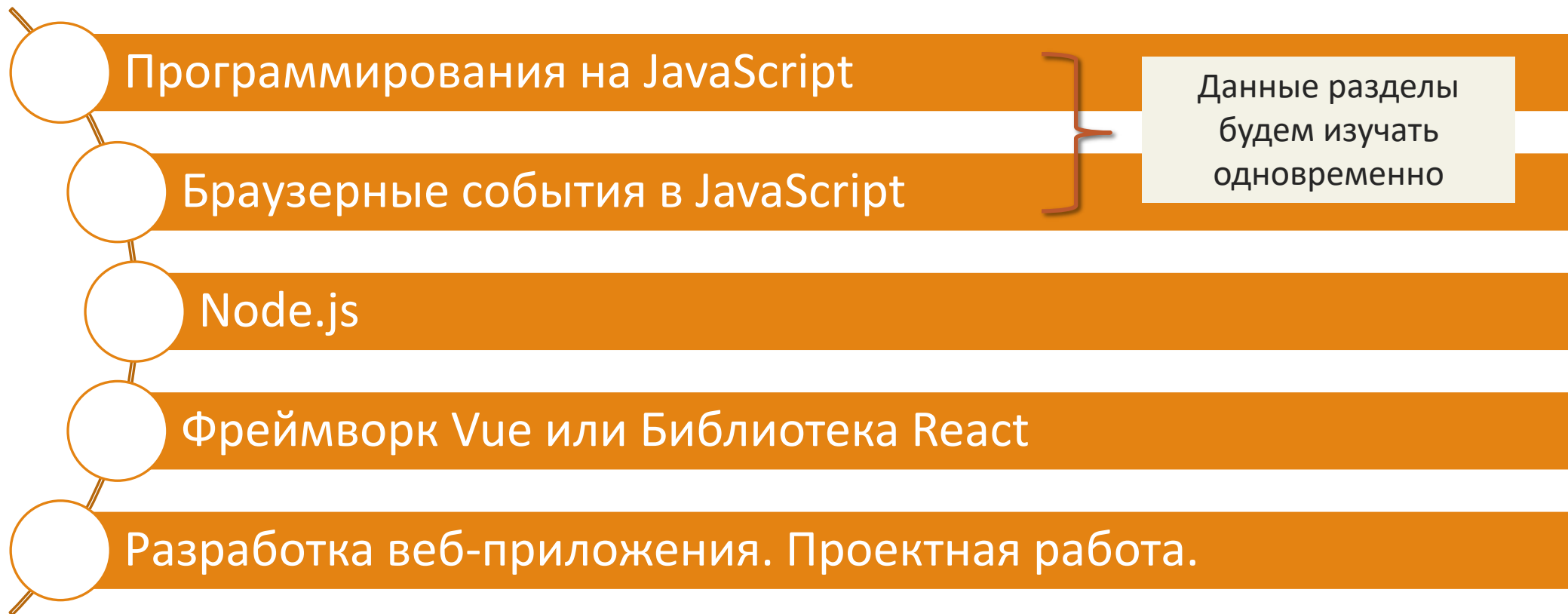


JAVASCRIPT

Javascript (сокращённо JS) – это **язык программирования**, который изначально был придуман для браузера, чтобы придать страницам **интерактивность и динамичность**.
Программы написанные на javascript называются **сценариями** или **скриптами**.

Будем корректировать скорость работы
относительно уровня усвоения

Примерный план изучения JS



Источники информации о JS

1. <https://learn.javascript.ru/js>
2. <https://itchief.ru/javascript/>
3. <https://html5css.ru/js/default.php>
4. <https://dmitrylavrik.ru/video/javascript>
5. <https://proity.ru/programming/free-js-courses/>
6. Ссылка на текстовый учебник JS
[https://drive.google.com/drive/folders/1MxVAaNI336fGYtB-bA3t1r_swx5fjGPQ?usp=share link](https://drive.google.com/drive/folders/1MxVAaNI336fGYtB-bA3t1r_swx5fjGPQ?usp=share_link)

