заголовки и параграфы веб-страницы;";
```

```
}
```

```
// document.getElementById("p1").hidden = true;
```

```
//document.getElementById("p1").style.visibility = "hidden";
```

```
break;
```

```
}
```

```
case 188: { // "<"
```

```
if( width>20) {
```

```
width-=1;
```

```
img_.style.width=width+"px";
```

```
height-=1;
```

```
img_.style.height=height+"px";
```

```
}
```

```

break;
}
case 190: { // ">"
width+=1;
img_.style.width=width+"px";
height+=1;
img_.style.height=height+"px";
break;
}
}
}
}

```

```

function help() {
alert("Выбор рисунка - клавиши '1','2' и '3'\n" +
"Перемещение рисунка - клавиши 'Стрелка влево','Стрелка вправо','+
'Стрелка вниз' и 'Стрелка вверх'\n"+
"Изменение размера рисунка: - клавиши '<' и '>'");
}

```

```

</script>

```

```

</head>

```

```

<body>

```

```

<span style="color:blue; font-size:5.0mm;margin:1%; position:absolute ; outline-style: outset;">

```

```

Comments for block press:ALT</span><br>

```

```

<p style = "color:black;margin-right:1%;font-size:20mm; position:absolute "></p>

```

```

<h1 style="color:#0000c0;margin-left:5%;font-size:7mm; position:absolute ; z-index:-2">

```

```

</h1>

```

```

</body>

```

```

</html>

```

Работа программы

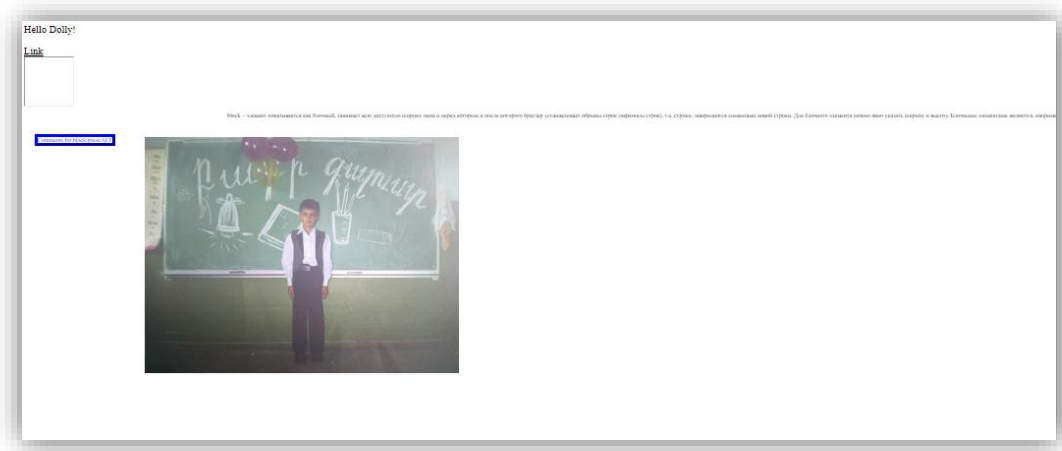


Рисунок 1 – работа программы

Вывод

Изучили средства и возможности языка JavaScript для скрывания элементов веб-страницы и обработки событий, связанных с работой пользователя на клавиатуре