

## Домашняя работа №14. «Модули в JS. Создание SPA»

### Задание:

Создайте страницу с продуктами и корзину товаров интернет-магазина. Дизайн страниц может быть любым. Предусмотрите адаптивность к мобильным устройствам. **Пример смотрите ниже.**

### Требования:

1. Сайт должен использовать одну точку входа — файл `index.html`. Создайте HTML-документ (`index.html`) и подключите в него файл `index.js`. Создайте стартовую структуру HTML-документа (точно так, как в примере):

```
<!DOCTYPE html>
<html>
<head></head>
<body>
  <script src="/index.js"></script>
</body>
</html>
```

В исходном файле `index.html` должна оставаться только стартовая структура. Остальные элементы и их атрибуты, в т.ч. код CSS, должны появляться динамически из JS.

2. Создайте папку «`components`» и добавьте в нее файл `App.js`. В файле создайте класс «`App`» (главный компонент вашего приложения) со следующими методами:

- Метод `create()` для создания главного элемента «`div`» с классом «`app`» и сохранения его в свойстве «`element`».
- Метод `render()` для добавления главного элемента в тег «`body`» HTML-документа и для последующего добавления в контейнер `<div class="app"></div>` других элементов из других загруженных компонентов.
- Метод `init()` для инициализации главного компонента вашего приложения — для описании будущей логики загрузки других компонентов, а также для вызова методов `create()` и `render()`.

Выполните экспорт по умолчанию «анонимного» объекта вашего класса с вызовом метода `init()`: `export default new App().init();`

В файле `index.js` выполните импорт вашего компонента. Результатом выполнения импорта должно быть появление в теге `body` контейнера `<div class="app"></div>`.

3. Заполните контейнер head: добавьте кодировку, заголовок и подключите стили. CSS стили описывайте в соотв. файле style.css, добавив его в каталог css. Выполните данные работы в методе init() главного компонента.

4. Создайте общие структурные компоненты (Header, Nav, Main, Footer). В каждом компоненте (в классе) должны быть реализованы методы:

- Метод create() для создания соотв. элемента со своим именем класса и сохранения его в свойстве «element», а также для добавления в него содержимого.
- Метод init() для инициализации компонента — для вызова метода create() и возврата элемента из метода.

Все созданные компоненты должны находится в папке «components».

Выполните экспорт элемента из объекта каждого класса с помощью метода init(), например:

```
const footer = new Footer().init();  
export {footer};
```

5. В классе главного компонента реализуйте метод для получения данных из «Fake» API сервиса <https://fakestoreapi.com/products>. Данные должны быть сохранены localStorage. Если в localStorage уже есть эти данные, повторно получать их не нужно. Метод должен быть запущен при инициализации и до отрисовки в DOM.

6. На главной странице сайта по адресу <http://localhost/> отобразите список товаров, полученных из localStorage. Дизайн может быть любым.

7. На сайте должна быть реализована возможность добавление товара в корзину с помощью COOKIE. В виджете корзины на всех страницах должно отображаться кол-во добавленных товаров и общая стоимость корзины.

8. Переход на страницу корзины должен быть реализован по адресу <http://localhost/#cart/>. Обратите внимание на «решеточку» (хештег).

9. На странице корзины товаров предусмотрите возможность удаления товаров и пересчет общей стоимости в зависимости от кол-ва.

10. Для каждого товара сделайте отдельную станицу с полной информацией по нему. Переход на страницу с товаров должен осуществляться по адресу <http://localhost/#product/21/>, где цифра в адресе соотв. идентификатору товара. Обратите внимание на «решеточку» (хештег).


11. Реализуйте «роутинг» (переход между страницами) в компоненте «Main». Переход между страницами сайта должен выполняться без перезагрузки браузер, т.е. содержимое должно меняться динамически с учетом получаемого адреса в браузере. Такой переход характерен для SPA-приложений.

В зависимости от полученного хештега:


- Адрес без хештега — главная страница;
- Адрес `/#product/21/` — страница конкретного товара;
- Адрес `/#cart/` — страница корзины.

Для отслеживания события изменения хештега в адресной строке используйте `addEventListener` и соотв. событие «`hashchange`».

Пример страницы товаров.




[Home](#) [Shop](#) [Contact](#)



1

\$30.00


Furniture



Comform Chair


\$99.00

ADD TO CART




Comform Foam Chair

\$99.00




Wooden Tool

\$30.00




Squire Table

\$23.00




lower Vase

\$20.00




Hot Water


\$15.00




Wooden Lights

\$27.00






59 Street, Newyork City, Rose Town, 05 Rive House




+123 456 7890




info@example.com





Пример страницы корзины.




HomeShopContact


\$30.00


Cart


Product	Price	Quantity	Subtotal
<div><div></div><div><div>Wooden Tool</div></div></div> <div>\$30.00</div> <div><div>1</div><div><div>+</div><div>-</div></div></div> <div>\$30.00</div>			
<div><div></div><div><div>Comform Foam Chair</div></div></div> <div>\$99.00</div> <div><div>1</div><div><div>+</div><div>-</div></div></div> <div>\$99.00</div>			

UPDATE CART



59 Street, Newyork City, Rose Town, 05 Rive House

+123 456 7890

info@example.com