Общие правила работы

- вся информация по курсу на сервере Discord. Присоединяйтесь по ссылке <https://discord.gg/Yp248eJCSg>, переименуйте себя, как Иванов_Иван.
- конспект по желанию (вы самостоятельно определяете для себя то, где хранить информацию). Конспект больше как рекомендательный, чтобы понимать, что мы изучали и понимать объем и перечень вопросов и заданий, которые будут на итоговом контроле.
- вся информация в презентациях (+ дополнительно ссылки на источники). Презентация дублирует основную информацию занятия + задания для выполнения + обязательная домашняя работа
- в презентации описаны задания, которые нужно выполнить, подготовить текстовый отчет и защитить лабораторную работу
- сдача ВСЕХ лабораторных работ обязательна для получения допуска к экзамену
- срок сдачи лабораторных работ – 10 дней после проведенного занятия. Работы, которые будут сданы позднее срока принимаются, но +1 вопрос по теме работы на экзамене.
- работы сдавать можно в течение занятия (в этом случае текстовый отчет не нужно оформлять). Если не успели выполнить, отсутствовали – выполняйте задания самостоятельно, оформляйте текстовый отчет. Присылайте на проверку в ЛС в Discord в течение 10 дней.
- текстовый отчет оформляйте в свободной форме. Обязательным является титульный лист, скрины с комментариями кода JS, ссылка на репозиторий github, содержащая файлы (архив), выполненного задания.

Занятие 1. Введение в JS

1. История JS
2. Как запускать скрипты
3. «3 кита» JavaScript
4. Первый скрипт. Практика. Решение задач 😊
5. Событие `click`
6. Домашнее задание

Что нужно вспомнить? (или изучить)



HTML

строение сайта



CSS

внешний вид сайта



JS

взаимодействие с сайтом

Данная триада языков является кроссплатформенной – все, что вы напишите на html, css и JS одинаково будет работать во всех браузерах на всех операционных системах.

Что нужно вспомнить? (или изучить)

Нулевой уровень	Начальный уровень	Средний уровень
МООС (открытые онлайн-курсы, бесплатный доступ)		
Основы HTML и CSS https://stepik.org/course/52164/promo	Web-технологии https://stepik.org/course/154/syllabus	Серверные веб-технологии и системы управления контентом https://openedu.ru/course/ITMOUniversity/NODEJS/#
Веб-разработка для начинающих: HTML и CSS https://stepik.org/course/38218/promo	Создание динамического веб-сайта https://loftschool.com/modules/335-sozдание-dinamiceskogo-veb-saita	
Веб-дизайн и программирование: как создать дизайн сайта и сверстать его с нуля https://netology.ru/free-lessons/fullstack-veb-dizajner-kak-zapustit-sajt-samomu	Первые шаги в JavaScript: создаём сайт https://netology.ru/programs/jsfree-async#/about	
Быстрый старт в Figma https://netology.ru/free-lessons/bystryj-start-v-figma		
Основы верстки сайта https://netology.ru/programs/html-css-base#/		
Интернет-источники (свободный доступ). Это необязательно. Предлагаю ознакомиться с информацией и ее подачей		
<u>Веб-дизайн. Быстрый старт</u> https://www.youtube.com/playlist?list=PLD-piGJ3DtI184zqkJd7Lva_XBVGjUtC	Уроки по JS #itprogerLessons https://www.youtube.com/watch?v=fHl7UyRjOf0&list=PLDyJYA6aTY1kJlwbYHzGOuvSMNTfqksmk	
	Эффективное обучение HTML CSS JAVASCRIPT #Фрилансерпозизни https://www.youtube.com/watch?v=yJcCKuxfb2o&list=PLM6XATa8CAG4F9nAIYNS5oAiPotxwLFir	

1. История JS

- Автор языка Брендан Эйх
- Придумал JS за 10 дней
- ... в 1995 году

Прочитать по вопросу:

