# Лабораторная работа №3

Инструкционно-технологическая карта

Тема: «Создание форм на HTML-странице и, изменение свойств элементов формы»

Цель: ознакомиться с созданием форм в HTML, с элементами, которые могут находиться в форме и их добавлением.

Время выполнения: 2 часа

СОДЕРЖАНИЕ РАБОТЫ

1. Контрольные вопросы.
2. Теоретические сведения для выполнения работы.
3. Порядок выполнения работы.
4. Домашнее задание.
5. Литература.
6. КОНТРОЛЬНЫЕ ВОПРОСЫ
7. Для чего предназначены формы?
8. Каким тегом определяются формы?
9. Напишите формат контейнера <FORM>.
10. Какие атрибуты имеет тег <FORM>?
11. Для чего предназначен атрибут Action?
12. Для чего предназначен атрибут Method?
13. Какие значения может принимать атрибут Method? В каком случае используется то или иное значение?
14. Для чего предназначен атрибут Enctype?
15. Какие теги применяются для создания полей для ввода данных внутри формы?
16. Каково назначение тега <INPUT>?
17. Перечислите атрибуты тега <INPUT>. Каково их назначение?
18. Какие значения может принимать атрибут Type тега <INPUT>?
19. Какой элемент будет добавлен в форму, если атрибут Type тега <INPUT> имеет значение Text?
20. ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ ДЛЯ ВЫПОЛНЕНИЯ РАБОТЫ

***Элемент <FORM>***

**HTML-формы** являются элементами управления, которые применяются для сбора информации от посетителей веб-сайта.

Веб-формы состоят из набора текстовых полей, кнопок, списков и других элементов управления, которые активизируются щелчком мыши. Технически формы передают данные от пользователя удаленному серверу.

Для получения и обработки данных форм используются языки веб-программирования, такие как PHP, Perl.

Определяются формы тэгом <**FORM**>. В одном документе может быть определено несколько форм для заполнения, но они не должны быть вложенными одна в другую.

Формат контейнера <**FORM**> следующий:

<FORM action="url" method="POST">...</FORM>

Тэг <**FORM**> имеет атрибуты, перечисленные в таблице 3.1.

Таблица 3.1 – Атрибуты тега <**FORM**>.

|  |  |
| --- | --- |
| Атрибут | Назначение |
| **Name** | имя формы |
| **action="..."** | Обязательный атрибут, который указывает URL обработчика формы на сервере, которому передаются данные. Представляет из себя файл (например, action.php), в котором описано, что нужно делать с данными формы. Если значение атрибута не будет указано, то после перезагрузки страницы элементы формы примут значения по умолчанию.  В случае, если вся работа будет выполняться на стороне клиента сценариями JavaScript, то для атрибута action можно указать значение #.  Также можно сделать так, чтобы заполненная посетителем форма приходила вам на почту. Для этого нужно внести следующую запись: <form action="mailto:адрес вашей электронной почты" enctype="text/plain"></form> |
| **method="..."** | определяет метод, используемый для посылки содержания заполненной формы на сервер. Возможные варианты при этом следующие:  Метод **GET** передает данные на сервер через адресную строку браузера. При формировании запроса к серверу все переменные и их значения формируют последовательность вида www.anysite.ru/form.php?var1=1&var2=2. Имена и значения переменных присоединяются к адресу сервера после знака ? и разделяются между собой знаком &. Все специальные символы и буквы, отличные от латинских, кодируются в формате %nn, пробел заменяется на +. Этот метод нужно использовать, если вы не передаете больших объемов информации. Если вместе с формой предполагается отправка какого-либо файла, этот метод не подойдет.  Метод **POST** применяется для пересылки данных больших объемов, а также конфиденциальной информации и паролей. Данные, отправляемые с помощью этого метода, не видны в заголовке URL, так как они содержатся в теле сообщения. <form action="action.php" enctype="multipart/form-data" method="post"></form> |
| **enctype="..."** | задает тип кодирования содержимого заполненной формы. Если значение атрибута не установлено, по умолчанию предполагается application/x-www-form-urlencoded. |
| **accept-charset** | Значение атрибута представляет собой разделенный пробелами **список кодировок символов**, которые будут использоваться для отправки формы, например, <form accept-charset="ISO-8859-1">. |
| **target** | Указывает окно, в которое будет направлена информация: \_blank — новое окно, \_self — тот же фрейм, \_parent — родительский фрейм (если он существует, если нет — то в текущий), \_top — окно верхнего уровня по отношению к данному фрейму. Если вызов происходит не из дочернего фрейма, то в тот же фрейм. |

Обработку данных, вводимых в формы, осуществляет программа, адрес которой указывается в атрибуте **action="..."** тэга <**FORM**>. Но далеко не все сервера, предоставляющее место для личных сайтов, разрешают работу таким программам. Поэтому для демонстрации примеров использования форм будет отправлять содержимое формы будет отправляться не на сервер а по электронной почте адресату, которого вы сами укажите. Для этого в атрибуте action="..." прописывается макрокоманда отправки почтового сообщения:

<FORM ACTION="mailto:freeuser@mail.net"> ,

где freeuser@mail.net ­ адрес электронной почты получателя. Для отправки содержимого формы в данном случае используется почтовая программа отправителя.

Внутри контейнера <**FORM**> могут находиться любые теги, кроме другого контейнера <**FORM**>. Для создания полей для ввода данных внутри формы применяют теги:

* <**INPUT**>,
* <**SELECT**>,
* <**TEXTAREA**>.

***Создание полей формы***

Тег <**INPUT**> используется для задания простого элемента ввода. Атрибуты элемента отличаются в зависимости от типа поля, для создания которого используется этот элемент. Могут быть использованы следующие атрибуты:

**accept** определяет тип файла, разрешенных для отправки на сервер. Указывается только для <input type="file">. Возможные значения:

file\_extension — разрешает загрузку файлов с указанным расширением, например, accept=".gif", accept=".pdf", accept=".doc"

audio/\* ­ разрешает загрузку аудиофайлов

video/\* ­ разрешает загрузку видеофайлов

image/\* ­ разрешает загрузку изображений

media\_type ­ указывает на медиа-тип загружаемых файлов.

**Alt** определяет **альтернативный текст** для изображений, указывается только для <input type="image">.

**Checked** проверяет, установлен ли флажок по умолчанию при загрузке страницы для полей типа type="checkbox" и type="radio".

**Disabled** отключает возможность редактирования и копирования содержимого поля.

**Maxlength="n"** задает максимальное количество символов, вводимых в поле. Значение по умолчанию 524288 символов.

**Name= "идентификатор"** - имя для создаваемого поля ввода, используемое для идентификации при обработке данных сервером.

**Readonly** не позволяет пользователю изменять значения элементов формы, выделение и копирование текста при этом доступно. Указывается без значения атрибута.

