ДИСЦИПЛИНА	Фронтенд и бэкенд разработка
ИНСТИТУТ	Институт перспективных технологий и индустриального программирования
КАФЕДРА	Кафедра индустриального программирования
ВИД УЧЕБНОГО МАТЕРИАЛА	Практические занятия
ПРЕПОДАВАТЕЛЬ	Загородних Николай Анатольевич
СЕМЕСТР	1 семестр, 2024-2025 гг.

Практическое занятие 12. Встраивание JavaScript-скриптов в HTML-документ: основные подходы и методы

Изучаемые вопросы

- Основные подходы к встраиванию скриптов в HTML-документ.
- Различия между добавлением скриптов в <head> и <body>.
- Использование атрибутов defer и async.

Краткая теория

Скрипты можно встраивать в HTML-документ несколькими способами:

1. Внутренние скрипты размещение JavaScript-кода внутри тега <script> в HTML.

2. Внешние скрипты подключение JavaScript-файлов через атрибут src в теге <script>.

3. Порядок загрузки скрипты можно размещать в <head> или в конце <body>. Скрипты, размещенные в <head>, могут блокировать рендеринг страницы, если не использовать атрибуты defer или async.

UTM-метки (Urchin Tracking Module) — это параметры, которые добавляются к URL для отслеживания эффективности рекламных кампаний. Они позволяют аналитическим системам (например, Яндекс.Метрика или Google Analytics) собирать данные о том, откуда пришел пользователь, и какие действия он совершает на сайте.

Основные параметры UTM-меток:

- utm_source - источник трафика (например, yandex, google).
- utm_medium - тип трафика (например, cpc, email).
- utm_campaign - название рекламной кампании.
- utm_term - ключевое слово или поисковый запрос.
- utm_content - дополнительная информация о контенте (например, A/B тестирование).

Пример URL с UTM-метками:

https://example.com?utm_source=yandex&utm_medium=cpc&utm_campaign=spring_sale&utm_term=сапоги

Допустим, ваш локальный сайт доступен по адресу http://localhost:3000. Чтобы добавить UTM-метки, вы можете использовать следующий формат:

http://localhost:3000?utm_source=yandex&utm_medium=cpc&utm_campaign=spring_sale&utm_term=сапоги

Полезные источники информации

1. Подключение и выполнение JavaScript

https://javascript.ru/tutorial/foundation/start?ysclid=m2yw19akv8286512177

2. Функции JS

https://developer.mozilla.org/ru/docs/Web/JavaScript/Reference/Functions

3. JavaScript ссылки на ошибки

https://developer.mozilla.org/ru/docs/Web/JavaScript/Reference/Errors

Тренажеры

https://codepen.io/pen

Задание

Часть 1. Создание файла script.js

В этом файле создайте четыре функции:

- Функция showMessage принимает строку в качестве аргумента и выводит это сообщение в консоль. Например, вызов showMessage("Скрипт загружен!") должен отобразить это сообщение в консоли браузера.

- Функция changeBackgroundColor принимает цвет (например, "red", "ff0000") в качестве аргумента и изменяет цвет фона страницы на указанный цвет.

- Функция toggleVisibility принимает селектор элемента (например, .content) и переключает его видимость, т.е. если элемент виден, он становится скрытым, и наоборот.

- Функция, которая будет:

  - Извлекать значение параметра utm_term из URL.

  - Если параметр utm_term присутствует, заменять текст H1 на значение этого параметра.

  - Если параметр отсутствует, оставить текст H1 по умолчанию.

Часть 2. Подключение скрипта script.js к HTML-документу

- Сначала подключите скрипт внутри тега <head>, используя атрибут defer. Это позволит браузеру загружать скрипт параллельно с загрузкой HTML-документа, но выполнять его только после полной загрузки страницы.

- Затем подключите тот же скрипт перед закрывающим тегом </body>. Это обеспечит выполнение скрипта после загрузки всего содержимого страницы.

Часть 3. Порядок вызова функций

После подключения скрипта, вызовите функции в следующем порядке:

1. Вызовите showMessage с сообщением "Скрипт загружен!" сразу после его определения, чтобы убедиться, что скрипт работает.

2. Затем вызовите changeBackgroundColor, передав ей цвет фона, который вы хотите установить (например, "lightblue"). Убедитесь, что этот вызов происходит после того, как страница полностью загружена. Для этого используйте обработчик события DOMContentLoaded.

3. Наконец, используйте toggleVisibility, чтобы переключить видимость элемента с классом .content. Убедитесь, что элемент с этим классом существует на вашей странице. Этот вызов также должен быть внутри обработчика события DOMContentLoaded.

Часть 4. Дополнительные функции

- Добавьте функцию logCurrentTime, которая выводит текущее время в консоль в формате "ЧЧ:ММ:СС". Вызовите эту функцию сразу после showMessage.

- Создайте функцию resetBackgroundColor, которая будет возвращать цвет фона к исходному значению (например, "white"). Вызовите эту функцию перед вызовом changeBackgroundColor, чтобы увидеть эффект смены цвета.

Критерии оценивания

1. Синтаксис (30%): Правильное использование синтаксиса JavaScript и методов работы с DOM.

2. Встраивание скриптов (30%): Корректное подключение внешнего скрипта и использование атрибутов defer.

3. Работа с функциями (20%): Реализация функций и обработчиков событий.

4. Логика программы (20%): Корректность выполнения задания и логика программного кода.

Форма сдачи

Студенты должны представить свой код в виде файлов .html .css .js (в случае подключения внешних скриптов), который можно открыть в браузере. Также важно прокомментировать код, чтобы объяснить, что делает каждый элемент.