|  |
| --- |
| const magicBtn = document.querySelector('.js-magic-btn'); |
|  |  |
|  | /\* |
|  | \* document.querySelector(selector) и document.querySelectorAll(selector) |
|  | \* selector - любой валидный CSS-селектор |
|  | \* |
|  | \* Что возвращают? |
|  | \*/ |
|  |  |
|  | const navEl = document.querySelector('.site-nav'); |
|  | console.log('navEl', navEl); |
|  |  |
|  | const navLinksEl = document.querySelectorAll('.site-nav\_\_link'); |
|  | console.log('navLinksEl', navLinksEl); |
|  |  |
|  | /\* |
|  | \* Document.querySelector\* и Element.querySelector\* |
|  | \*/ |

* © 2020 GitHub, Inc.

## 02-properties.js

37 lines (31 sloc)  1.13 KB

[Raw](https://github.com/luxplanjay/js-22/raw/07-1-dom/js/02-properties.js)[Blame](https://github.com/luxplanjay/js-22/blame/07-1-dom/js/02-properties.js)

Начало формы

Конец формы

Начало формы

Конец формы

|  |  |
| --- | --- |
|  | const magicBtn = document.querySelector('.js-magic-btn'); |
|  |  |
|  | /\* |
|  | \* Свойства элемента (hero) |
|  | \* - Изображение |
|  | \* - Текст и textContent |
|  | \* https://images.pexels.com/photos/1870376/pexels-photo-1870376.jpeg?auto=compress&cs=tinysrgb&dpr=2&w=480 |
|  | \*/ |
|  |  |
|  | const imageEl = document.querySelector('.hero\_\_image'); |
|  | // console.log('imageEl', imageEl); |
|  | imageEl.src = |
|  | 'https://images.pexels.com/photos/1870376/pexels-photo-1870376.jpeg?auto=compress&cs=tinysrgb&dpr=2&w=480'; |
|  | imageEl.alt = 'Это новый котик'; |
|  |  |
|  | const heroTitleEl = document.querySelector('.hero\_\_title'); |
|  | // console.log('heroTitleEl', heroTitleEl); |
|  | heroTitleEl.textContent = 'Я сладкий пирожочек!'; |
|  |  |
|  | /\* |
|  | \* Атрибуты |
|  | \* - get(имя-атрибута) |
|  | \* - set(имя-атрибута) |
|  | \* - remove(имя-атрибута) |
|  | \* - has(имя-атрибута) |
|  | \*/ |
|  |  |
|  | // console.log(imageEl.getAttribute('src')); |
|  | // console.log(imageEl.src); |
|  | // console.log(imageEl.hasAttribute('src')); |
|  |  |
|  | /\* |
|  | \* Data-атрибуты |
|  | \*/ |
|  | const actions = document.querySelectorAll('.js-actions button'); |
|  | console.log(actions); |
|  | console.log(actions[2].dataset.action); |

## ****03-classlist.js****

|  |
| --- |
| const magicBtn = document.querySelector('.js-magic-btn'); |
|  | /\* |
|  | \* Интерфейс classList |
|  | \* - add(класс) |
|  | \* - remove(класс) |
|  | \* - toggle(класс) |
|  | \* - replace(старыйКласс, новыйКЛасс) |
|  | \* - contains(класс) |
|  | \*/ |
|  |  |
|  | const currentPageUrl = '/contact'; |
|  |  |
|  | const linkEl = document.querySelector( |
|  | `.site-nav\_\_link[href="${currentPageUrl}"]`, |
|  | ); |
|  |  |
|  | console.log(linkEl); |
|  | linkEl.classList.add('site-nav\_\_link--current'); |

## [js-22](https://github.com/luxplanjay/js-22/tree/07-1-dom)/[js](https://github.com/luxplanjay/js-22/tree/07-1-dom/js)/****04-dom-traversal.js****

12 lines (10 sloc)  435 Bytes

[Raw](https://github.com/luxplanjay/js-22/raw/07-1-dom/js/04-dom-traversal.js)[Blame](https://github.com/luxplanjay/js-22/blame/07-1-dom/js/04-dom-traversal.js)