<https://learn.javascript.ru/intro>



1. История JS

JS умеет

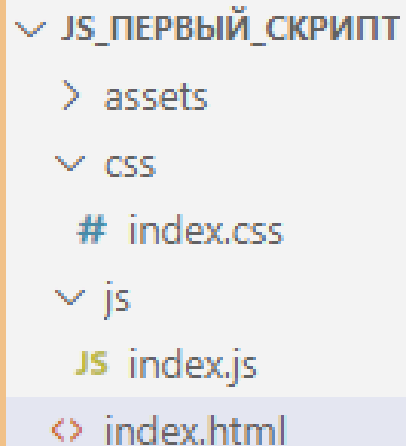
- работать с DOM
- реагировать на действия пользователя
- посылать запросы на сервер
- и др.

JS не умеет

- не имеет прямого доступа к системным функциям ОС
- не может получить доступ к веб-страницам, размещенным в другом домене

2. Как запускать скрипты

1. Любая работа (задача) – отдельный проект, который имеет структуру



```
▼ JS_ПЕРВЫЙ_СКРИПТ
  > assets
  ▼ css
    # index.css
  ▼ js
    JS index.js
  <> index.html
```

Практика:

Откройте проект: **JS_первый_скрипт**

- изучите способы подключения js файла к html
- напишите первый скрипт в файле index.js

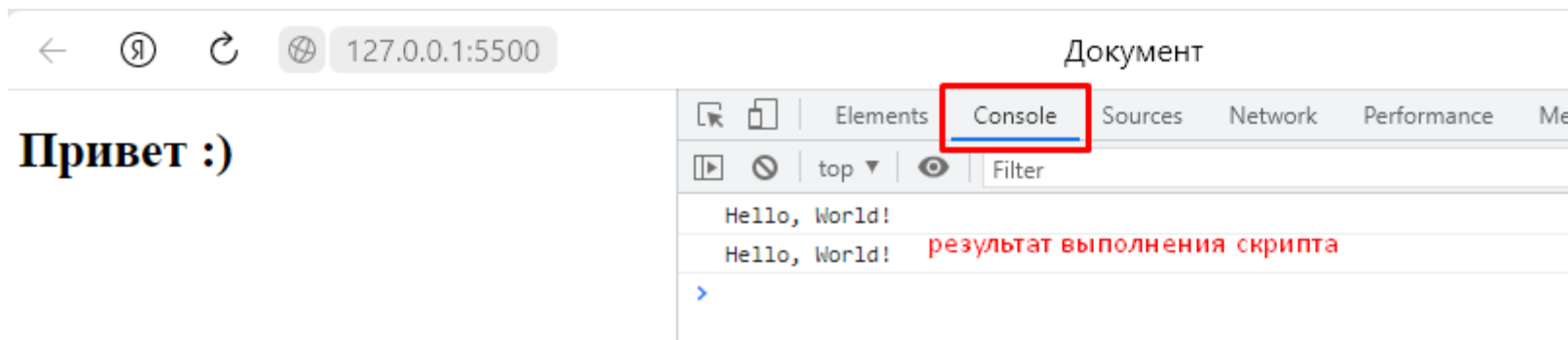
```
console.log(`Hello, World!`)
```
- рассмотрите все способы запуска js-скрипта

Прочитать по вопросу:

