**Профессиональное образовательное частное учреждение**

**«Московский кооперативный техникум им. Г. Н. Альтшуля»**

**КУРСОВАЯ РАБОТА**

**ПМ.09 Проектирование, разработка и оптимизация веб-приложений МДК 09.01**

**Проектирование и разработка веб-приложений**

**Тема:**

**(название темы)**

**Выполнил(а)\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Специальность: 09.02.07 «Информационные системы и программирование»**

**Курс 4**

**Группа ИС-41**

**Руководитель**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**(фамилия, инициалы, подпись)**

**Мытищи**

**2024**

**СОДЕРЖАНИЕ**

**ВВЕДЕНИЕ**

**РАЗДЕЛ 1. ТЕХНИЧЕСКИЙ ПРОЕКТ**

**1.1 Описание предметной области**

**1.2 Описание технологии обработки задач**

**1.3 Информационно-логическая модель системы**

**1.4 Характеристика инструментальных средств разработки**

**РАЗДЕЛ 2. РАБОЧИЙ ПРОЕКТ**

**2.1 Разработка веб-приложений**

**2.2 Отладка и тестирование веб-приложений**

**2.3 Публикация веб-приложения**

**2 .4 Руководство программиста**

**ЗАКЛЮЧЕНИЕ**

**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

**ПРИЛОЖЕНИЯ**

**Курсовая**

**Олейников Григорий Николаевич**

**Разработка одностроничного лэндинга зоомагазина**

**Инструменты;**

1. HTML (HyperText Markup Language)
2. CSS (Cascading Style Sheets)
3. JavaScript
4. Google Fonts
5. CSS Grid
6. Intersection Observer API
7. Анимации в CSS

1. HTML (HyperText Markup Language)

История: HTML был разработан в начале 1990-х годов Тимом Бернерсом-Ли, который создал Всемирную паутину. Первая версия HTML была выпущена в 1993 году и содержала базовые теги для форматирования текста и создания ссылок. С тех пор HTML претерпел множество изменений. В 1995 году была представлена версия HTML 2.0, которая добавила поддержку форм и таблиц. В 1999 году вышла версия HTML 4.01, которая улучшила поддержку мультимедиа и скриптов. В 2014 году был представлен HTML5, который стал значительным шагом вперед, добавив новые семантические элементы и API для работы с мультимедиа.

Назначение: HTML используется для создания структуры веб-страниц. Он определяет, как контент будет отображаться в браузере, включая текст, изображения, ссылки и другие элементы.

Возможности: HTML позволяет создавать заголовки, параграфы, списки, таблицы, формы и ссылки. HTML5 добавил поддержку новых типов контента, таких как видео и аудио, а также API для работы с геолокацией, локальным хранилищем и многими другими функциями. HTML также поддерживает атрибуты, которые позволяют добавлять дополнительную информацию к элементам.

2. CSS (Cascading Style Sheets)

История: CSS был разработан в 1996 году для отделения содержания от представления. Это позволило разработчикам управлять стилями веб-страниц более эффективно. Первая версия CSS (CSS1) была выпущена в 1996 году, а CSS2 в 1998 году добавил поддержку медиа-запросов и позиционирования. В 2011 году началась работа над CSS3, который стал модульным, что позволило добавлять новые функции по мере их разработки.

Назначение: CSS используется для стилизации веб-страниц. Он позволяет задавать внешний вид элементов HTML, включая цвета, шрифты, отступы, размеры и расположение.

Возможности: CSS позволяет создавать сложные макеты, управлять типографикой, добавлять фоны, границы и тени. CSS3 добавил новые возможности, такие как анимации, трансформации и медиа-запросы для адаптивного дизайна. Это позволяет создавать интерфейсы, которые хорошо выглядят на различных устройствах и экранах.

3. JavaScript

История: JavaScript был создан в 1995 году Бренданом Айком, когда он работал в Netscape. Изначально язык назывался Mocha, затем был переименован в LiveScript, а затем в JavaScript. JavaScript быстро стал стандартом для добавления интерактивности на веб-страницы. В 1997 году JavaScript был стандартизирован как ECMAScript, и с тех пор язык претерпел множество изменений и улучшений.

Назначение: JavaScript используется для добавления интерактивности и динамики на веб-страницы. Он позволяет разработчикам создавать интерактивные элементы, обрабатывать события и взаимодействовать с сервером.

Возможности: JavaScript позволяет создавать динамические элементы на веб-страницах, обрабатывать события (например, клики и наведение мыши), взаимодействовать с сервером через AJAX и манипулировать DOM. С помощью JavaScript можно создавать сложные веб-приложения, игры и даже серверные приложения с использованием Node.js.