Начало формы

Конец формы

Начало формы

Конец формы

|  |  |
| --- | --- |
|  | /\* |
|  | \* Свойства «навигации» по DOM-узлам (взять список) |
|  | \* http://fecore.net.ua/books/m5ph3r-javascript/module-07/dom-traversal.html |
|  | \*/ |
|  |  |
|  | const navEl = document.querySelector('.site-nav'); |
|  |  |
|  | // const firstNavItemEl = navEl.querySelector('.site-nav\_\_item'); |
|  | const firstNavItemEl = navEl.firstElementChild; |
|  | console.log(firstNavItemEl); |
|  | console.log(navEl.children); |
|  | console.log(navEl.lastElementChild); |

## ****05-creating-elements.js****

|  |
| --- |
| /\* |
|  | \* - Создание элементов |
|  | \* - Вставка узлов: appendChild(elem), insertBefore(elem, nextSibling), append(...elems), prepend(...elems) |
|  | \*/ |
|  |  |
|  | /\* |
|  | \* Создаём заголовок |
|  | \*/ |
|  | const titleEl = document.createElement('h1'); |
|  | titleEl.classList.add('page-title'); |
|  | titleEl.textContent = 'Это заголовок страницы :)'; |
|  | // console.log(titleEl); |
|  |  |
|  | // document.body.appendChild(titleEl); |
|  |  |
|  | /\* |
|  | \* Создаём изображение |
|  | \* https://cdn.pixabay.com/photo/2018/07/26/07/45/valais-3562988\_1280.jpg |
|  | \* valais-alpine-mountains-glacier |
|  | \*/ |
|  | const imageEl = document.createElement('img'); |
|  | imageEl.src = |
|  | 'https://cdn.pixabay.com/photo/2018/07/26/07/45/valais-3562988\_1280.jpg'; |
|  | imageEl.alt = 'valais-alpine-mountains-glacier'; |
|  | imageEl.width = 320; |
|  | // console.log('imageEl', imageEl); |
|  |  |
|  | // document.body.appendChild(imageEl); |
|  |  |
|  | const heroEl = document.querySelector('.hero'); |
|  | // heroEl.appendChild(titleEl); |
|  | // heroEl.appendChild(imageEl); |
|  | heroEl.append(titleEl, imageEl); |
|  |  |
|  | /\* |
|  | \* Создаём и добавляем новый пункт меню |
|  | \*/ |
|  | const navItemEl = document.createElement('li'); |
|  | navItemEl.classList.add('site-nav\_\_item'); |
|  |  |
|  | const navLinkEl = document.createElement('a'); |
|  | navLinkEl.classList.add('site-nav\_\_link'); |
|  | navLinkEl.textContent = 'Личный кабинет'; |
|  | navLinkEl.href = '/profile'; |
|  |  |
|  | navItemEl.appendChild(navLinkEl); |
|  | // console.log(navItemEl); |
|  |  |
|  | const navEl = document.querySelector('.site-nav'); |
|  |  |
|  | // navEl.appendChild(navItemEl); |
|  | navEl.insertBefore(navItemEl, navEl.firstElementChild); |

## ****06-collections.js****

|  |
| --- |
| /\* |
|  | \* Создём и добавляем коллекцию |
|  | \*/ |
|  | const colorPickerOptions = [ |
|  | { label: 'red', color: '#F44336' }, |
|  | { label: 'green', color: '#4CAF50' }, |
|  | { label: 'blue', color: '#2196F3' }, |
|  | { label: 'grey', color: '#607D8B' }, |
|  | { label: 'pink', color: '#E91E63' }, |
|  | { label: 'indigo', color: '#3F51B5' }, |
|  | ]; |
|  |  |
|  | const colorPickerContainerEl = document.querySelector('.js-color-picker'); |
|  |  |
|  | // const elements = colorPickerOptions.map(option => { |
|  | // const buttonEl = document.createElement('button'); |
|  | // buttonEl.type = 'button'; |
|  | // buttonEl.classList.add('color-picker\_\_option'); |
|  | // buttonEl.textContent = option.label; |
|  | // buttonEl.style.backgroundColor = option.color; |
|  |  |
|  | // return buttonEl; |
|  | // }); |
|  |  |
|  | // console.log(elements); |
|  |  |
|  | /\* |
|  | \* Пишем функцию для создания разметки колорпикера |
|  | \*/ |
|  | const makeColorPickerOptions = options => { |
|  | return options.map(option => { |
|  | const buttonEl = document.createElement('button'); |
|  | buttonEl.type = 'button'; |
|  | buttonEl.classList.add('color-picker\_\_option'); |
|  | buttonEl.textContent = option.label; |
|  | buttonEl.style.backgroundColor = option.color; |
|  |  |
|  | return buttonEl; |
|  | }); |
|  | }; |
|  |  |
|  | const elements = makeColorPickerOptions(colorPickerOptions); |
|  | colorPickerContainerEl.append(...elements); |