**Size="n"** задает физический размер поля ввода в символах, действует для элементов ввода текста или пароля.

**Value="значение"** - для полей ввода текста **text** или пароля **password** используется для задания начального содержания поля, а для **checkbox** или **radio** задает значение, когда элемент находится в отмеченном состоянии.

**Src** задает URL изображения, используемого в качестве кнопки отправки данных формы. Указывается только для поля <input type="image">.

**type= "параметр"** задает тип создаваемого элемента ввода; этот параметр может принимать одно из значений, указанных в таблице 3.2.

Таблица 3.2 – Возможные значения атрибута **type** и их применение

|  |  |  |
| --- | --- | --- |
| **Значение атрибута type** | **Назначение** | **Пример** |
| **text** | создает поле для ввода текста. Это значение используется по умолчанию. | <form method= "get" action="/cgi-bin/handler.cgi">  <b>Зaполните Ваши данные:</b>  Имя: <input type=text maxlength=25 size=20>  Фамилия:<input type=text maxlength=25 size=20>  </form> |
| **password** | создает поле ввода пароля, аналогичное типу "text", но при этом вводимые символы отображаются на экране звездочками;  <input type=password параметры> | <form method= "get" action="/cgi-bin/handler.cgi">  <b>Логин: </b><input type=text maxlength=25 size=20 name="text">  <b>Пароль:</b> <input type=password maxlength=15 size=20 name="pass">  </form> |
| **file** | создает поле для ввода имени локального файла, который необходимо куда-то послать. Сопровождается созданием кнопки "Обзор": |  |
| **hidden** | создает скрытый элемент не показываемый пользователю. Информация, хранящаяся в скрытом поле, всегда пересылается на сервер и не может быть изменена ни пользователем, ни браузером. Может применятся, например, при обработке нескольких форм, поочередно отправляемых пользователем и каким-либо образом связанных между собой; |  |
| **radio** | создает радиокнопку - элемент-переключатель в составе группы, принимающий положения "включено" и "выключено", из которых может быть выбран только один из элементов; | <form action="/cgi-bin/handler.cgi"> <b>Какое время года Вы больше любите?)</b>  <input type=radio name=answer value=a1>Весну  <input type=radio name=answer value=a2>Лето  <input type=radio name=answer value=a3>Осень  <input type=radio name=answer value=a4>Зиму  </form> |
| **checkbox** | создает переключатель-поле для установки флажка "включено"/"выключено"; | <form action="/cgi-bin/handler.cgi">  <b>С какими операционными системамы вы знакомы?</b>  <input type=checkbox name=option1 value=a1 checked>Windows 95/98  <input type=checkbox name=option2 value=a2>Windows 2000  <input type=checkbox name=option3 value=a3>Linux  </form> |
| **submit** | создает кнопку передачи данных, действие которой сводится к отсылке содержимого заполненной формы на сервер; | <form action="/cgi-bin/handler.cgi">  <p><input type=submit value="Отправить данные"> |
| **image** | определяется элемент в виде графического изображения, действующий аналогично кнопке "submit" | <form action="/cgi-bin/handler.cgi">  <table align=center>  <tr><td colspan=2>Введите ваше имя: <tr>  <td><input type=text width=25></td>  <td><input type=image src=../school/examples3/ser.gif width=90 height=68 border=0></td>  </tr>  </table>  </form> |
| **reset** | создает кнопку, которая сбрасывает значения введенных полей в исходное состояние. | <form action="/cgi-bin/handler.cgi">  <input type=text width=20> <input type=reset value="Сбросить данные">  </form> |

***Раскрывающийся список***

Раскрывающиеся списки создаются при помощи элемента <**SELECT**>...</**SELECT**>. Списки дают возможность расположить большое количество пунктов компактно. Они позволяют выбрать одно или несколько значений из предложенного множества. По умолчанию в поле списка отображается его первый элемент.

Атрибуты тега <**SELECT**> следующие:

**Autofocus** устанавливает автоматический фокус на элементе при загрузке страницы.

**Disabled** отключает раскрывающийся список.

**Form** определяет форму, которой принадлежит данный список. В качестве значения атрибута указывается идентификатор формы.

**Multiple** дает возможность выбора одного или нескольких пунктов, для этого при выборе нужно нажать и удерживать нажатой клавишу Ctrl.

**Name** определяет имя для выпадающего списка. Значение атрибута содержит название, отражающее тематику списка.

**Required** выводит сообщение о том, что пользователь должен выбрать значение из раскрывающегося списка перед отправкой формы.

**Size** задает количество одновременно видимых на экране элементов списка. Если количество элементов списка превышает установленное количество, появляется полоса прокрутки. Значение атрибута задается целым положительным числом.

Для добавления в список пунктов используются элементы <**OPTION**>...</**OPTION**>, которые располагаются внутри <**SELECT**>.

Атрибуты тега <**OPTION**>:

**Disabled** делает недоступным для выбора элемент списка.

**Label** задает укороченную версию для элемента, которая будет отражаться в выпадающем списке. Значение атрибута содержит текст, описывающий соответствующий пункт выпадающего списка.

**Selected** отображает выбранный элемент списка по умолчанию при загрузке страницы браузером.

**Value** указывает значение, которое будет отправлено на сервер при отправке формы.

Для систематизации списков применяется элемент <**OPTGROUP**>...</**OPTGROUP**>, который создает заголовки в списках.

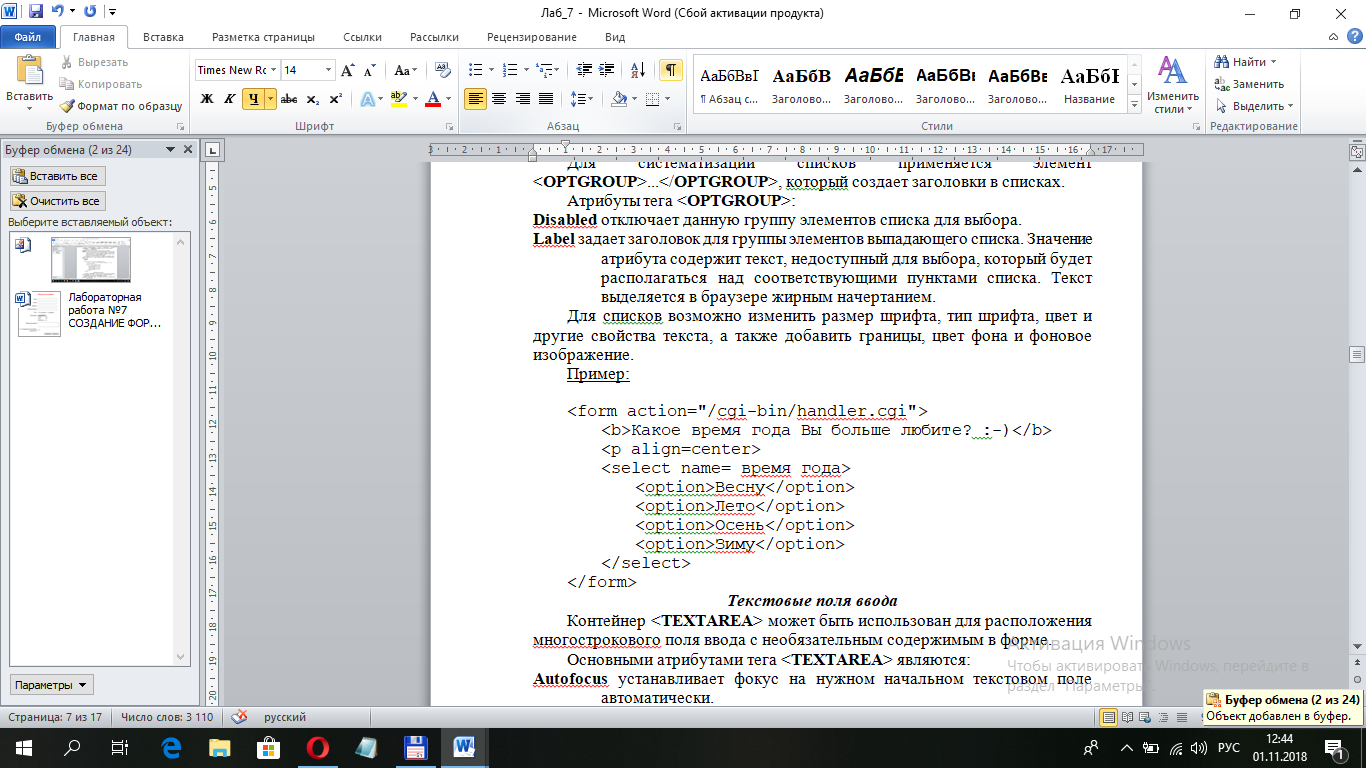
Атрибуты тега <**OPTGROUP**>:

**Disabled** отключает данную группу элементов списка для выбора.

**Label** задает заголовок для группы элементов выпадающего списка. Значение атрибута содержит текст, недоступный для выбора, который будет располагаться над соответствующими пунктами списка. Текст выделяется в браузере жирным начертанием.

Для списков возможно изменить размер шрифта, тип шрифта, цвет и другие свойства текста, а также добавить границы, цвет фона и фоновое изображение.

Пример:



***Текстовые поля ввода***

Контейнер <**TEXTAREA**> может быть использован для расположения многострокового поля ввода с необязательным содержимым в форме.

Основными атрибутами тега <**TEXTAREA**> являются:

**Autofocus** устанавливает фокус на нужном начальном текстовом поле автоматически.

**Cols** устанавливает ширину через количество символов. Если пользователь вводит больше текста, появляется полоса прокрутки.

**Disabled** отключает возможность редактирования и копирования содержимого поля.

**Form** - значение атрибута должно быть равно значению атрибута id элемента <form> в этом же документе. Определяет одну или несколько форм, которым принадлежит данное текстовое поле.

**Maxlength** - значение атрибута задает максимальное число символов для ввода в поле.

**Name** задает имя текстового поля.

**Placeholder** определяет короткую текстовую подсказку, которая описывает ожидаемое вводимое значение.

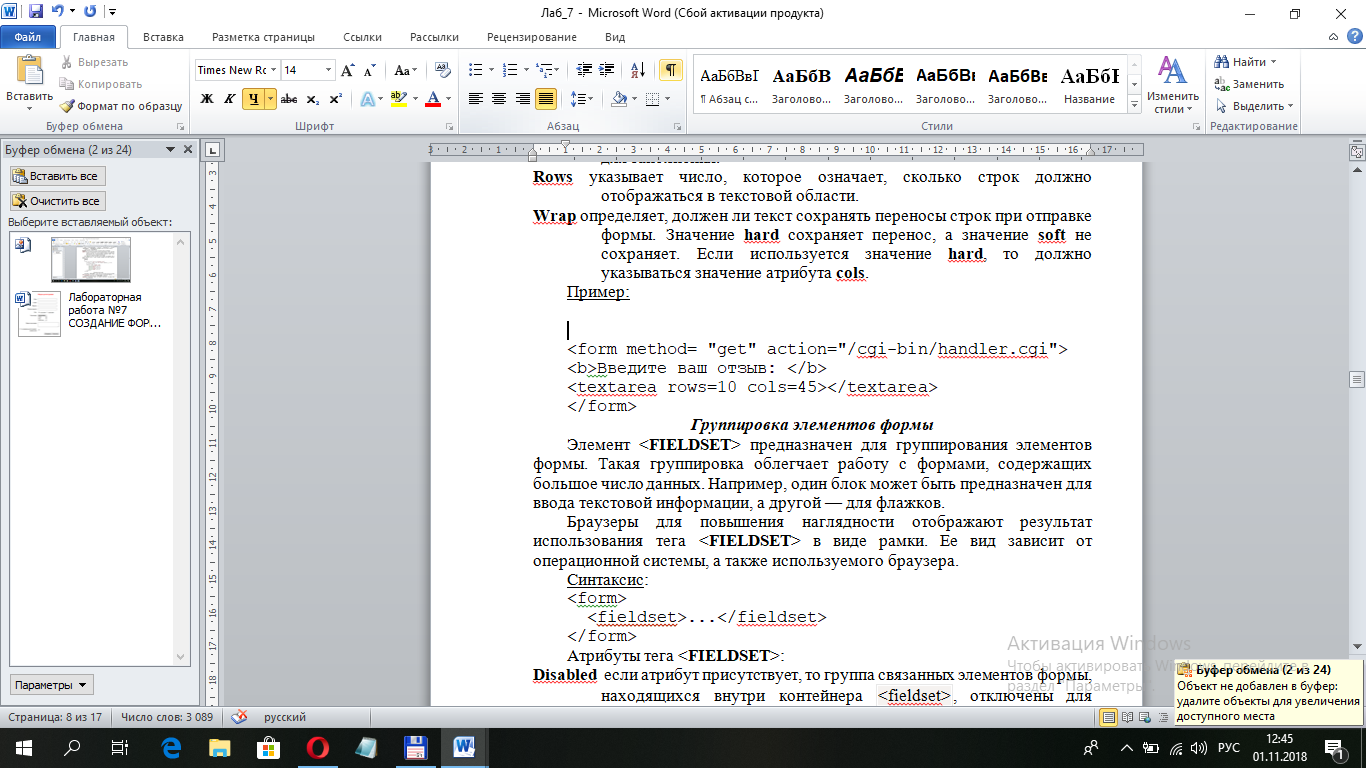
**Readonly** отключает возможность редактирования содержимого поля.

**Required** выводит сообщение о том, что данное поле является обязательным для заполнения.

**Rows** указывает число, которое означает, сколько строк должно отображаться в текстовой области.

**Wrap** определяет, должен ли текст сохранять переносы строк при отправке формы. Значение **hard** сохраняет перенос, а значение **soft** не сохраняет. Если используется значение **hard**, то должно указываться значение атрибута **cols**.