<https://learn.javascript.ru/hello-world>

<https://learn.javascript.ru/structure>

2. Как запускать скрипты



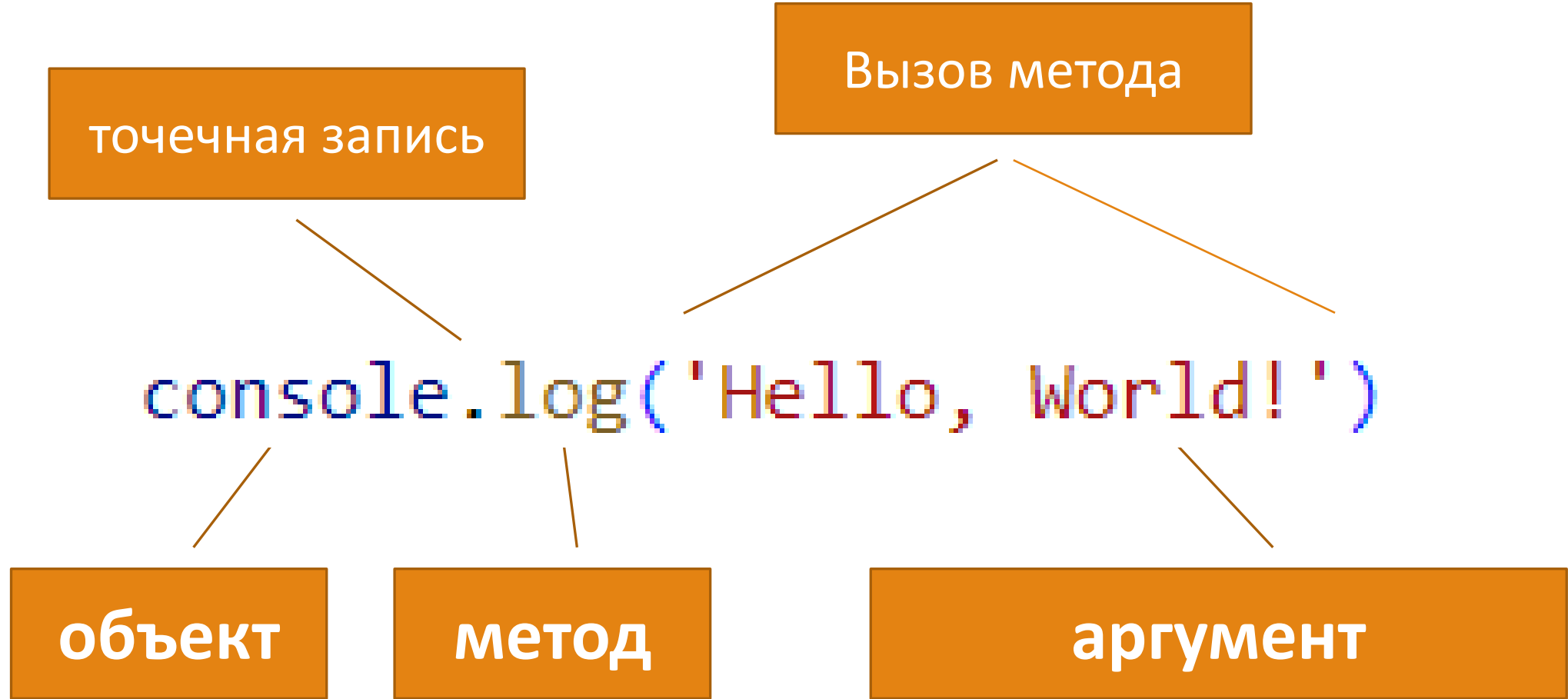
DevTools: Console

Консоль – лучший друг разработчика

<https://ru.hexlet.io/blog/posts/devtools>

Прочитать по вопросу:

<https://learn.javascript.ru/devtools>



Типы данных в JS

Чтобы хранить и использовать данные, их разделяют на типы:

1. число
2. строка
3. и др.

Прочитать по вопросу:

<https://learn.javascript.ru/types>

ПРИМИТИВНЫЕ ТИПЫ

String
Boolean
Number
Null
Undefined
Symbol

В памяти компьютера
эти переменные
непосредственно
содержат значение

ССЫЛОЧНЫЙ ТИП

Object

Типы данных в JS

Число

- без кавычек
- для чисел работают все математические операции

Строка

- Двойные кавычки
"Привет"
- Одинарные кавычки
'Привет'
- Обратные кавычки
`Привет`
- Строки можно складывать со строками и числами – результат всегда строка

Переменные в JS

let

```
> let a = 10
< undefined
> a = 20
< 20
> let b
< undefined
> b = true
< true
>
```

Появились в
ECMAScript 6

const

```
> const c = 10
< undefined
> c=20
✖ ▶ Uncaught TypeError: Assignment to constant variable.
   at <anonymous>:1:2
>
```

Var

это
устаревший
способ и мы
его
использовать
не будем

Прочитать по вопросу:

<https://learn.javascript.ru/variables>

Имена переменных в JS

<https://skillbox.ru/media/code/notatsii-v-programmirovanii/>

Рекомендации по названию переменных

1. **PascalCase** для типов и классов
2. **DB_PASSWORD** значения известны до запуска приложения и не меняются
3. **camelCase** для всех переменных

3. “3 кита” в JS. Как взаимодействуют html + js? Алгоритм

1. В HTML нужно создать элемент для вывода данных – указать для него уникальное имя
2. Найти этот элемент через JS
3. Записать ответ в элемент

1. Добавить атрибут id

```
<div id="output">  
  <!-- место для данных -->  
</div>
```

index.html

2. Найти элемент на странице по id

```
let outputNode = document.querySelector(`#output`);
```

3. Записать ответ в этот элемент

```
outputNode.innerHTML = `Это увидит  
пользователь!`;
```

js/index.js

4. Первый скрипт

Решение задач:

1. Задача 1. Gallery

2. Задача 2. Profile

3. Задача 3. Собери корзину

Задача 1. Gallery

Задача 1. Gallery

В нашу галерею привезли одну очень ценную и красивую картину с морским пейзажем.

Но сегодня ночью в галерею проникли воришки и украли её!

Они были технически прокаченные и взломали камеру наблюдения.

Камера показывает, что картина на месте, хотя её уже и след простыл...

С помощью JS почини камеру и покажи кадр с воришкой!

Задача 1. Алгоритм решения

1. Скачай архив gallery. Открой в VC Code

Как называется картина, которую украли?

Название написано в классе card-title.

2. Создай файл js/index.js и подключи его на страницу

3. Добавь атрибут в HTML

Чтобы заменить кадр с картиной, нам нужно найти элемент, внутри которого она лежит, и подписать его специальным атрибутом. Как он называется?

4. В файле js напиши скрипт, позволяющий поменять изображение

Задача 2. Profile

Задача 2. Profile

Тебе повезло — верстальщик уже подготовил страницу-шаблон со всей информацией о персонаже игрока.

Твоя задача — вынести все данные в переменные, а затем вставить их на страницу с помощью JS



Эмиль

Возраст: 20

Класс: Бард

Способности: Сила убеждения и остроумные шутки

В кармане 3 золотых

Задача 2. Profile

Задача 2. Profile. Алгоритм выполнения

1. Скачай архив profile
2. Объяви переменные

Имя name

Возраст age

Класс role

Умения ability

Золото gold

3. Выведи данные на экран



Эмиль

Возраст: 20

Класс: Бард

Способности: Сила убеждения и остроумные шутки

В кармане 3 золотых

Задача 3. Собери корзину

Задача 3. Собери корзину

1. Скачай архив shop

2. Выведи вместо знаков вопроса сумму за покупку яблок

Добавь количеству и цене id с названиями **appleCount** и **applePrice**

Добавь элементу со знаками вопроса id total

3. Заполни пропуски. Используй подсказку, как алгоритм выполнения — решения задачи.

```
JS index.js ×
1
2 // Найди элементы по id
3
4 let appleCountNode = ;
5
6 let applePriceNode = ;
7
8 let totalNode = ;
9
10 // Прочитай из HTML числа
11
12 let appleCount = ;
13
14 let applePrice = ;
15
16 // Выведи в элемент totalNode произведение количества на цену
17
18
19
20 console.log(totalNode);
21
```

Запомни!

1. В HTML нужно создать элемент для вывода данных – указать для него уникальное имя

Т.о. в разметке html для элемента необходимо указать идентификатор. Именно «обращаясь» к нему, браузер будет понимать с каким DOM-элементом работать

2. Найти этот элемент через JS

Чтобы найти элемент на странице html, в скрипте JS для поиска элемента чаще используют метод `querySelector('#idElem')` - <https://learn.javascript.ru/searching-elements-dom>

Т.к. элемент html мы ищем на странице, то все операции с DOM начинаются с объекта `document` - <https://learn.javascript.ru/dom-navigation>

Поэтому в переменную мы записываем результат выполнения – обращение к объекту `document` и выполнение метода `querySelector('#idElem')` – поиска элемента по его идентификатору.