## [js-22](https://github.com/luxplanjay/js-22/tree/07-1-dom)/[js](https://github.com/luxplanjay/js-22/tree/07-1-dom/js)/****07-products.js****

|  |
| --- |
| import products from './data/proucts.js'; |
|  |  |
|  | /\* |
|  | \* Создаём карточку продукта |
|  | \* - В классе продукта может быть product--on-sale product--not-available |
|  | \*/ |
|  |  |
|  | { |
|  | /\* |
|  | <article class="product"> |
|  | <h2 class="product\_\_name">Название</h2> |
|  | <p class="product\_\_descr">Описание</p> |
|  | <p product\_\_pridct>Цена: 1111 кредитов</p> |
|  | </article> |
|  | \*/ |
|  | } |
|  |  |
|  | /\* |
|  | \* Пишем функцию для создания карточки продукта |
|  | \*/ |
|  |  |
|  | const productsContainerEl = document.querySelector('.js-products'); |
|  |  |
|  | const makeProductCard = ({ name, description, price }) => { |
|  | const productEl = document.createElement('article'); |
|  | productEl.classList.add('product'); |
|  |  |
|  | const nameEl = document.createElement('h2'); |
|  | nameEl.textContent = name; |
|  | nameEl.classList.add('product\_\_name'); |
|  |  |
|  | const descrEl = document.createElement('p'); |
|  | descrEl.textContent = description; |
|  | descrEl.classList.add('product\_\_descr'); |
|  |  |
|  | const priceEl = document.createElement('p'); |
|  | priceEl.textContent = `Цена: ${price} кредитов`; |
|  | priceEl.classList.add('product\_\_price'); |
|  |  |
|  | productEl.append(nameEl, descrEl, priceEl); |
|  |  |
|  | return productEl; |
|  | }; |
|  |  |
|  | const elements = products.map(makeProductCard); |
|  |  |
|  | console.log(elements); |
|  | productsContainerEl.append(...elements); |

* © 2020 GitHub, Inc.

## ****08-templating.js****

|  |
| --- |
| /\* |
|  | \* Свойство innerHTML |
|  | \* - чтение |
|  | \* - запись |
|  | \*/ |
|  |  |
|  | const titleEl = document.querySelector('.title'); |
|  | // titleEl.innerHTML = '<a href="">Это ссылка)</a>'; |
|  | // titleEl.innerHTML = ''; |
|  |  |
|  | /\* |
|  | \* Вставка разметки с insertAdjacentHTML() |
|  | \* http://fecore.net.ua/books/m5ph3r-javascript/module-07/dom-manipulation.html#%D0%BC%D0%B5%D1%82%D0%BE%D0%B4-insertadjacenthtml |
|  | \*/ |
|  |  |
|  | titleEl.insertAdjacentHTML( |
|  | 'beforeend', |
|  | '<a href="" class="title\_\_link">Это ссылка)</a>', |
|  | ); |

## ****09-transactions.js****

|  |
| --- |
| import transactionHistory from './data/transactions.js'; |
|  |  |
|  | const makeTransactionTableRowMarkup = transaction => { |
|  | const { id, amount, date, business, type, name, account } = transaction; |
|  |  |
|  | return ` |
|  | <tr> |
|  | <td>${id}</td> |
|  | <td>${amount}</td> |
|  | <td>${date}</td> |
|  | <td>${business}</td> |
|  | <td>${type}</td> |
|  | <td>${name}</td> |
|  | <td>${account}</td> |
|  | </tr> |
|  | `; |
|  | }; |
|  |  |
|  | console.log(transactionHistory); |
|  | const tableEl = document.querySelector('.js-transaction-table'); |
|  | const transactionTableRowsMarkup = transactionHistory |
|  | .map(makeTransactionTableRowMarkup) |
|  | .join(''); |
|  |  |
|  | tableEl.insertAdjacentHTML('beforeend', transactionTableRowsMarkup); |
|  |  |
|  | console.log(transactionTableRowsMarkup); |

* © 2020 GitHub, Inc.

