

Работа с DOM в JavaScript

Выполнен на 0%

Введение

Общая теория

Углублённая теория

Что же такое DOM?

Поиск элементов в DOM

Навигация по элементам

Работа со свойствами и содержимым DOM элементов

Создание и наполнение DOM элементов

Реализация плавной прокрутки на JavaScript

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 / Углублённая теория /

Реализация плавной прокрутки на JavaScript

Плавную прокрутку страницы до нужного элемента сейчас можно встретить на большом количестве сайтов. Есть несколько способов её реализовать, например, через CSS-свойство `scroll-behavior`. Покажем, как это можно сделать на *JavaScript* с помощью метода `scrollIntoView()`. Метод прокручивает страницу к элементу, на котором он был вызван.

Как обычно, разберём всё на примере.

У нас есть статья, в которой встречаются ссылки на сноски. Внизу статьи в разделе *Примечание* собраны определения для сносок.

`<p class="paragraph">Количество зарегистрированных на Московской бирже индивидуальных инвестиционных счетов по состоянию на 6 апреля 2020 года достигло`

...

`<h2>Примечания</h2>`
`<p class="footnotes-item" id="anchor1">На начало апреля на Московской бирже зарегистрировано два миллиона индивидуальных инвестиционных счетов (ИИС), открытых физическими лицами. Всего на бирже 4,57 млн человек имеют брокерские счета. На начало года было открыто 1,65 млн ИИС, их число за первый квартал увеличилось на 407 тыс. На долю брокерских счетов приходится 87% ИИС, счета доверительного управления (ИИС-ДУ) составляют 13%.</p>`
...

Добавим плавные переходы от ссылок на тексты с определением. Для этого в событии `click` у ссылки на сноску будем получать идентификатор якорного элемента (атрибут `id`). Находить этот элемент `document.querySelector` и вызывать для него метод `scrollIntoView()`. В параметрах метода укажем `{behavior: 'smooth', block: 'start'}`, что означает плавную анимацию скролла с вертикальным выравниванием равным значению `start`. Чтобы не переходить по «битым» ссылкам, перед переходом будем проверять элемент на `null`.

```
const anchor = document.querySelector('.footnote-label');

const elementClickHandler = (evt) => {
  const element = evt.target.closest('a');

  if (!element) {
    return;
  }

  // отменяем действие по умолчанию
  evt.preventDefault();

  // получаем идентификатор элемента для перехода
  const blockId = element.getAttribute('href');

  if (blockId && blockId !== '#') {
    // находим элемент для перехода
    const block = document.querySelector(blockId);

    if (block) {
      // плавно переходим к элементу
      block.scrollIntoView({
        behavior: 'smooth',
        block: 'start',
      });
    }
  }
};

anchor.addEventListener('click', elementClickHandler);
```

Метод `scrollIntoView()` имеет хорошую браузерную поддержку, о чем говорит [Can I Use](#). Более подробное описание метода можно найти в [спецификации](#).

Чтобы отменить для ссылок действие по умолчанию (переход по указанному URL), мы использовали метод `preventDefault()` у объекта *Событие*. Подробнее о методе `preventDefault()` можно прочитать в нашем блоге в статье «[Зачем нужен метод preventDefault](#)».

Ознакомились со статьёй?

Сохранить прогресс

Создание и наполнение DOM элементов

Методика работы с DOM



Практикум

Тренажёры

Подписка

Для команд и компаний

Учебник по PHP

Профессии

Фронтенд-разработчик

Реакт-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