3. Записать ответ в элемент

Для записи ответа в элемент (вывода информации на странице внутри выбранного элемента) используют свойство `innerHTML`, это свойство элемента, в которое можно записать любую строку, даже HTML-код!

Читать только про `innerHTML` <https://learn.javascript.ru/basic-dom-node-properties#innerHTML-soderzhimoe-elementa>

5. Событие `click`.

Как работать событие `click`? Алгоритм

1. В HTML нужно для элемента (например, кнопки) указать уникальное имя
2. Найти этот элемент через JS
3. С помощью метода `addEventListener()` «поймать» событие клика по элементу (кнопке) и описать алгоритм действий

1. Добавить кнопку с атрибутом id

```
<button id="click">Кнопка</button>
```

index.html

2. Найти кнопку на странице по id

```
let buttonNode = document.querySelector('#click');
```

3. Подписаться на событие Клик по кнопке

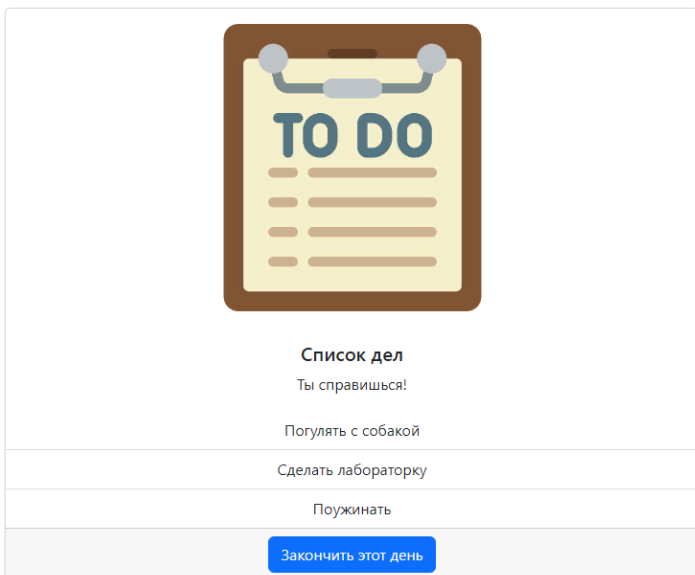
```
buttonNode.addEventListener('click', function () {  
    console.log('Меня нажали!');  
});
```

js/index.js

Задача 4. С чистого листа

Задача 4. С чистого листа

Очисти список дел. При нажатии на кнопку «Закончить этот день» сделай так, чтобы список полностью был очищен.



Задача 4. Алгоритм решения

1. Скачай архив todo
2. По клику на кнопку очищай элемент с id="hide"
3. Заполни пропуски

JS index.js ×

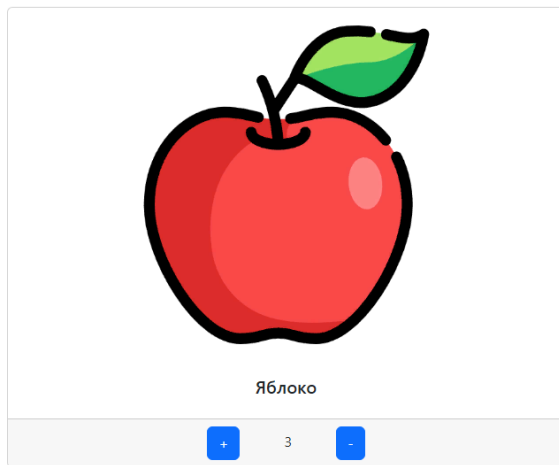
js > JS index.js > ...

```
1 let hideNode = document.querySelector(`#hide`);
2 let clickNode = document.querySelector(`#click`);
3 let textNode = document.querySelector(`#text`);
4
5 [redacted].addEventListener([redacted], function(){
6   hideNode.innerHTML = [redacted]
7   [redacted] = `Ура! Можно отдохнуть`;
8 })
```

