Ответы на контрольные вопросы

1. **Приведите пример структуры HTML5 документа.**

<!DOCTYPE html>

<html>

<head>

<title>Заголовок документа</title>

</head>

<body>

<h1>Заголовок</h1>

<p>Здесь содержание документа...</p>

</body>

</html>

|  |  |
| --- | --- |
| Тег | Описание |
| <!DOCTYPE html> | Этот тег определяет тип документа и версию HTML. |
| <html> | Этот тег охватывает полный HTML-документ и в основном состоит из заголовка документа, который представлен тегами <head>...</head> и телом документа, которое представлено тегами <body>...</body>. |
| <head> | Этот тег представляет заголовок документа, который может содержать другие html-теги, такие как <title>, <link> и т.д. |
| <title> | Тег <title> используется внутри тега <head>, чтобы указать название документа. |
| <body> | Этот тег представляет тело документа, в котором хранятся другие html-теги, такие как <h1>, <div>, <p> и другие. |
| <h1> | Этот тег представляет заголовок. |
| <p> | Этот тег представляет абзац. |

1. **Какие новые семантические элементы появились в HTML5?**

В HTML4 веб-разработчики использовали свои собственные имена в идентификаторах/классах элементов для их стилизации: header, top, bottom, footer, menu, navigation, main, container, content, article, sidebar, topnav и т.п.Такое положение дел не позволяло поисковым системам корректно идентифицировать роль того или иного контента веб-страницы.

Благодаря новым элементам HTML5 (<header>, <footer>, <nav>, <section>, <article>), сделать это стало гораздо проще.

|  |  |
| --- | --- |
| Тег | Описание |
| <article> | Определяет статью |
| <aside> | Определяет блок сбоку от основного контента |
| <details> | Определяет дополнительную информацию, которую пользователь может открывать или закрывать |
| <figcaption> | Определяет пояснение для элемента <figure> |
| <figure> | Используется для группирования различных самодостаточных элементов - иллюстраций, диаграмм, фотографий, листингов кода и т.д. |
| <footer> | Определяет "подвал" документа или раздела |
| <header> | Определяет "шапку" документа или раздела |
| <main> | Определяет основной контент документа |
| <mark> | Определяет маркированный/подсвеченный текст |
| <nav> | Определяет блок навигационных ссылок |
| <section> | Определяет раздел в документе |
| <summary> | Определяет видимый заголовок элемента <details> |
| <time> | Определяет дату/время |

1. **Перечислите и опишите семантические элементы HTML5 для текстового содержимого.**
2. Теги заголовков

Заголовки являются важными элементами веб-страницы, они упорядочивают текст, формируя его визуальную структуру. Теги <h1>...<h6> должны использоваться только для выделения заголовков нового раздела или подраздела. При использовании заголовков необходимо учитывать их иерархию, т.е. за <h1> должен следовать <h2> и т.д. Также не допускается вложение других тегов в теги <h1>...<h6>.

1.1. Тег <h1>

Заголовок самого верхнего уровня, на странице рекомендуется использовать только один раз, по возможности частично дублируя заглавие страницы. Тег <h1> должен быть уникальным для каждой страницы сайта. Рекомендуется прописывать тег в начале статьи, используя ключевое слово в тексте заголовка. Размер шрифта в браузере равен 2em, верхний и нижний отступ по умолчанию 0.67em.

1.2. Тег <h2>

Им обозначаются подзаголовки тега <h1>. Размер шрифта в браузере равен 1.5em, верхний и нижний отступ по умолчанию 0.83em.

1.3. Тег <h3>

Показывает подзаголовки тега <h2>. Размер шрифта в браузере равен 1.17em, верхний и нижний отступ по умолчанию 1em.

1.4. Теги <h4>, <h5>, <h6>

Обозначают подзаголовки четвёртого, пятого и шестого уровня. Размер шрифта в браузере равен 1em / 0.83em / 0.67em, верхний и нижний отступ по умолчанию 1.33em / 1.67em / 2.33em соответственно.

Для всех тегов доступны ‎глобальные атрибуты.

1. Теги для форматирования текста

2.1. Тег <b>

Задаёт полужирное начертания шрифта. Выделяет текст без акцента на его важность.

