

Работа с DOM в JavaScript

Выполнен на 0%

Введение

Общая теория

Дополнительные возможности синтаксиса JavaScript

Обработка данных

Использование сторонних библиотек

Углублённая теория

Методика работы с DOM

Кейс 1, лёгкий уровень

Кейс 2, лёгкий уровень

Кейс 3, лёгкий уровень

Кейс 4, лёгкий уровень

Кейс 5, средний уровень

Кейс 6, средний уровень

Кейс 7, средний уровень

Кейс 8, сложный уровень

Кейс 9, сложный уровень

Кейс 10, сложный уровень

Главная / Моё обучение / Работа с DOM в JavaScript / Общая теория /

Использование сторонних библиотек

Иногда дешевле и проще подключить стороннюю библиотеку к сайту, чем изобретать что-то своё. Например, библиотеку для работы с датами [Day.js](#), библиотеку для построения графиков и диаграмм или библиотеку, которая добавляет нужное окончание числительному.

Но при выборе готового решения не забывайте смотреть на некоторые параметры, а именно:

— когда библиотека обновлялась в последний раз (поддерживают ли разработчики свой продукт, правят ли баги),

— большой вес библиотеки может замедлять загрузку скриптов,

— важно наличие документации, в которой описывается, как вызывать методы, хорошо, чтобы были примеры.

Сторонние библиотеки ещё называют вендорными.

Есть несколько способов подключить библиотеку к проекту.

Самый простой — подключать библиотеку как обычный js-файл, используя тег `script`.

Разберём этот способ на примере. Подключим к нашему сайту стороннюю библиотеку для правильного склонения числительных `get-declension`. Добавим папку `vendor` и поместим в неё файл `get-declension.min.js`, в котором находится исходный код библиотеки. В `HTML` перед закрывающим тегом `body` добавим строку подключения js-файла. Как правило, сторонние библиотеки подключают перед js-файлами с пользовательским кодом.

```
...
<script src="/base/accessory-03/js/vendor/get-declension.min.js"></script>
<script src="js/script.js"></script>
</body>
```

После этого в коде можно вызывать функции из подключенной библиотеки:

```
const length = 5;

const declension = window.getDeclension({count: length, one: 'ступенька', few: 'ступеньки', many: 'ступенек'}));

console.log(` ${length} ${declension}`);
// выведет 5 ступенек
```

При таком способе переменная попадает в глобальную область видимости `window` и нужно быть осторожными, так как можно перетереть другие переменные в `window`. Для изоляции переменных используют импорт модулей (оператор `import`). Но об этом и о других вариантах подключения сторонних библиотек мы расскажем в других навыках.

Ознакомились со статьёй?

Сохранить прогресс

Обработка данных

Углублённая теория



Практикум

Тренажёры

Подписка

Для команд и компаний

Учебник по PHP

Профессии

Фронтенд-разработчик

React-разработчик

Фулстек-разработчик

Бэкенд-разработчик

Услуги

Работа наставником

Для учителей

Стать автором

Курсы

HTML и CSS. Профессиональная вёрстка сайтов

HTML и CSS. Адаптивная вёрстка и автоматизация

JavaScript. Профессиональная разработка веб-интерфейсов

JavaScript. Архитектура клиентских приложений

React. Разработка сложных клиентских приложений

PHP. Профессиональная веб-разработка

PHP и Yii. Архитектура сложных веб-сервисов

Node.js. Разработка серверов приложений и API

Анимация для фронтендеров

Вёрстка email-рассылок

Vue.js для опытных разработчиков

Регулярные выражения для фронтендеров

Шаблонизаторы HTML

Алгоритмы и структуры данных

Анатомия CSS-каскада

Блог

С чего начать

Шпаргалки для разработчиков

Отчеты о курсах

Информация

Об Академии

О центре карьеры

Остальное

Написать нам

Мероприятия

Форум