**1. Пример структуры HTML5 документа**

**Базовая структура документа HTML5 чистая и семантическая.**

html

<!DOCTYPE html>

<html lang="ru">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Заголовок страницы</title>

</head>

<body>

</body>

</html>

**2. Новые семантические элементы в HTML5**

**HTML5 представил ряд семантических элементов, которые описывают своё назначение и содержание как браузерам, так и разработчикам.**

<header>: Заголовок страницы или раздела.

<footer>: Нижний колонтитул страницы или раздела.

<nav>: Навигационный раздел сайта.

<main>: Основное, уникальное содержимое документа.

<article>: Независимый, самодостаточный контент (пост в блоге, статья новости).

<section>: Тематическая группировка контента, обычно с заголовком.

<aside>: Контент, косвенно связанный с основным (сайдбар, боковая панель).

<figure> и <figcaption>: Для самостоятельного контента с подписью (изображения, диаграммы, код).

<mark>: Выделяет текст как помеченный или выделенный (как маркером).

<time>: Помечает текст как дату или время.

**3. Семантические элементы HTML5 для текстового содержимого**

**Эти элементы придают смысл текстовым блокам.**

<h1>-<h6>: Заголовки разного уровня. Самый важный <h1>, наименее важный <h6>.

<p>: Абзац текста.

<blockquote>: Блочная цитата, обычно с отступами.

<q>: Короткая строчная цитата. Браузер обычно автоматически добавляет кавычки.

<cite>: Название творческой работы (книги, фильма, песни).

<em>: Семантическое выделение для придания акцента. Отображается курсивом по умолчанию.

<strong>: Семантическое выделение для обозначения важности или срочности. Отображается жирным шрифтом по умолчанию.

<code>: Фрагмент кода.

<pre>: Сохраняет форматирование текста, включая пробелы и переносы строк. Используется для показа кода.

<mark>: Визуально и семантически выделяет часть текста как имеющую relevance (например, результат поиска).

<time>: Помечает текст как дату/время. Может иметь атрибут datetime для машинно-читаемого формата.

**4. Элементы для поддержки аудио, видео и подключаемых модулей**

<audio>: Для встраивания звукового контента.

Атрибуты: src, controls (показать элементы управления), autoplay, loop, muted.

Пример:

html

<audio controls>

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

Ваш браузер не поддерживает аудио элемент.

</audio>

<video>: Для встраивания видео.

Атрибуты: src, controls, autoplay, loop, muted, poster (изображение-заставка), width, height.

Пример:

html

<video controls width="600">

<source src="movie.mp4" type="video/mp4">

Ваш браузер не поддерживает видео тег.

</video>

<embed>: Универсальный контейнер для внешнего контента или плагинов (например, Flash). Устарел для большинства задач.

<object> и <param>: Более мощная и гибкая альтернатива <embed> для встраивания различных типов объектов. <param> используется для передачи параметров плагину внутри <object>.

**5. Типы позиционирования в CSS**

Свойство position определяет, как элемент позиционируется на странице.

static (Статическое):

Значение по умолчанию. Элемент располагается в нормальном потоке документа.

Свойства top, right, bottom, left и z-index на него не влияют.

relative (Относительное):

Элемент располагается в нормальном потоке, но его можно сдвинуть относительно его исходного положения с помощью top, right, bottom, left.

Соседние элементы ведут себя так, как будто он не сдвинут.

absolute (Абсолютное):

Элемент изымается из нормального потока. Место, которое он занимал, освобождается.

Позиционируется относительно своего ближайшего позиционированного предка (у которого position не static). Если такого нет — относительно <html>.

Положение задаётся свойствами top, right, bottom, left.

fixed (Фиксированное):

Элемент изымается из нормального потока.

Позиционируется относительно окна просмотра (viewport). Остаётся на одном и том же месте при прокрутке.

Идеально для фиксированных header'ов, боковых панелей или кнопок "Наверх".

sticky (Липкое):

Гибрид relative и fixed. Элемент ведёт себя как relative, пока не достигнет заданного порога прокрутки (top: 10px, например), после чего начинает вести себя как fixed.

Часто используется для "липких" заголовков таблиц или навигации.

**6. Позиционирование внутри элемента**

Это достигается с помощью комбинации position: relative у родителя и position: absolute у дочернего элемента.

Родительскому элементу задаётся position: relative (он становится "точкой отсчета").

Дочернему элементу задаётся position: absolute.

Теперь свойства top, right, bottom, left дочернего элемента будут отсчитываться от границ родительского элемента.

Пример:

html

<div class="parent">

<div class="child">Я внутри родителя!</div>

</div>

css

.parent {

position: relative;

width: 300px;

height: 200px;

background: lightgray;

}

.child {

position: absolute;

top: 20px; /\* 20px от верха .parent \*/

left: 50px; /\* 50px от левого края .parent \*/

background: coral;

}

**7. Составные компоненты формы**

Форма (<form>) состоит из следующих ключевых компонентов:

<form>: Контейнер для всех элементов формы. Имеет атрибуты action (URL для отправки) и method (GET/POST).

Поля ввода (<input>):

Универсальное поле. Тип задаётся атрибутом type:

text: Текстовое поле.

password: Поле для пароля.

email: Для адреса электронной почты.

number: Для чисел.

checkbox: Флажок.

radio: Переключатель.

file: Для загрузки файлов.

submit: Кнопка для отправки формы.

reset: Кнопка для сброса формы.

И многие другие (date, color, range и т.д.).

<label>: Подпись к полю ввода. Связывается с полем через атрибут for (который должен равняться id поля) или путем оборачивания поля в тег <label>.

<textarea>: Многострочное текстовое поле.

<select> и <option>: Раскрывающийся список. <select> — контейнер, <option> — элементы списка.

<button>: Кнопка. Может быть типа submit, reset или button.

<fieldset> и <legend>: Группируют логически связанные элементы формы. <fieldset> — рамка, <legend> — заголовок группы.

**8. Глобальные атрибуты тега в HTML5**

Глобальные атрибуты могут быть использованы с любым HTML-элементом.

id: Уникальный идентификатор элемента в документе.

class: Список классов элемента, разделённых пробелами, для стилизации или доступа через JavaScript.

style: Позволяет задавать CSS-стили напрямую элементу (инлайново).

title: Всплывающая подсказка, которая появляется при наведении курсора на элемент.

lang: Определяет язык текстового содержимого элемента (например, lang="ru").

data-\* (пользовательские атрибуты данных): Позволяет хранить пользовательские данные. \* заменяется на любое имя (напр., data-user-id="123").

hidden: Указывает, что элемент должен быть скрыт.

tabindex: Определяет порядок перехода между элементами с помощью клавиши Tab.

contenteditable: Указывает, может ли пользователь редактировать содержимое элемента.

draggable: Определяет, можно ли перетаскивать элемент с помощью Drag and Drop API.

accesskey: Определяет клавишу быстрого доступа для активации/фокусировки элемента.

dir: Определяет направление текста (ltr - слева направо, rtl - справа налево, auto).