🗔 Кейс 7, средний уровень

🖸 Кейс 8, сложный уровень

🖸 Кейс 9, сложный уровень

🖸 Кейс 10, сложный уровень

html academy

Главная / Моё обучение / Работа с DOM в JavaScript / Методика работы с DOM , Изменение свойств элементов страницы

Вот несколько типовых ситуаций.

■ Практикум ▼ 😚 Профессии ▼ У Курсы ▼ У Моё обучение ▼

Первая: пользователь заполняет форму опросника. В пункте *«Скажите, откуда Вы узнали о волонтёрском проекте* в нашей организации?» ему предлагается выбрать только один из вариантов ответа, также есть вариант «Другое», который делает доступным для изменения текстовое поле. По умолчанию текстовое поле заблокировано для ввода.

🚱 Виталий Ризун 🔻

```
Скажите, откуда Вы узнали о волонтёрском проекте
           в нашей организации?
              Из СМИ
               Рассказали друзья/знакомые
               Из поисковых систем
              На сайте организации
              Увидел(а) акцию организации
              Другое
Форма с вариантом Другое
```

вариант *«Друг*ое», снимать атрибут **disabled** с текстового поля, иначе добавлять этот атрибут. Нам понадобится обработчик события **change** у формы. На ней через делегирование будет отлавливать изменение состояния

радиокнопок. Но прежде надо найти элементы. Ищем форму и текстовое поле, для которого будем убирать/ добавлять атрибут disabled. Разметка для опросника: <form class="feedback" action="#" method="POST"> <fieldset>

Получается, что при изменении отметки в группе радиокнопок нужно выяснить, какая из них отмечена. Если это

<legend>Скажите, откуда Вы узнали о волонтёрском проекте в нашей организации?</legend> <label> <input type="radio" name="source" id="news" value="news" checked> Из СМИ

// нашли поле для ввода текста

const textField = document.querySelector('#other-answer');

</label>

```
<label>
       <input type="radio" name="source" id="friends" value="friends">
       Рассказали друзья/знакомые
     </label>
     <label>
       <input type="radio" name="source" id="search" value="search">
       Из поисковых систем
     </label>
     <label>
       <input type="radio" name="source" id="site" value="site">
       На сайте организации
     </label>
     <label>
       <input type="radio" name="source" id="share" value="share">
       Увидел(а) акцию организации
     </label>
     <label>
       <input type="radio" name="source" id="other" value="other">
       Другое
     </label>
     <textarea name="other-answer" id="other-answer" disabled>
     </textarea>
   </fieldset>
 </form>
Код с решением:
 // нашли форму
 const feedbackForm = document.querySelector('.feedback');
```

```
const formChangeHandler = (evt) => {
   // нас интересует только изменение состояний радиокнопок
   const item = evt.target.closest('input[type="radio"]');
   if (!item) {
     return;
   // если изменяется состояние радиокнопки Другое и она становится отмеченной, убираем атрибут disabled
   textField.toggleAttribute('disabled', item.id !== 'other' || !item.checked);
 // подписались на событие изменение формы (через него будем отлавливать изменение радиокнопок)
 feedbackForm.addEventListener('change', formChangeHandler);
Другая ситуация.
В разделе добавлена внутренняя навигация — список ссылок и закладок. За счёт неё можно сэкономить место
на странице. Переключение между закладками выполняет JavaScript-код. Активная ссылка может быть выделена
с помощью специального CSS-класса, например, item-active. При клике на ссылку будем добавлять ей этот
```

закладку также выделяем специальным классом — page-active.