Для тега доступны ‎глобальные атрибуты.

2.2. Тег <em>

Отображает шрифт курсивом, придавая тексту значимость.

Для тега доступны ‎глобальные атрибуты.

2.3. Тег <i>

Отображает шрифт курсивом.

Для тега доступны ‎глобальные атрибуты.

2.4. Тег <small>

Уменьшает размер шрифта на единицу по отношению к обычному тексту.

Для тега доступны ‎глобальные атрибуты.

2.5. Тег <strong>

Задаёт полужирное начертание шрифта, относится к тегам логической разметки, указывая браузеру на важность текста.

Для тега доступны ‎глобальные атрибуты.

2.6. Тег <sub>

Используется для создания нижних индексов. Сдвигает текст ниже уровня строки, уменьшая его размер.

Для тега доступны ‎глобальные атрибуты.

2.7. Тег <sup>

Используется для создания степеней. Сдвигает текст выше уровня строки, уменьшая его размер.

Для тега доступны ‎глобальные атрибуты.

2.8. Тег <ins>

Выделяет текст в новой версии документа, подчёркивая его.

Для тега доступны следующие атрибуты: cite, datetime.

2.9. Тег <del>

Перечёркивает текст. Используется для выделения текста, удаленного из документа.

Для тега доступны следующие атрибуты: cite, datetime.

2.10. Тег <mark>

Применяется для выделения фрагментов текста в справочных целях, окрашивая блок символов желтым цветом.

Для тега доступны ‎глобальные атрибуты.

1. Теги для ввода «компьютерного» текста

3.1. Тег <code>

Служит для выделения фрагментов программного кода. Отображается моноширинным шрифтом.

Для тега доступны ‎глобальные атрибуты.

3.2. Тег <kbd>

Отмечает фрагмент как вводимый пользователем с клавиатуры. Отображается моноширинным шрифтом.

Для тега доступны ‎глобальные атрибуты.

3.3. Тег <samp>

Применяется для выделения результата, полученного в ходе выполнения программы. Отображается моноширинным шрифтом.

Для тега доступны ‎глобальные атрибуты.

3.4. Тег <var>

Выделяет имена переменных, отображая курсивом.

Для тега доступны ‎глобальные атрибуты.

3.5. Тег <pre>

Позволяет вывести текст на экран, сохранив изначальное форматирование. Пробелы и переносы строк при этом не удаляются.

Для тега доступны ‎глобальные атрибуты.

1. Теги для оформления цитат и определений

4.1. Тег <abbr>

Применяется для форматирования аббревиатур. Браузером обычно подчеркивается пунктирной линией. Расшифровка сокращения осуществляется с помощью атрибута title, она появляется при наведении курсора мыши на текст.

Для тега доступны ‎глобальные атрибуты.

4.2. Тег <bdo>

Используется для замещения текущего направления текста, т.е. текст в теге отображается зеркально.

Для тега доступен атрибут dir.

4.3. Тег <blockquote>

Выделяет цитаты внутри документа, выделяя его отступами и переносами строк.

Для тега доступен атрибут cite.

4.4. Тег <q>

Используется для выделения коротких цитат. Браузерами заключается в кавычки.

Для тега доступен атрибут cite.

4.5. Тег <cite>

Применяется для выделения цитат, названий произведений, сносок на другие документы.

Для тега доступны ‎глобальные атрибуты.

4.6. Тег <dfn>

Позволяет выделить текст как определение. Несмотря на наличие специального тега, рекомендуется выделять текст силами CSS.

Для тега доступен атрибут title.

1. Абзацы, средства переноса текста

5.1. Тег <p>

Разбивает текст на отдельные абзацы, отделяя друг от друга пустой строкой. Браузер автоматически добавляет верхний и нижний отступ, равный 1em, при этом отступы соседних абзацев «схлопываются».

Для тега доступны ‎глобальные атрибуты.

5.2. Тег <br>

Переносит текст на следующую строку, создавая разрыв строки.

Для тега доступны ‎глобальные атрибуты.

5.3. Тег <hr>

Используется для разделения контента на веб-странице. Отображается в виде горизонтальной линии.

Для тега доступны ‎глобальные атрибуты.