Пример:



***Группировка элементов формы***

Элемент <**FIELDSET**> предназначен для группирования элементов формы. Такая группировка облегчает работу с формами, содержащих большое число данных. Например, один блок может быть предназначен для ввода текстовой информации, а другой — для флажков.

Браузеры для повышения наглядности отображают результат использования тега <**FIELDSET**> в виде рамки. Ее вид зависит от операционной системы, а также используемого браузера.

Синтаксис:

<form>

<fieldset>...</fieldset>

</form>

Атрибуты тега <**FIELDSET**>:

**Disabled** ­ если атрибут присутствует, то группа связанных элементов формы, находящихся внутри контейнера <fieldset>, отключены для заполнения и редактирования. Используется для ограничения доступа к некоторым полям формы, содержащих ранее введенные данные. Атрибут используется без указания значения — <fieldset disabled>..

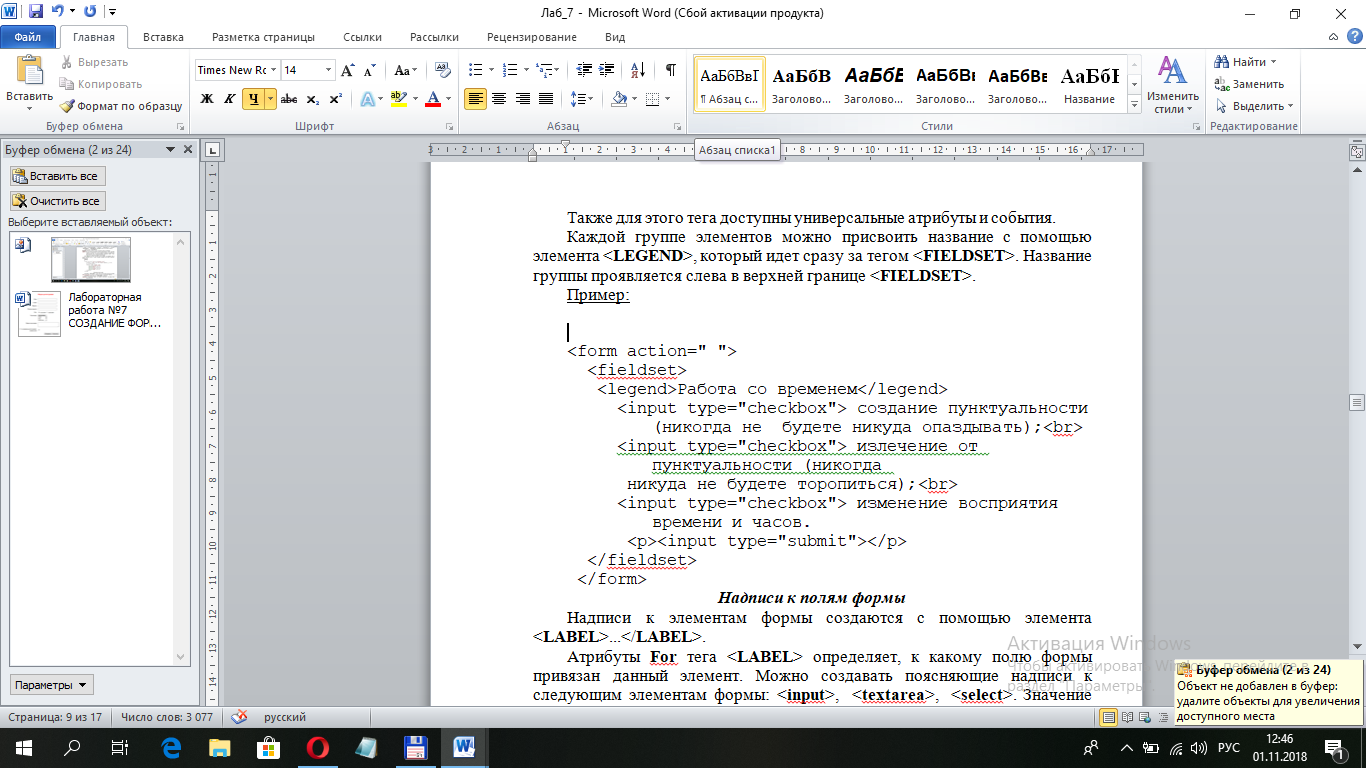
**Form** связывает группу с формой. Значение атрибута должно быть равно атрибуту id элемента <form> в этом же документе. Указывает на одну или несколько форм, к которым принадлежит данная группа элементов. На данный момент атрибут не поддерживается ни одним браузером.

**Name** определяет имя, которое будет использоваться для ссылки на элементы в JavaScript, или для ссылки на данные формы после заполнения и отправки формы. Является аналогом атрибута id.

Также для этого тега доступны универсальные атрибуты и события.

Каждой группе элементов можно присвоить название с помощью элемента <**LEGEND**>, который идет сразу за тегом <**FIELDSET**>. Название группы проявляется слева в верхней границе <**FIELDSET**>.

Пример:

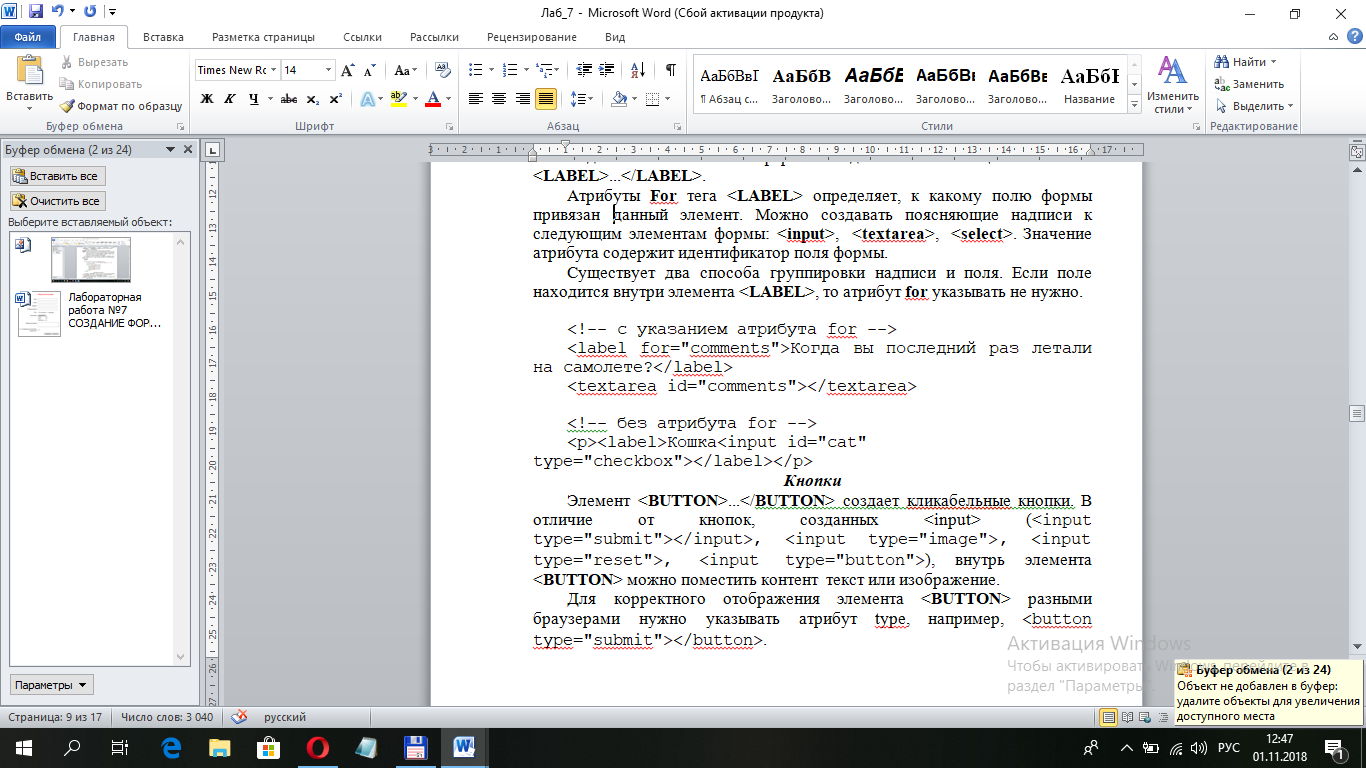


***Надписи к полям формы***

Надписи к элементам формы создаются с помощью элемента <**LABEL**>...</**LABEL**>.

Атрибуты **For** тега <**LABEL**> определяет, к какому полю формы привязан данный элемент. Можно создавать поясняющие надписи к следующим элементам формы: <**input**>, <**textarea**>, <**select**>. Значение атрибута содержит идентификатор поля формы.

Существует два способа группировки надписи и поля. Если поле находится внутри элемента <**LABEL**>, то атрибут **for** указывать не нужно.



***Кнопки***

Элемент <**BUTTON**>...</**BUTTON**> создает кликабельные кнопки. В отличие от кнопок, созданных <input> (<input type="submit"></input>, <input type="image">, <input type="reset">, <input type="button">), внутрь элемента <**BUTTON**> можно поместить контент ­ текст или изображение.

Для корректного отображения элемента <**BUTTON**> разными браузерами нужно указывать атрибут type, например, <button type="submit"></button>.

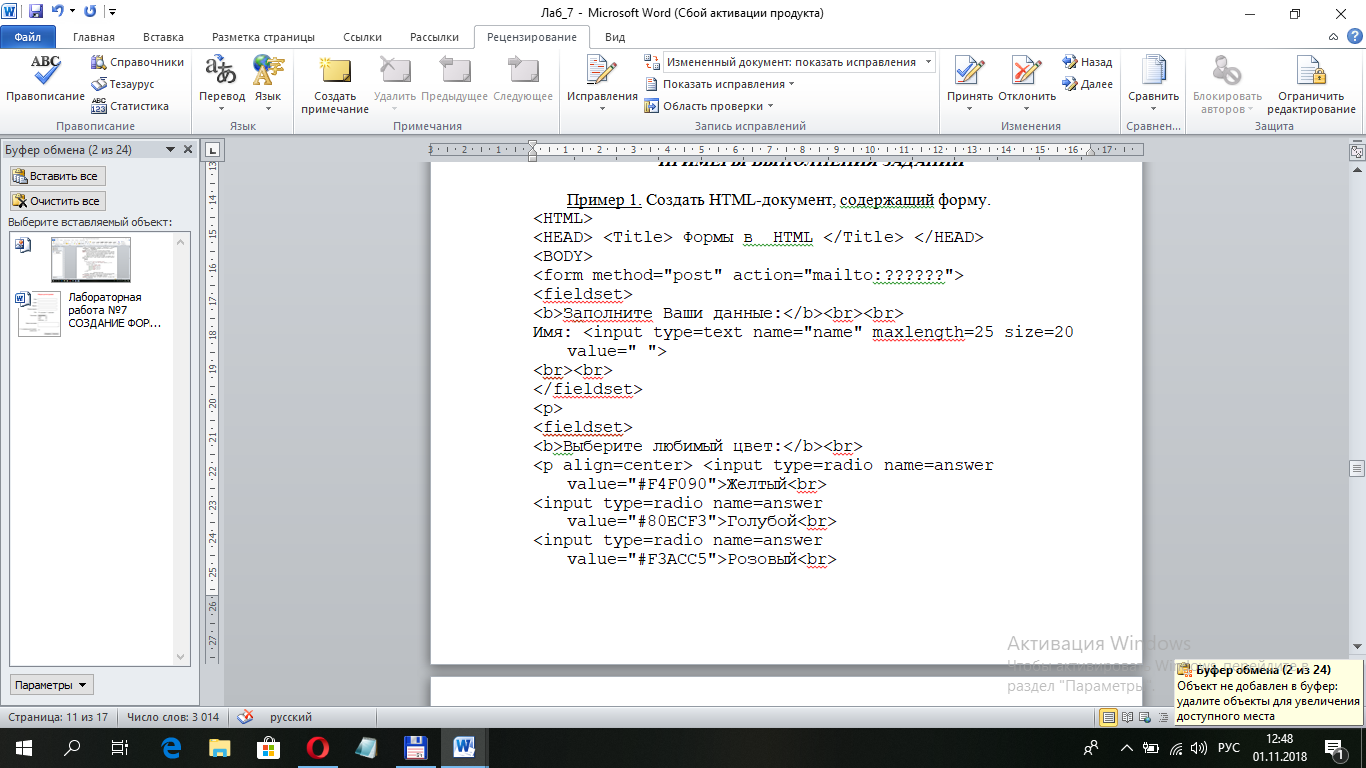
Кнопки позволяют пользователям передавать данные в форму, очищать содержимое формы или предпринимать какие-либо другие действия. Можно создавать границы, изменять фон и выравнивать текст на кнопке.