## [js-22](https://github.com/luxplanjay/js-22/tree/07-2-%D1%81%D0%BE%D0%B1%D1%8B%D1%82%D0%B8%D1%8F)/[js](https://github.com/luxplanjay/js-22/tree/07-2-%D1%81%D0%BE%D0%B1%D1%8B%D1%82%D0%B8%D1%8F/js)/****01-event-listeners.js****

|  |
| --- |
| /\* |
|  | \* События. |
|  | \* - Создание и удаление слушателей |
|  | \* - Именование колбеков для слушателей |
|  | \* - handle\*: handleSubjectEvent |
|  | \* - \*Handler: subjectEventHandler |
|  | \* - on\*: onSubjectEvent |
|  | \* - Ссылочная идентичность колбеков |
|  | \* - Объект события |
|  | \*/ |
|  |  |
|  | const targetBtn = document.querySelector('.js-target-btn'); |
|  | const addListenerBtn = document.querySelector('.js-add-listener'); |
|  | const removeListenerBtn = document.querySelector('.js-remove-listener'); |
|  |  |
|  | addListenerBtn.addEventListener('click', event => { |
|  | console.log(event); |
|  | console.log('Вешаю слушателя события на целевую кнопку'); |
|  |  |
|  | targetBtn.addEventListener('click', onTargetBtnClick); |
|  | }); |
|  |  |
|  | removeListenerBtn.addEventListener('click', event => { |
|  | console.log(event); |
|  | console.log('Снимаю слушателя события с целевой кнопки'); |
|  |  |
|  | targetBtn.removeEventListener('click', onTargetBtnClick); |
|  | }); |
|  |  |
|  | function onTargetBtnClick(event) { |
|  | console.log(event); |
|  | console.log('Клик по целевой кнопке'); |
|  | } |

## ****02-forms.js****

|  |
| --- |
| /\* |
|  | \* - Событие submit |
|  | \* - Действия браузера по умолчанию |
|  | \* - Свойство elements |
|  | \* - Класс FormData - https://developer.mozilla.org/en-US/docs/Web/API/FormData |
|  | \*/ |
|  |  |
|  | const form = document.querySelector('.js-register-form'); |
|  |  |
|  | form.addEventListener('submit', onFormSubmit); |
|  |  |
|  | function onFormSubmit(event) { |
|  | event.preventDefault(); |
|  |  |
|  | const formData = new FormData(event.currentTarget); |
|  |  |
|  | console.log(formData); |
|  |  |
|  | formData.forEach((value, name) => { |
|  | console.log('onFormSubmit -> name', name); |
|  | console.log('onFormSubmit -> value', value); |
|  | }); |
|  | } |

## ****03-input-events.js****

|  |
| --- |
| /\* |
|  | \* Паттерн «Объект ссылок» |
|  | \* |
|  | \* События |
|  | \* - focus и blur |
|  | \* - input и change |
|  | \* - Чекбоксы и свойство checked |
|  | \*/ |
|  |  |
|  | const refs = { |
|  | input: document.querySelector('.js-input'), |
|  | nameLabel: document.querySelector('.js-button > span'), |
|  | licenseCheckbox: document.querySelector('.js-license'), |
|  | btn: document.querySelector('.js-button'), |
|  | }; |
|  |  |
|  | // refs.input.addEventListener('focus', onInputFocus); |
|  | // refs.input.addEventListener('blur', onInputBlur); |
|  | // refs.input.addEventListener('change', onInputChange); |
|  | // refs.input.addEventListener('input', onInputChange); |
|  |  |
|  | refs.input.addEventListener('input', onInputChange); |
|  | refs.licenseCheckbox.addEventListener('change', onLicenseChange); |
|  |  |
|  | function onInputFocus() { |
|  | console.log('Инпут получил фокус - событие focus'); |
|  | } |
|  |  |
|  | function onInputBlur() { |
|  | console.log('Инпут потерял фокус - событие blur'); |
|  | } |
|  |  |
|  | function onInputChange(event) { |
|  | refs.nameLabel.textContent = event.currentTarget.value; |
|  | } |
|  |  |
|  | function onLicenseChange(event) { |
|  | refs.btn.disabled = !event.currentTarget.checked; |
|  | } |