1. **Перечислите и опишите элементы для поддержки аудио, видео и подключаемых модулей.**

**Элемент <audio>** внедряет звуковой файл на веб-страницу. Это заменяемый элемент, но он не относится к пустым элементам, поэтому он может содержать и собственный контент и другие элементы. В своей простейшей форме элемент <audio> должен иметь только атрибут src, указывающий путь к звуковому файлу:

<audio src="audio/file.mp3"></audio>

Однако он редко используется в такой простой форме. По умолчанию элемент <audio> не имеет элементов управления и поэтому полностью невидимый. Он будет располагаться в HTML-документе и может быть доступен в браузере или JavaScript, но простое внедрение аудио файла не принесет особой пользы. С помощью логического атрибута controls можно добавить к аудио-файлу его собственные элементы управления, которые будут отображаться на веб-странице в месте расположения элемента <audio>:

<audio src="audio/file.mp3" controls></audio>

Эти элементы управления предоставляются браузером и могут визуально отличаться в зависимости от браузера, но предоставляемые функции остаются неизменными:

* кнопка play/pause
* полоса прокрутки
* отображение времени
* громкость

Элемент <audio> может содержать один или несколько элементов <source>, каждый из которых будет ссылаться на свой аудио-файл, что будет полезно из-за отсутствия единого формата, поддерживаемого всеми браузерами. В этом случае браузер будет проигрывать первый медиа-файл, который он поддерживает, игнорируя все остальные

**Элемент <video>** встраивает цифровое видео на веб-страницу. Как и в случае с <audio>, элемент <video> может иметь атрибут src, указывающий URL-адрес видеофайла:

<video src="video/file.mp4" controls></video>

Атрибут controls добавляет элементы управления, с помощью которых пользователь сможет взаимодействовать с плеером. Атрибут autoplay указывает браузеру, что видео должно начать воспроизведение сразу, как только страница будет загружена, если он не указан, то в плеере будет отображен первый кадр исходного видео. Но чаще всего он не несет никакой полезной информации о том, что будет представлено с помощью видео-контента. Вместо этого можно предоставить любое изображение, указав к нему путь в атрибуте poster

1. **Перечислите и опишите типы позиционирования, которые вы знаете.**

* STATIC. Это значение по умолчанию для любого элемента на странице. Это самый простой тип и практически ничего не означает. Все выглядит как должно выглядеть.
* RELATIVE. Этот тип позиционирования наиболее трудный и неправильно используемый. Если ВЫ присвоете элементу position: relative;, но не придадите каких-либо других атрибутов (top, left, bottom или right), тогда ничего не изменится. Все останется на своих местах (также как и первым типом позиционирования). Но если Вы добавите атрибуты (например, top: 10px), тогда элемент сместится на 10 пикселей вниз от того места, где он обычно располагался. Это очень полезно при выравнивании элементов формы в один ряд.
* ABSOLUTE. Очень мощный тип позиционирования, который позволяет поместить любой элемент в любом месте страницы. Для этого используются аргументы top, left, bottom и right. Самое главное запомнить, что элементы с абсолютным позиционированием как бы вырваны из структуры страницы. На элемент с таким типом позиционирования не влияют другие элементы, и он также не влияет на них. Это необходимо всегда держать в уме при создании сайтов. Неправильное или частое использование такого типа может ограничить гибкость Вашего сайта.
* FIXED. Этот тип позиционирования очень редкий. Элемент с таким типом позиционируется относительно к окну просмотра, то есть браузеру. При скролле, такой элемент будет оставаться всегда на том же месте. Преимуществом является то, что навигация всегда на виду, и вообще это интересный эффект для посетителей. Минусы возможно не видны с первого взгляда. Если зайти на эту страницу с ноутбука с маленьким экраном, например, тогда панель с навигацией будет видна не полностью (и никак не удасться ее просмотреть, так как не будет скролла).

1. **Позиционирование внутри элемента**.

* Позиционирование через float

Один из способов позиционирования элементов на странице — через свойство float. Это свойство довольно универсально и может применяться разными путями.

По существу, свойство float берёт элемент, убирает его из обычного потока страницы и позиционирует слева или справа от родительского элемента. Все остальные элементы на странице будут обтекать такой элемент. Например, абзацы будут обтекать изображение, если для элемента <img> установлено свойство float.