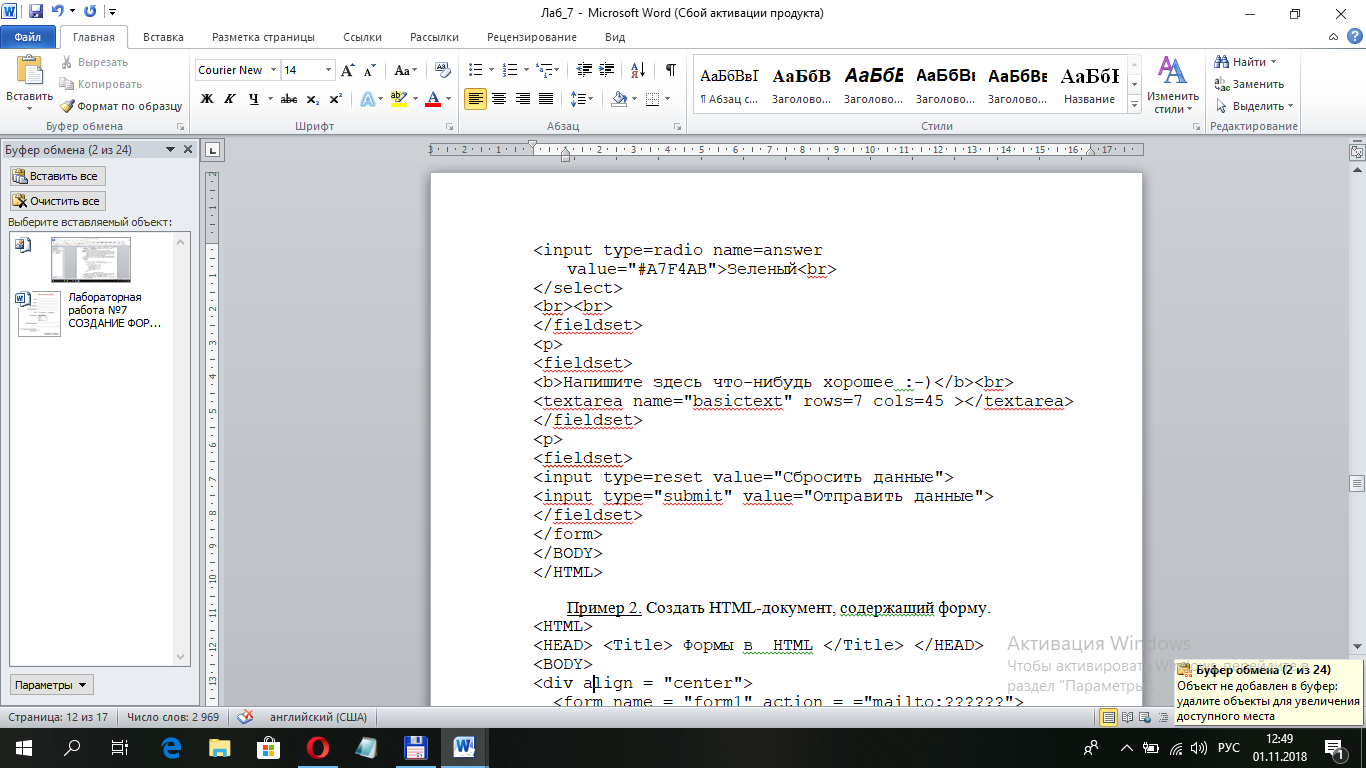
Таблица 3.3. ­ Атрибуты тега <**BUTTON**>

|  |  |
| --- | --- |
| **Атрибут** | **Значение / описание** |
| autofocus | Устанавливает фокус на кнопке при загрузке страницы. |
| disabled | Отключает кнопку, делая ее некликабельной. |
| form | Указывает на одну или несколько форм, которым принадлежит данная кнопка. Значение атрибута — идентификатор соответствующей формы. |
| formaction | Значение атрибута содержит u;rl-адрес обработчика данных формы, отправляемых при нажатии на кнопку. Только для кнопки типа type="submit". Переопределяет значение атрибута **action**, указанного для элемента <**form**>. |
| formenctype | Задает тип кодировки данных формы перед отправкой на сервер при нажатии на кнопки типа type="submit". Переопределяет значение атрибута enctype, указанного для элемента <**form**>. Возможные значения: application/x-www-form-urlencoded ­ значение по умолчанию. Все символы перед отправкой будут закодированы.  multipart/form-data ­ символы не кодируются. Используется в случае, когда с помощью формы загружаются файлы.  text/plain ­ символы не кодируются, а пробелы заменяются на символ +. |
| formmethod | Атрибут определяет метод, который браузер будет использовать для отправки формы. Переопределяет значение атрибута method, указанного для элемента <**form**>. Указывается только для кнопок типа type="submit". Возможные значения:  **Get** — данные из формы (пара имя/значение) добавляются в url-адрес и отправляются на сервер. Данный способ имеет ограничения по размеру отправляемых данных и не подходит для отправки паролей и конфиденциальной информации.  **Post** — данные из формы добавляются в виде http-запроса. Метод является более надежным и безопасным, чем get и не имеет ограничений по размеру. |
| formnovalidate | Атрибут задает, что данные формы не должны проверяться при отправке. Указывается только для кнопок типа type="submit". |
| formtarget | Атрибут задает, в каком окне выводить результат после отправки формы. Указывается только для кнопок типа type="submit". Переопределяет значение атрибута target, указанного для элемента <form>.  \_blank ­ загружает ответ в новое окно/вкладку  \_self ­ загружает ответ в то же окно (значение по умолчанию)  \_parent ­ загружает ответ в родительский фрейм  \_top ­ загружает ответ во весь экран  framename ­ загружает ответ во фрейм с указанным именем. |
| name | Задает имя кнопки, значение атрибута — текст. Используется для ссылки на данные формы, после того как форма была отправлена, или для ссылки на данную кнопку (кнопки) в JavaScript. |
| type | Определяет тип кнопки. Возможные значения:  button ­ кликабельная кнопка  reset — кнопка сброса, возвращает первоначальное значение  submit — кнопка для отправки данных формы. |
| value | Задает значение по умолчанию, отправляемое при нажатии на кнопку. |