4. Google Fonts

История: Google Fonts был запущен в 2010 году как бесплатный сервис, предоставляющий доступ к библиотеке шрифтов для веб-дизайнеров и разработчиков. Идея заключалась в том, чтобы упростить использование веб-шрифтов и сделать их доступными для всех, без необходимости загружать шрифты на сервер.

Назначение: Google Fonts позволяет разработчикам легко интегрировать шрифты на веб-страницы, улучшая типографику и визуальное восприятие.

Возможности: Google Fonts предлагает широкий выбор шрифтов, которые можно использовать на веб-сайтах. Пользователи могут выбирать из множества шрифтов и настраивать их стили, что позволяет создавать уникальные и привлекательные дизайны.

5. CSS Grid

CSS Grid Layout был представлен в спецификации CSS Grid в 2017 году и стал мощным инструментом для создания сложных макетов веб-страниц. Он позволяет разработчикам создавать сетки, которые могут адаптироваться к различным размерам экранов и устройствам.

Основное назначение CSS Grid — упрощение процесса создания адаптивных и отзывчивых макетов. С помощью CSS Grid разработчики могут легко управлять расположением элементов на странице, определять размеры строк и столбцов, а также создавать сложные структуры без необходимости использования фреймов или дополнительных библиотек.

Простота в создании сложных макетов.

Гибкость и адаптивность для различных устройств.

Упрощение работы с расположением элементов и их выравниванием.

6. Intersection Observer API

Intersection Observer API был представлен в 2016 году и стал стандартом для отслеживания видимости элементов на веб-странице. Он позволяет разработчикам асинхронно наблюдать за изменениями в пересечении элемента с его родительским контейнером или с областью просмотра (viewport).

Основное назначение Intersection Observer — оптимизация производительности. Вместо того чтобы постоянно проверять положение элементов с помощью событий прокрутки, разработчики могут использовать этот API для получения уведомлений, когда элемент становится видимым или скрытым. Это особенно полезно для ленивой загрузки изображений, анимаций, которые должны запускаться при прокрутке, и других интерактивных элементов.

Эффективное отслеживание видимости элементов.

Оптимизация производительности за счёт снижения нагрузки на обработку событий.

Удобство для реализации ленивой загрузки и анимаций при прокрутке.

8.Анимации в css

CSS-анимации были введены в спецификацию CSS3 и стали доступны в браузерах в начале 2010-х годов. Они позволяют разработчикам создавать плавные переходы и анимации без необходимости использования JavaScript или Flash. Основное назначение CSS-анимаций — улучшение пользовательского интерфейса, добавление визуальных эффектов и привлечение внимания к определённым элементам на странице.

CSS-анимации позволяют задавать ключевые кадры (keyframes), которые определяют, как элемент будет изменяться в течение времени. Это делает анимации более гибкими и мощными, чем простые переходы, которые применяются к изменениям свойств.

Плавные переходы между состояниями элементов.

Возможность создания сложных анимаций с использованием ключевых кадров.

Улучшение пользовательского опыта и привлечение внимания.

HTML (index.html)

Объявление типа документа:

<!DOCTYPE html>

Указывает, что документ является HTML5. Это важно для браузеров, чтобы правильно интерпретировать и отображать содержимое страницы.

Начало HTML-документа:

<html lang="ru">

Открывает тег <html> и задает атрибут lang, указывая, что язык документа — русский. Это помогает поисковым системам и вспомогательным технологиям правильно обрабатывать контент.

Секция <head>:

<head>

Начинает секцию <head>, где размещаются метаданные, ссылки на стили и скрипты.

Кодировка документа:

<meta charset="UTF-8">

Устанавливает кодировку символов на UTF-8, что позволяет корректно отображать текст на разных языках.

Настройки для адаптивного дизайна:

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

Устанавливает параметры для адаптивного дизайна, позволяя странице корректно отображаться на мобильных устройствах.

Заголовок страницы:

<title>Зоомагазин - Ваш питомец заслуживает лучшего</title>

Определяет заголовок страницы, который отображается на вкладке браузера и в результатах поиска.

Подключение шрифтов:

<link href="https://fonts.googleapis.com/css2?family=Roboto:wght@300;400;700&display=swap" rel="stylesheet">

Подключает шрифт Roboto из Google Fonts, который будет использоваться на странице для улучшения визуального восприятия текста.

Подключение CSS-файла:

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

Подключает внешний файл стилей styles.css, который содержит все стили для оформления страницы.

Начало тела документа:

<body>

Открывает тег <body>, который содержит весь видимый контент страницы.