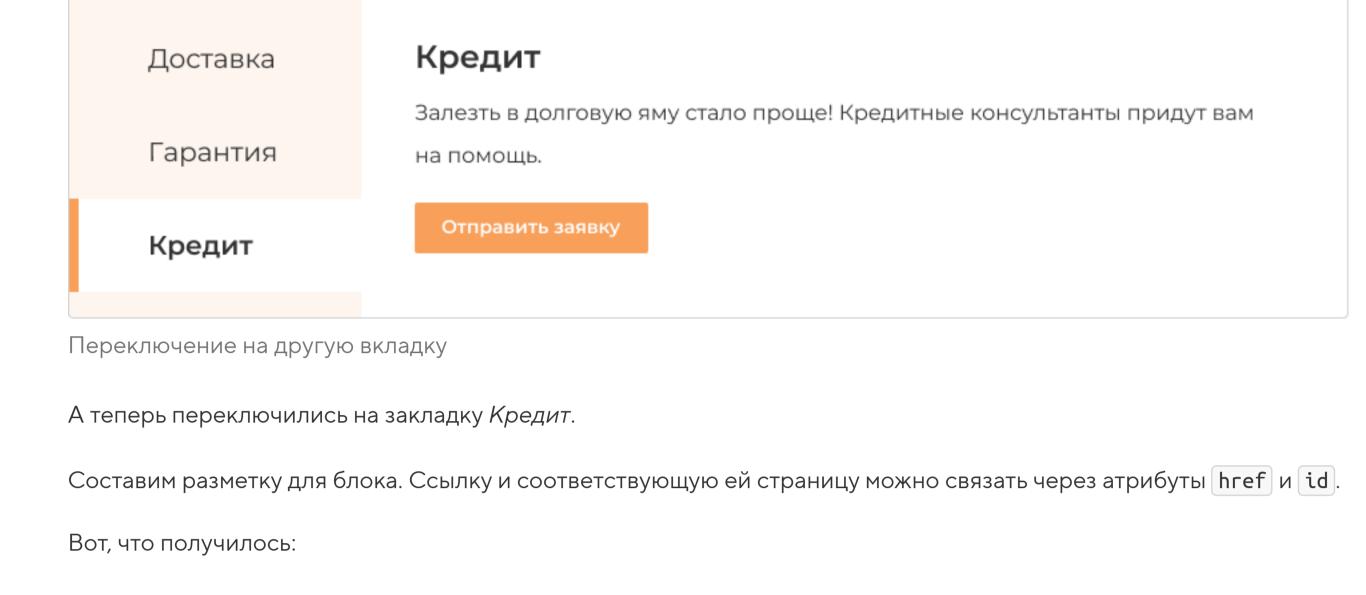
Внутренняя навигация на странице

Активна закладка Гарантия.

Гарантия Доставка Если купленный у нас товар поломается или заискрит, а также в случае Гарантия пожара, спровоцированного его возгоранием, вы всегда можете быть уверены в нашей гарантии. Мы обменяем сгоревший товар на новый. Дом уж восстановите как нибудь сами. Кредит

класс, при этом убирать его с других ссылок. Скрыть закладки также можно за счёт CSS. Допустим, по умолчанию

закладки не показываем, то есть display: none; , а для активной устанавливаем display: block; . Активную



Гарантия

Доставка

Кредит

<div class="services">

```
<div class="services-content">
     <section class="page" id="page-1">
       <h3>Доставка</h3>
       <р>Мы с удовольствием доставим ваш товар прямо к вашему подъезду совершенно бесплатно! Ведь мы
 неплохо заработаем, поднимая его на ваш на этаж!
     </section>
     <section class="page" id="page-2">
       <h3>Гарантия</h3>
       Если купленный у нас товар поломается или заискрит, а также в случае пожара, спровоцированного
 его возгоранием, вы всегда можете быть уверены в нашей гарантии. Мы обменяем сгоревший товар на новый.
 Дом уж восстановите как нибудь сами.
     </section>
     <section class="page page-active" id="page-3">
       <h3>Кредит</h3>
       <р>Залезть в долговую яму стало проще! Кредитные консультанты придут вам на помощь.
       <a href="#">Отправить заявку</a>
     </section>
   </div>
 </div>
Опишем стили:
 .services-navigation-link {
```

padding: 10px 15px;

color: inherit;