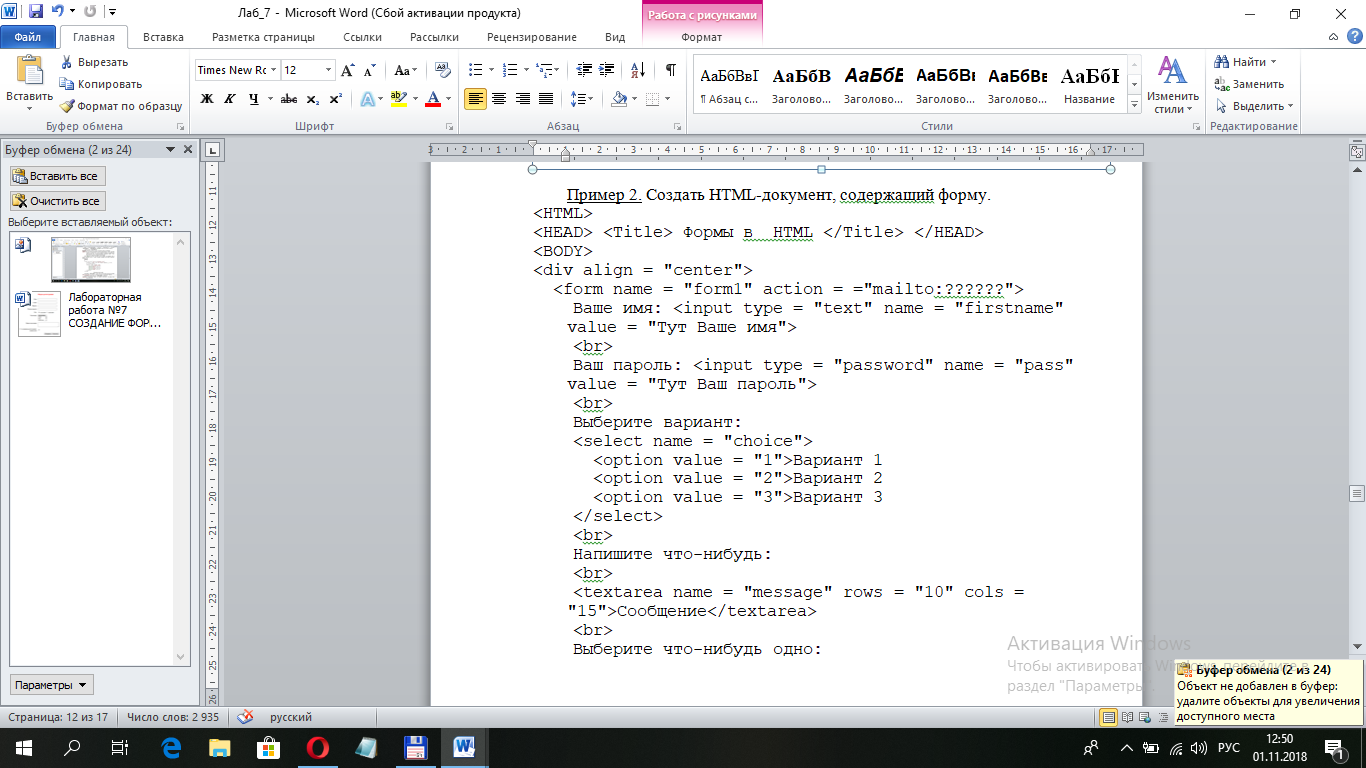
***ПРИМЕРЫ ВЫПОЛНЕНИЯ ЗАДАНИЙ***

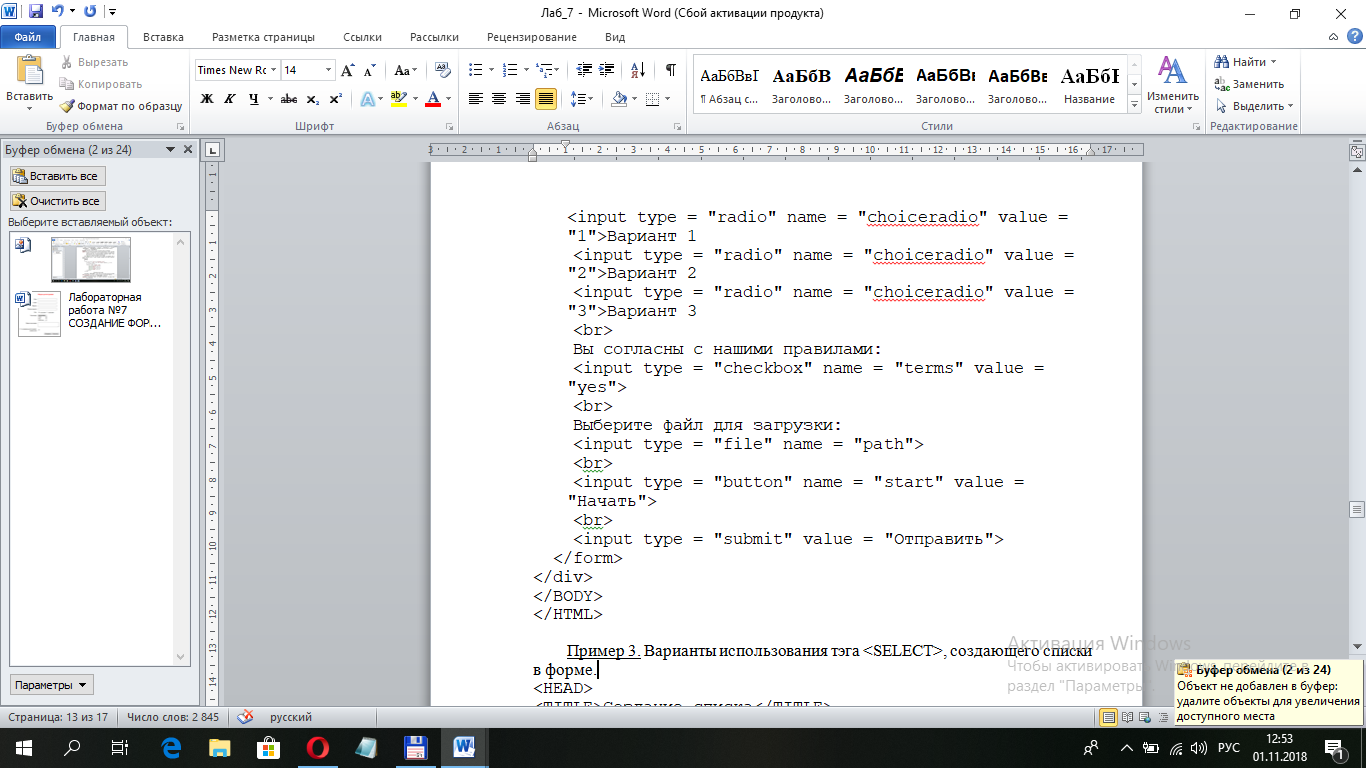
Пример 1. Создать HTML-документ, содержащий форму.