Хедер с заголовком и описанием:

<header class="hero">

<div class="container">

<h1>Зоомагазин</h1>

<p>Все для ваших любимых питомцев: качество, забота и радость в каждой покупке</p>

</div>

Создает заголовок с названием магазина и кратким описанием, помещая их в контейнер для центрирования и стилизации.

Плавающие элементы (изображения):

<img src="/placeholder.svg?height=50&width=50" alt="Лапка" class="floating-element" style="top: 10%; left: 10%;">

Добавляет изображение с классом floating-element, которое будет анимироваться (плавать) на странице. Атрибут style задает его начальное положение.

Вторая плавающая иконка:

<img src="/placeholder.svg?height=40&width=40" alt="Косточка" class="floating-element" style="top: 20%; right: 15%;">

Добавляет еще одно плавающее изображение с другим начальным положением.

Третья плавающая иконка:

<img src="/placeholder.svg?height=60&width=60" alt="Миска" class="floating-element" style="bottom: 15%; left: 20%;">

Добавляет третье плавающее изображение, которое также будет анимироваться.

Секция "Почему выбирают нас":

<section class="info-section">

<div class="container">

<h2>Почему выбирают нас</h2>

<p>Широкий ассортимент, профессиональные консультации и любовь к животным</p>

</div>

</section>

Создает информационную секцию, объясняющую преимущества магазина.

Контейнер для сетки товаров:

<div class="container">

<div class="grid">

Создает контейнер для сетки, в которой будут размещены товары и акции. Это позволяет организовать контент в виде сетки.

Первый элемент сетки - акция:

<div class="grid-item">

<h3>Не пропустите!</h3>

<div class="promo">Акция 1+1</div>

<p>Купите один товар и получите второй в подарок</p>

Описывает акцию, которая привлекает внимание пользователей. Заголовок и описание делают акцию более заметной.

Кнопка "Подробнее" для акции:

<div class="button-container">

<a href="#" class="button" data-info="Акция действует до конца месяца">Подробнее</a>

</div>

Создает кнопку, которая ведет к дополнительной информации об акции. Атрибут data-info может использоваться для отображения подсказки или дополнительной информации.

Второй элемент сетки - категория "Корма":

<div class="grid-item">

<div class="category-icon">🍖</div>

<h3>Корма</h3>

<p>Широкий выбор кормов для всех видов животных</p>

Описывает категорию товаров — корма. Иконка с изображением корма делает элемент более визуально привлекательным.

Кнопка "Смотреть" для кормов:

<div class="button-container">

<a href="#" class="button" data-info="Более 1000 наименований">Смотреть</a>

</div>

Создает кнопку для перехода к разделу с кормами. Атрибут data-info предоставляет дополнительную информацию о количестве наименований.

Третий элемент сетки - категория "Аксессуары":

<div class="grid-item">

<div class="category-icon">🎾</div>

<h3>Аксессуары</h3>

<p>Игрушки, поводки, миски и многое другое</p>

Описывает категорию аксессуаров для животных, включая игрушки и другие предметы.

Кнопка "Смотреть" для аксессуаров:

<div class="button-container">

<a href="#" class="button" data-info="Товары для игр и комфорта">Смотреть</a>

</div>

Создает кнопку для перехода к разделу с аксессуарами, предоставляя информацию о типах товаров.

Четвертый элемент сетки - категория "Наполнители":

<div class="grid-item">

<div class="category-icon">🐾</div>

<h3>Наполнители</h3>

<p>Качественные наполнители для кошачьих туалетов</p>

Описывает категорию наполнителей для туалетов, что также важно для владельцев домашних животных.

Кнопка "Смотреть" для наполнителей:

<div class="button-container">

<a href="#" class="button" data-info="Эко-френдли варианты">Смотреть</a>

</div>

Создает кнопку для перехода к разделу с наполнителями, акцентируя внимание на экологически чистых вариантах.

Секция "Наши преимущества":

<section class="info-section">

<div class="container">

<h2>Наши преимущества</h2>

<p>Профессиональный подход, быстрая доставка и поддержка 24/7</p>

</div>

</section>

Завершает основную часть страницы, подчеркивая ключевые преимущества магазина, такие как профессионализм и доступность поддержки.

Футер с контактной информацией:

<footer>

<div class="container">

<div class="grid">

Открывает секцию футера, где размещается контактная информация и адрес магазина.

Контактная информация:

<div class="grid-item">

<h3>Контакты</h3>

<p>Телефон: +7 (XXX) XXX-XX-XX</p>

<p>Email: info@zoomagaz.ru</p>

Этот элемент футера содержит заголовок "Контакты" и предоставляет информацию о способах связи с магазином, включая телефон и электронную почту.