/* активная ссылка */

text-decoration: none;

.services-navigation-link.item-active {

const pages = document.querySelectorAll('.page');

const setActivePage = (anchor) => {

navItems.forEach((navItem) => {

if (!link) {

return;

/* по умолчанию страница скрыта */

```
.services-content section {
   display: none;
 /* активная страница */
 .services-content .page-active {
   display: block;
B JavaScript-коде найдём элемент с навигацией. Также найдём элементы с ссылками и элементы со страницами.
Для элемента с навигацией добавим обработчик события click (снова используем делегирование, чтобы
не создавать много обработчиков событий). А дальше всё просто: при клике на ссылку добавляем ей класс item-
active и удаляем его у других ссылок. С помощью метода getAttribute узнаем идентификатор страницы,
которая станет активной. По этому идентификатору добавляем класс page-active для будущей активной
страницы и удаляем его у других страниц.
 // Поиск элементов
 // Навигация
 const nav = document.querySelector('.services-navigation');
 // отдельные ссылки в навигации
 const navItems = document.querySelectorAll('.services-navigation-link');
 // элементы со страницами
```

navItem.classList.toggle('item-active', anchor === navItem.getAttribute('href')); }); // показываем страницу (переключаем стили), на которую ссылается пункт меню pages.forEach((page) => { page.classList.toggle('page-active', anchor === ('#' + page.id)); const navClickHandler = (evt) => { // используем делегирование и обрабатываем Click только на ссылках в навигации const link = evt.target.closest('a');

// делаем активным пункт меню (переключаем стили), ориентир - значение атрибута href

// получаем значение якоря, по нему будем открывать соответствующую страницу

```
const anchor = link.getAttribute('href');
   // переключение страницы по значению якоря
   setActivePage(anchor);
 const init = () => {
   // подписываемся на событие Click в навигации
   nav.addEventListener('click', navClickHandler);
 // начальная инициализация элементов страницы
 init();
Ещё одна типовая ситуация: карточки с объёмным текстовым содержимым и ссылкой Показать полностью....
                           HTML Academy
                           11 фев в 14:35
                  Стоит ли делать слайдеры на CSS?
                  Что такое слайдеры:
                  Обычно это блок на странице, в пределах которого пользователю показывают
                  статьи, новости, изображения или другой контент с определённой
                  периодичностью. Например, когда на сайте вам показывают друг за другом
                  несколько баннеров с акциями. Давайте разберёмся, как стоит их делать.
                   Показать полностью...
```

СТОИТ ЛИ ДЕЛАТЬ > CЛАЙДЕРЫ HA CSS?

_0

ПОДПИСАТЬСЯ

→ ПОДЕЛИТЬСЯ = COXPAHUТЬ ...

HTML Academy 27,1 тыс. подписчиков В прямом эфире провели разбор тестового задания NKH Studio вместе с фронтендразработчиком компании Витольдом Демишкевичем.

6:45 Зачем даётся тестовое задание в NKH Studio

8:23 Из чего состоит тестовое задание NKH Studio

50:03 Как готовиться к собеседованиям

<div class="description description-short">

компании Витольдом Демишкевичем.

/* класс для свёрнутого описания */

content.addEventListener('click', moreLinkClickHandler);

Если не вдаваться в подробности, то схема работы во всех описанных случаях одинаковая:

.description-short { overflow: hidden;

max-height: 60px;

37:47 Опыт Витольда в выполнении тестовых заданий

46:30 Что раньше — тестовое задание или собеседование

13:33 На что Витольд обращает внимание при проверке выполненных работ

ЕЩЁ

Элемент Ещё

1 065 просмотров • Прямой эфир состоялся 18 февр. 2021 г.

Что ждать от тестового задания фронтенд-разработчика

Элемент Показать полностью... на странице

#неглупые_вопросы

Или так: подробную информацию можно показать, а затем снова свернуть.