Когда свойство float применяется к нескольким элементам одновременно, это даёт возможность создать макет с обтекаемыми элементами расположенными рядом или напротив друг друга, как показано в многоколоночном макете.

Свойство float принимает несколько значений, два самых популярных — это left и right, они позволяют элементу располагаться слева или справа от своего родителя.

* Позиционирование через inline-block

В дополнение к использованию float, ещё один способ, которым мы можем позиционировать контент — это применение свойства display в сочетании со значением inline-block. Метод с inline-block, как мы ещё обсудим, в первую очередь полезен для компоновки страниц или для размещения элементов в линию рядом друг с другом.

* Абсолютное позиционирование

Значение absolute для свойства position отличается от значения relative тем, что элемент с абсолютным позиционированием не появляется в обычном потоке документа, исходное пространство и положение абсолютно позиционируемого элемента не резервируется.

Кроме того, абсолютно позиционируемые элементы перемещаются относительно их ближайшего относительно позиционированного родительского элемента. Если относительно позиционированного родителя не существует, то абсолютно позиционированный элемент будет позиционироваться относительно элемента <body>.

1. **Перечислите и опишите составные компоненты формы.**

|  |  |
| --- | --- |
| Тег | Описание |
| <form> | Устанавливает форму на веб-странице |
| <input> | Создает различные элементы формы (текстовые поля, кнопки, переключатели и т.д.) в зависимости от значения атрибута type |
| <textarea> | Создает многострочное текстовое поле |
| <fieldset> | Группирует элементы формы. Может также содержать внутри себя элемент legend с условными обозначениями элементов формы, составляющих логическую группу |
| <button> | Многофункциональная кнопка |
| <label> | Устанавливает связь между определенной меткой и элементом формы (input, select, textarea) |
| <legend> | Применяется для создания заголовка группы элементов формы, которая определяется с помощью тега fieldset |
| <optgroup> | Представляет собой контейнер, внутри которо-го располагаются теги option, объединенные в одну группу |
| <option> | Определяет отдельные пункты списка, созда-ваемого с помощью контейнера select |
| <select> | Позволяет создать элемент интерфейса в виде раскрывающегося или прокручиваемого списка |
| <datalist> | Позволяет предоставить раскрывающийся список с предложенными значениями для любого типа поля ввода |

1. **Перечислите и опишите глобальные атрибуты тега в HTML5.**

* Accesskey - Генерирует сочетания клавиш для доступа к текущему элементу. Состоит из разделенного пробелами списка символов. Браузер в первую очередь выбирает те клавиши, которые существуют на раскладке клавиатуры. Применяется к следующим элементам: <a>, <area>, <button>, <input>, <label>, <legend>, <textarea>.Принимаемые значения: перечень названий клавиш.
* Class - Определяет имя класса для элемента (используется для определения класса в таблице стилей).Принимаемые значения: имя класса.
* Contenteditable - Определяет, может ли пользователь редактировать содержимое (контент). Позволяет преобразовать любое поле HTML в редактируемый элемент.Принимаемые значения: true/false.
* Dir - Определяет направление текста контента в элементах <bdo> и <bdi>. Принимаемые значения: ltr/rtl/auto.
* Draggable - Определяет, может ли пользователь перетащить элемент. Принимаемые значения: true/false/auto.
* Hidden - Указывает на то, что элемент должен быть скрыт. Принимаемые значения: hidden.
* Id - Определяет уникальный идентификатор элемента. Принимаемые значения: id - идентификатор элемента.
* Lang - Определяет код языка содержимого (контента) в элементе. Принимаемые значения: код языка.
* Spellcheck - Указывает, подлежит ли содержимое элемента проверке орфографии и грамматики. Принимаемые значения: true/false.
* Style - Указывает на код CSS, применяемую для оформления элемента. Принимаемые значения: код CSS.
* Tabindex - Определяет порядок перехода к элементу при помощи клавиши TAB. Принимаемые значения: порядковый номер.
* Title - Определяет дополнительную информацию об элементе, задавая всплывающую подсказку для страницы. Принимаемые значения: текст.
* Translate - Разрешает или запрещает перевод текста внутри элемента. Принимаемые значения: yes/no.