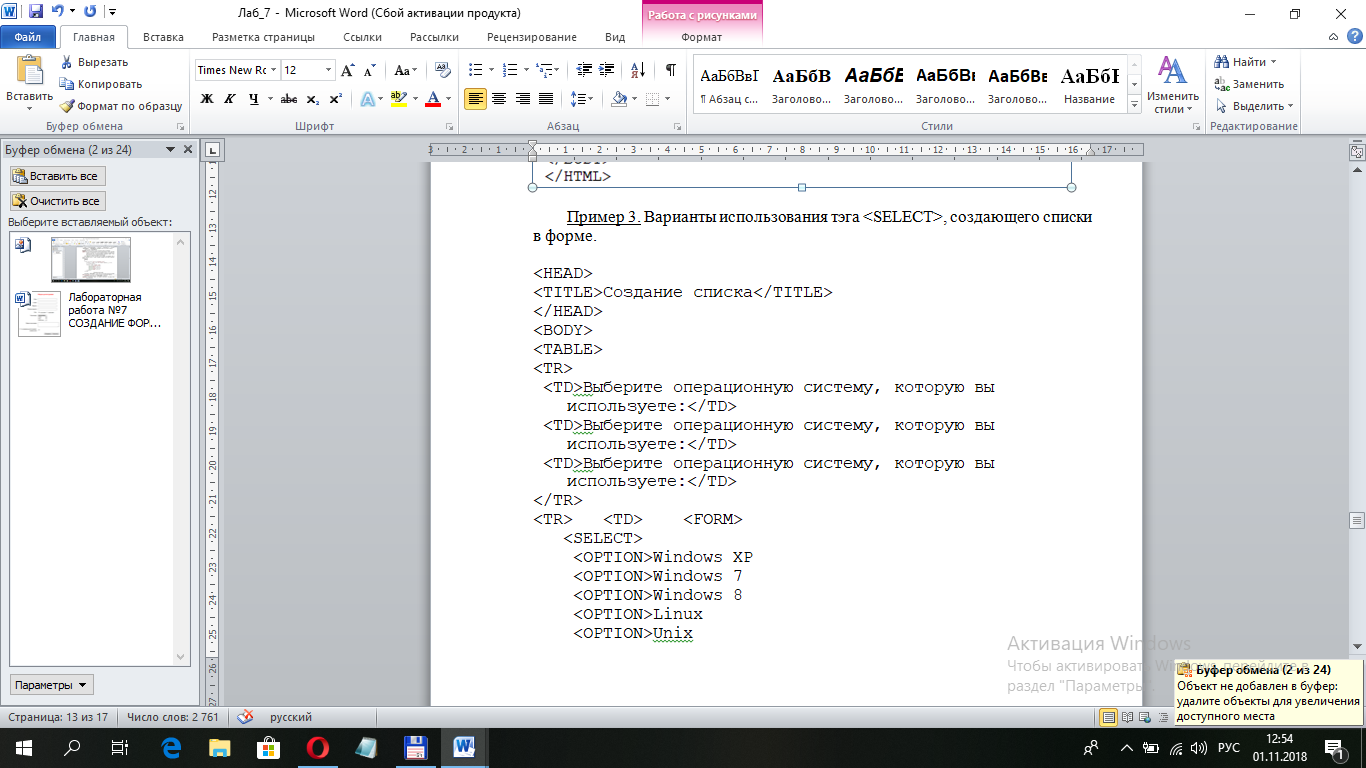


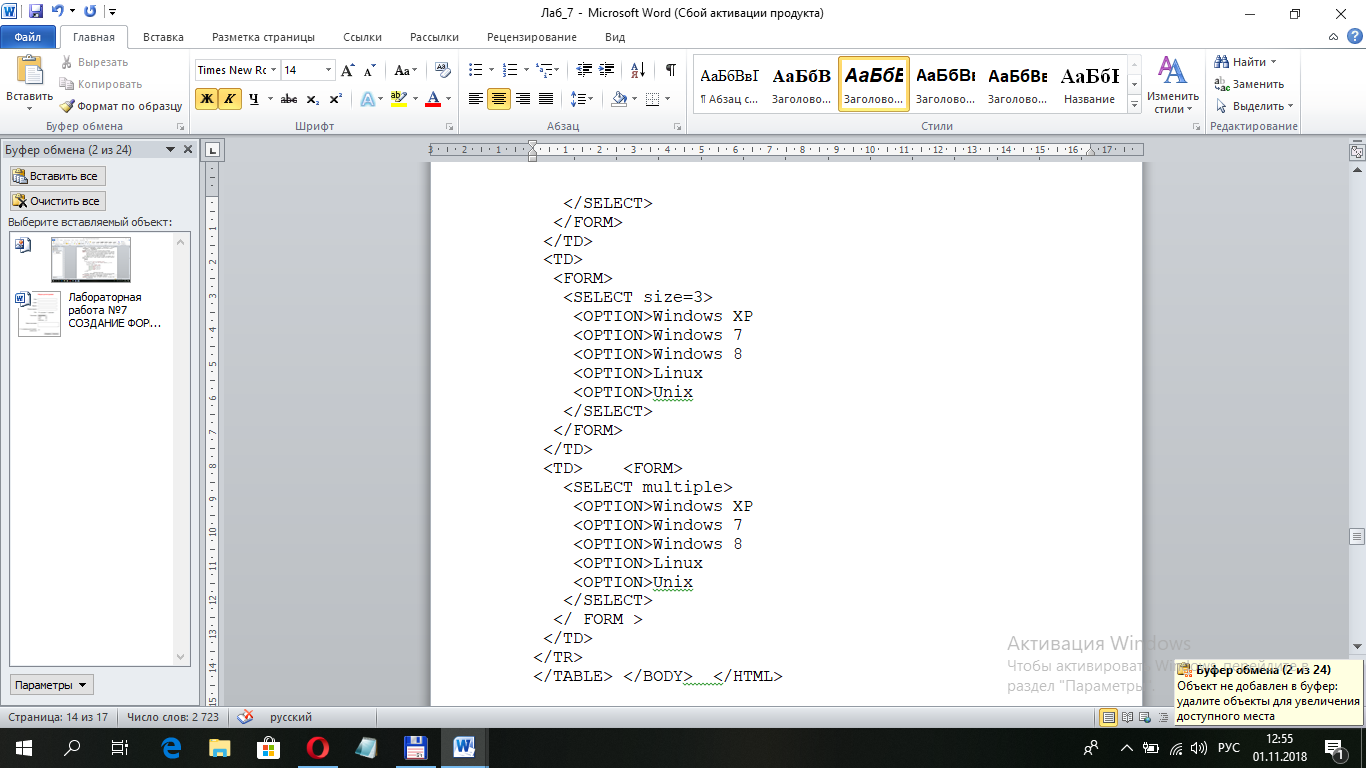
Пример 2. Создать HTML-документ, содержащий форму.





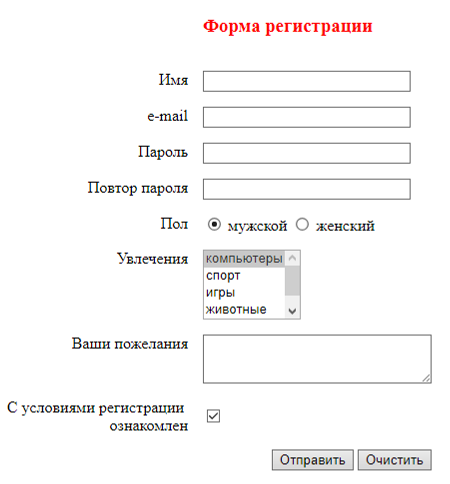
Пример 3. Варианты использования тэга <SELECT>, создающего списки в форме.





1. ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ
2. **Изучите теоретические сведения.**
3. Выполните задания с изображениями на веб-странице:

Задание 1. Создайте HTML-документ по предложенному образцу:



В списке увлечений должно быть более 7 наименований.

\*Сделать проверку на e-mail, при отправке данных все поля должны быть заполнены.

Задание 2. Сверстать анкету при помощи HTML форм. Тематику анкеты можно выбрать любую.

В работе вам необходимо использовать:

* Элемент <form>. Знать атрибуты action и method (какие значения принимают данные атрибуты, чем отличаются принимаемые значения);
* Элемент <input>. Атрибут type, данного элемента, должен принимать значения checkbox, file, password, radio, text, tel, email, submit, reset;
* Элемент <label>, понимать каким образом данный элемент связан с конкретным элементом <input>;
* Элементы <fieldset> и <legend>.

P.S. Это необходимый минимум, прочие элементы использовать можно.