0:00 / 1:28:54 • Наливаем чай, ждём подключения 🗲

```
0:00 / 1:28:54 • Наливаем чай, ждём подключения 🗲
Что ждать от тестового задания фронтенд-разработчика
1 065 просмотров • Прямой эфир состоялся 18 февр. 2021 г.
                                                                           HTML Academy
                                                                                                 подписаться
        27,1 тыс. подписчиков
        В прямом эфире провели разбор тестового задания NKH Studio вместе с фронтенд-
        разработчиком компании Витольдом Демишкевичем.
        Поговорили с ним также о том:
        - Какие навыки необходимы для выполнения тестового задания;
        - На что обращают внимание работодатели при проверке тестовых заданий;
        - Как не провалить собеседование.
        Раскрыть тему и ответить на вопросы зрителей помогал Серёжа Попов — руководитель
        фронтенд-аутсорса «Лига А.»
        Все необходимые навыки для трудоустройства можно получить с помощью обновлённой
        профессии «Фронтенд-разработчик». Курс начинается 17 мая, но записаться на профессию
        можно уже сейчас — https://tml.io/ovr6h
        Тайм-коды:
        0:00 Наливаем чай, ждём подключения
        0:09 Действительно начинаем. Знакомимся
        3:40 Зачем нужны тестовые задания
```

54:01 Про работу и собеседования на удалёнке 59:45 Что не стоит делать на собеседованиях 1:05:30 Специфика трудоустройства в NKH Studio 1:13:03 Вопросы по трудоустройству джунов 1:26:32 С чего начать учиться 1:28:10 Прощаемся

Элемент Свернуть

Итак, разметка:

</div>

Стили:

Код:

<div class="content">

СВЕРНУТЬ

```
<a class="more" href=""#>Ещё</a>
</div>
.content {
 padding: 20px 0;
 width: 570px;
```

Реализуем второй вариант, то есть будем скрывать и показывать полное описание по клику на ссылку Ещё/

с соответствующими свойствами. При клике на ссылку будет переключаться этот класс и дополнительно меняться

В прямом эфире провели разбор тестового задания NKH Studio вместе с фронтенд-разработчиком

Свернуть. Скрыть описание можно за счёт стилей — установить максимальную высоту блока, а всё, что

не поместилось, обрезать при помощи overflow: hidden; . Опишем CSS-класс description-short

текст ссылки, в зависимости от того, какое сейчас описание — полное или сокращённое.

const content = document.querySelector('.content'); const moreLinkClickHandler = (evt) => { // отлавливаем событие только на ссылке с классом тоге const element = evt.target.closest('a.more'); if (!element) { return; // отменяем для ссылки действие по умолчанию evt.preventDefault(); const descriptionElement = document.querySelector('.description'); if (descriptionElement) { // переключаем класс descriptionElement.classList.toggle('description-short'); // меняем текст ссылки element.textContent = descriptionElement.classList.contains('description-short') ? 'Ещё' : 'Свернуть';

```
— затем изменение свойств у элемента (это может быть добавление/удаление класса, добавление/удаление
  атрибута, изменение содержимого элемента и так далее)
    Ознакомились со статьёй?
      Сохранить прогресс
```

Практикум

Стать автором

Тренажёры

сначала — поиск элемента,

< Методика работы с DOM

Курсы

Шаблонизаторы HTML

Анатомия CSS-каскада

© ООО «Интерактивные обучающие технологии», 2013-2022 **⟨/⟩ → ♥ → !!!**

Алгоритмы и структуры данных

Конфиденциальность Сведения об образовательной организации Лицензия № 3026

HTML и CSS. Профессиональная вёрстка сайтов

Добавление новых элементов в DOM >

Шпаргалки для разработчиков

Блог

С чего начать

Отчеты о курсах

Информация

Об Академии

Форум