**Обработка на събития**

**Зад 1. Създаване на уеб магазин**

В тази задача, ще създадем ново уеб магазин, в който потребителите ще могат да поръчват различни стоки. За целта, изпълнете следните задачи:

1. Създайте ново react приложение, като използвате използвате npm и vite (вижте в предишното упражнение).
2. В папката src създайте нов компонент с името Shop.
3. Променете връщания от App код, за да визуализира <Shop/>
4. Във функцията Shop() създайте масив от обекти с името items, съдържащ елементите, които се продават в магазина. Всеки обект трябва да съдържа id, име, цена и име на файла, съдържащ изображението на продукта (изображенията поставяйте в папката src/assets).
5. В папкат src, създайте компонента Item (Item.jsx). Функцията Item трябва да приема променливата props, през която пристига свойството (атрибута/променливата) itemData. Свойството itemData съдържа един обект от масива, дефинирам в Shop().
6. Нарисувайте масива дефиниран в Shop() на екрана, като всеки елемент трябва да бъде нарисуван от Item. За целта, обходете масива и за всеки обект извикайте <Item/>

<Item itemData={ПореденОбектОтМасива}/>

1. Във функцията App(), чрез useState(), създайте променливата cart (количка), представляваща празен масив. Променливата ще съдържа артикулите, добавени в кличката. Може да използвате следния код:

let [cart, updateCart] = useState([]);

1. От App, предайте cart и updateCart на Shop като атрибути:

<Shop cart={cart} updateCart={updateCart} />

1. От Shop(), предайте cart и updateCart на Item като атрибути.
2. В Shop() добавете код, който показва колко артикула има в количката.
3. До всеки артикул, добавете бутон на който пише „Добави в количката“.
4. Създайте функция, която да добавя текущия артикул към количката и я свържете за бутона. За да добавите текущия артикул към бутона, използвайте код подобен на следния:

updateCart([…cart, {…props.itemData, count:1}]);

**Зад 2. Добавяне на страница, визуализираща количката**

За целта, изпълнете следните задачи:

1. Към проекта добавете компонента Basket (Basket.jsx).
2. Към проекта добавете пакета react-router-dom, като изпълните следната команда в папката, в която се намира package.json:

npm install react-router-dom

1. В App(), Около <Shop/> сложете етикетите <BrowserRoter>
2. Премахнете <Shop/> и го заменете с етикетите <Routes> и няколко етикета <Route>, за да укажете, че базовия адрес „/” трябва да показва <Shop>, а адреса „basket“ трябва да показва <Basket/>

Забележка: Вижте лекцията „Лекция 5. React куки и навигация“

1. Добавете нов компонент Menu (Menu.jsx), който да съдържа хипервръзки към страниците.
2. В App() между <BrowserRouter/> и <Routes/> добавете <Menu/>
3. Добавете функционалност, за да покажете всички добавени в кошницата артикули във в компонента Basket (функцията Basket()). Basket трябва да позволява промяна на бройката на поръчаните артикули и изтриване на артикул.

**Зад 3. Замяна на стойностите на масива съдържащ артикулите с уеб услуги**

В тази задача ще заменим стойностите на масива items, намиращ се в Shop() с уеб услуга. Няма да създаваме сами уеб услугата а ще използваме готова библиотека, която на базата на json файл сама създава уеб услугите. За целта, изпълнете следните задачи:

1. В папката src, създайте нова папка с името restful.
2. В папката restful, създайте файла restful.json. В него поставете следното съдържание и го променете, за да отговаря на вашите артикули:

{

"items": [

{

"id": 1,

"name": "Тениска",

"price": "10,30",

"img": "img1.pmg"

},

{

"id": 2,

"name": "Блуза",

"price": "27,10",

"img": "img2.png"

},

]

}

1. Отворете команден прозорец (може и допълнителен терминал във Visual Studio Code) и изпълнете следната команда (ако ви попитат дали да бъде инсталиран сървъра, изберете да):

npx json-server --watch **ПЪЛНИЯТ\_ПЪТ\_ДО\_ФАЙЛА**\restful.json --port 8000

Горния код ще стартира програмата json-server, която предоставя restful Уеб услуги на порт 8000.

Можем например, да направим заявка, чрез баузъра до:

http://localhost:8000/items/

или

http://localhost:8000/items/1

1. Във вътрешността на Shop() добавете куката useEffect(). Трябва да я импортирате от react. useEffect() може да изглежда така:

useEffect(()=>{

код извикващ <http://localhost:8000/items/>,чрез fetch() и

записващ върнатия резултат в масива items.

При извикването на fetch() ще искаме да получим данните, чрез json(). Tova

ще ни върне масив от JavaScript обекти, които директно можем да

запишем в масива items.

}, [])