## ****04-keyboard-events.js****

|  |
| --- |
| /\* |
|  | \* Типы событий: keypress, keydown, keyup |
|  | \* - Ограничения keypress |
|  | \* - Свойства KeyboardEvent.key и KeyboardEvent.code |
|  | \*/ |
|  |  |
|  | const refs = { |
|  | output: document.querySelector('.js-output'), |
|  | clearBtn: document.querySelector('.js-clear'), |
|  | }; |
|  |  |
|  | window.addEventListener('keypress', onKeypress); |
|  | refs.clearBtn.addEventListener('click', onClearOutput); |
|  |  |
|  | function onKeypress(event) { |
|  | // console.log('event.key: ', event.key); |
|  | // console.log('event.code: ', event.code); |
|  |  |
|  | refs.output.textContent += event.key; |
|  | } |
|  |  |
|  | function onClearOutput() { |
|  | refs.output.textContent = ''; |
|  | } |

## ****05-mouse-events.js****

[Go to file](https://github.com/luxplanjay/js-22/find/07-2-%D1%81%D0%BE%D0%B1%D1%8B%D1%82%D0%B8%D1%8F)

|  |
| --- |
| /\* |
|  | \* События мыши |
|  | \* - mouseenter и mouseleave (это ховер) |
|  | \* - mouseover и mouseout |
|  | \* - mousemove (chatty event - болтливое событие) |
|  | \* - Допмат по координатам: https://nerdparadise.com/programming/javascriptmouseposition |
|  | \* - Хорошая задачка - https://learn.javascript.ru/task/move-ball-field |
|  | \*/ |
|  |  |
|  | const boxRef = document.querySelector('.js-box'); |
|  |  |
|  | // boxRef.addEventListener('mouseover', onMouseEnter); |
|  | // boxRef.addEventListener('mouseout', onMouseLeave); |
|  | // boxRef.addEventListener('mousemove', onMouseMove); |
|  |  |
|  | function onMouseEnter(event) { |
|  | const box = event.currentTarget; |
|  | box.classList.add('box--active'); |
|  | } |
|  |  |
|  | function onMouseLeave(event) { |
|  | const box = event.currentTarget; |
|  | box.classList.remove('box--active'); |
|  | } |
|  |  |
|  | function onMouseMove(event) { |
|  | console.log(event); |
|  | } |

## ****06-modal.js****

|  |
| --- |
| /\* |
|  | \* 1. Открыть и закрыть по кнопке |
|  | \* 2. Закрыть по клику в бекдроп: onBackDropClick |
|  | \* 3. Закрыть по ESC: onEscapeKeypress |
|  | \* |
|  | \* В CSS есть класс show-modal, который необходимо добавить на body при открытии модалки |
|  | \*/ |
|  |  |
|  | const refs = { |
|  | openModalBtn: document.querySelector('[data-action="open-modal"]'), |
|  | closeModalBtn: document.querySelector('[data-action="close-modal"]'), |
|  | backdrop: document.querySelector('.js-backdrop'), |
|  | }; |
|  |  |
|  | refs.openModalBtn.addEventListener('click', onOpenModal); |
|  | refs.closeModalBtn.addEventListener('click', onCloseModal); |
|  | refs.backdrop.addEventListener('click', onBackdropClick); |
|  |  |
|  | function onOpenModal() { |
|  | window.addEventListener('keydown', onEscKeyPress); |
|  | document.body.classList.add('show-modal'); |
|  | } |
|  |  |
|  | function onCloseModal() { |
|  | window.removeEventListener('keydown', onEscKeyPress); |
|  | document.body.classList.remove('show-modal'); |
|  | } |
|  |  |
|  | function onBackdropClick(event) { |
|  | if (event.currentTarget === event.target) { |
|  | console.log('Кликнули именно в бекдроп!!!!'); |
|  | onCloseModal(); |
|  | } |
|  | } |
|  |  |
|  | function onEscKeyPress(event) { |
|  | const ESC\_KEY\_CODE = 'Escape'; |
|  | const isEscKey = event.code === ESC\_KEY\_CODE; |
|  |  |
|  | if (isEscKey) { |
|  | onCloseModal(); |
|  | } |
|  | } |