Адрес магазина:

<div class="grid-item">

<h3>Адрес</h3>

<p>ул. Примерная, 123</p>

<p>Время работы: 9:00 - 21:00</p>

Этот элемент футера содержит заголовок "Адрес" и предоставляет физический адрес магазина, а также часы работы.

Закрытие контейнера футера:

</div>

</div>

</footer>

Закрывает контейнер футера и сам футер, завершая структуру страницы.

Подключение JavaScript:

<script src="script.js"></script>

Подключает внешний JavaScript-файл script.js, который содержит функциональность для интерактивности на странице.

Закрытие тега <body>:

</body>

Закрывает тег <body>, завершив основной контент страницы.

Закрытие тега <html>:

</html>

Закрывает тег <html>, завершая документ

Общая структура и функциональность

Структура: Код организован в логические секции, включая заголовок, основное содержимое и футер. Это делает его удобным для восприятия и редактирования.

Адаптивный дизайн: Использование контейнеров и сеток позволяет странице адаптироваться к различным размерам экранов, что важно для мобильных пользователей.

Интерактивные элементы: Кнопки и ссылки имеют атрибуты data-info, которые могут использоваться для отображения дополнительной информации, что улучшает пользовательский опыт.

Визуальные элементы: Плавающие изображения добавляют динамичности и визуального интереса к странице, что может привлечь внимание пользователей.

Контактная информация: Футер содержит важные данные для пользователей, которые хотят связаться с магазином или посетить его физически.

Заключение

Код HTML, является основой для создания современного и функционального веб-сайта зоомагазина. Он демонстрирует лучшие практики веб-разработки и включает в себя множество элементов, которые способствуют улучшению пользовательского опыта и визуальной привлекательности сайта.

Структурированность и семантика:

Код организован в логические секции, такие как <header>, <section>, и <footer>, что делает его легко читаемым и поддерживаемым. Использование семантических тегов, таких как <h1>, <h2>, и <p>, помогает поисковым системам и вспомогательным технологиям лучше понимать содержание страницы. Это также улучшает доступность для пользователей с ограниченными возможностями.

Адаптивный дизайн:

Включение метатега <meta name="viewport"> и использование контейнеров и сеток позволяет сайту адаптироваться к различным размерам экранов. Это особенно важно в современном мире, где все больше пользователей обращаются к интернету через мобильные устройства. Адаптивный дизайн обеспечивает удобство навигации и чтения контента на любых устройствах.

Интерактивные элементы:

Код включает кнопки и ссылки с атрибутами data-info, которые могут использоваться для отображения дополнительной информации. Это создает интерактивный опыт для пользователей, позволяя им получать больше информации о товарах и акциях, не покидая текущую страницу. Интерактивные элементы, такие как кнопки "Подробнее" и "Смотреть", способствуют вовлечению пользователей и могут увеличить конверсию.

Визуальные элементы:

Плавающие изображения добавляют динамичности и визуального интереса к странице. Эти элементы могут привлекать внимание пользователей и создавать ощущение живости и активности на сайте. Использование иконок для категорий товаров делает интерфейс более интуитивно понятным и помогает пользователям быстро находить нужные разделы.

Контактная информация:

Футер содержит важные данные, такие как телефон, электронная почта и адрес магазина, что позволяет пользователям легко связаться с магазином или посетить его. Это создает доверие и прозрачность, что особенно важно для онлайн-магазинов, где пользователи могут быть обеспокоены безопасностью и надежностью.

SEO и производительность:

Структурированный и семантически правильный код способствует лучшему SEO (поисковой оптимизации), что может помочь сайту занять более высокие позиции в результатах поиска. Это, в свою очередь, может привести к увеличению трафика и потенциальных клиентов. Кроме того, использование внешних файлов CSS и JavaScript помогает оптимизировать загрузку страницы, так как браузеры могут кэшировать эти файлы.

Возможности для расширения:

Код легко расширяем. В будущем можно добавить новые секции, такие как блоги, отзывы клиентов или разделы с акциями. Это позволяет адаптировать сайт к меняющимся потребностям бизнеса и пользователей.

Заключение

В целом, представленный код HTML создает прочную основу для веб-сайта зоомагазина, который не только информирует пользователей о товарах и услугах, но и обеспечивает удобный и приятный опыт взаимодействия. Он сочетает в себе элементы дизайна, функциональности и доступности, что делает его подходящим для современного онлайн-торгового пространства. С учетом всех этих факторов, данный код может служить отличным стартом для создания успешного веб-сайта, который будет привлекать и удерживать клиентов, способствуя росту бизнеса.