Задача 5. Товар в корзину

Задача 5. Товар в корзину

Добавь товар в корзину



Задача 5. Алгоритм решения

1. Скачай архив todo
2. По клику на кнопку очищай элемент с id="hide"
3. Заполни пропуски

```
JS index.js  X
JS > JS index.js > ...
1  let plusNode = document.querySelector(`#plus`);
2  let amountNode = document.querySelector(`#amount`);
3
4  // Объяви переменную count со значением 0
5  // Добавь обработки событий на кнопку plusNode
6
7  // Увеличь count на 1
8  // Выведи count на экран
9
```

4. Добавь кнопку «Минус»

Задача 6. Картинная галерея

Задача 6. Картинная галерея

Создай галерею из 3-х рисунков, которые можно переключать по нажатию на кнопки.



Задача 6. Алгоритм решения

1. Скачай архив shop-gallery
2. По клику на кнопку 2 показывай картинку с сыром.
3. Заполни пропуски

```
let rowNode = document.querySelector(`#row`);
let button1Node = document.querySelector(`#button1`);
let button2Node = document.querySelector(`#button2`);
let button3Node = document.querySelector(`#button3`);

// При загрузке страницы показывай шаблон с яблоками

// Добавь обработчик событий, который по клику на кнопку 2
// показывает на экране картинку с сыром

// см.подсказку, но есть пропущенные элементы
[button2Node].addEventListener(button2Node, function(){
  rowNode.innerHTML = button2Node;
});
```

4. Добавь другие картинки, которые открываются по клику на кнопку 1 и 3

Запомни!

1. С помощью метода `addEventListener()` можно «поймать» событие клика по элементу (кнопке) и описать алгоритм действий

Читать только про `addEventListener()` <https://learn.javascript.ru/introduction-browser-events#addeventlistener>

Задача 7. Тайное послание

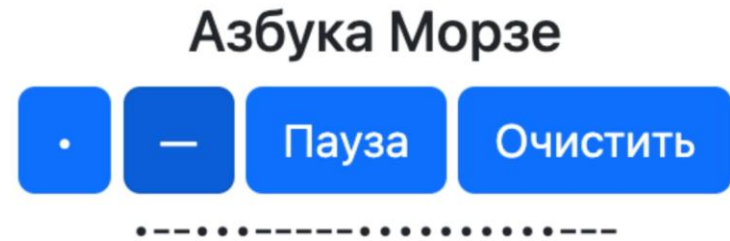
Домашнее задание

Создайте свой телеграфный аппарат Морзе.

Это простой интерфейс из четырех кнопок и элемента для вывода сообщения.

Каждая кнопочка добавляет свой символ к сообщению, а последняя все удаляет.

Используйте символы — и *, а для паузы используйте пробел.



Задача 7. Тайное послание

Ответь на вопросы:

1. Какое ключевое слово(а) объявляет(ют) переменные?
2. Какое свойство управляет содержимым элемента в DOM?
3. Ставится ли ; в конце строк в JS?
4. Какое сообщение закодировано?

*_ *_* ** *_ * _ *_*_ _ _ ** *_* __*__

Азбука Морзе				
А •—	К —•—	Ф ••—•	1 •— — — —	. •••••
Б —•••	Л •—••	Х ••••	2 •• — — —	, •—•—•—
В •— —	М — — —	Ц —•—•	3 ••• — —	; —•—•—•
Г — — •	Н — •	Ч — — — •	4 •••• —	: — — — •••
Д — ••	О — — —	Ш — — — —	5 •••••	? ••— — ••
Е •	П •— — •	Щ — — • —	6 — ••••	! — — •• — —
Ж ••• —	Р • — •	Ъ,ъ — •• —	7 — — •••	- — •••• —
З — — ••	С •••	Ы — • — —	8 — — — ••	« •—••—•
И ••	Т —	Э ••—••	9 — — — — •	(— — — — •
Й • — — —	У •• —	Ю •• — —	0 — — — — —	/ — •• — •
		Я • — • —		