7.REDUX НА ХУКАХ

1.Пишем App на хуках.

2.Импортируем useSelector из библиотеки react-redux и useDispatch оттудаже в компоненте App.

3.Начальное состояние App:

import './App.scss';

import {Header,Cart} from './components';

import React,{useEffect,useState} from 'react'

import {Route} from "react-router-dom";

import {useSelector,useDispatch} from "react-redux"

import Home from './components/pages/Home';

function App(setPizzas,pizzas) {

useEffect(() => {

  fetch("http://localhost:3000/db.json")

.then((response)=>response.json()).then(json=>{

})

}, []

)

  return (

  <div>

   <div className="wrapper">

      <Header/>

      <div className="content">

        <Route exact path="/" render={()=><Home pizzas= {[]}/>}/>

        <Route  path="/cart" component={Cart}/>

      </div>

    </div>

    </div>

  )

}

export default App

4.Используем useDispatch для создания конст dispatch в App компоненте:

const dispatch=useDispatch()

и потом так

useEffect(() => {

  fetch("http://localhost:3000/db.json")

.then((response)=>response.json()).then(json=>{

  dispatch(setPizzasActionCreator(json.pizzas))

})

}, []

)

5.Используем useSelector как аналог mapStateToProps:

const state =useSelector(({pizzas,filters})=>{

return{

pizzas:pizzas.pizzas,

sortBy:filters.sortBy

}

}

)

6.Переносим логику useSelector из App в Home:

7.Меняем в render на component:

<Route exact path="/" render={()=><Home pizzas= {state.pizzas}/>}/>

На:

<Route exact path="/" component={Home}/>

8.Прописываем onClickItem в CategoriesWithHooks:

function onSelectItem(index){

  setActiveItem(index)

  props.onClickItem(index)

}

9.Создаем в Home конст dispatch:

const dispatch = useDispatch()

и меняем в компоненте:

 <CategoriesWithHooks onClickItem={(index)=>onClickItem(index)}

           items={["Мясные","Вегетарианская","Гриль","Острые","Закрытые"]}/>

функция :

const onClickItem=(index)=>{

  dispatch(setCategoryActionCreator(index))

}

onClickItem переписали.

10.В CategoriesWithHooks сделали изменения:

function onSelectItem(index){

  setActiveItem(index)

-----------------------------

  props.onClickItem(index)

------------------------------

}

11.Далее идет про ререндер…

12.Устанавливаем json server:

Npx json server –help

Или

Npm install json.serv

13.Решение

Кто использует npm и имеет проблемы с запуском приложения вместе c json-server, делайте следующее:

1. в консоли Кто использует npm и имеет проблемы с запуском приложения вместе c json-server, делайте следующее: 1) в консоли Кто использует npm и имеет проблемы с запуском приложения вместе c json-server, делайте следующее: 1) в консоли npm i -D concurrently; 2) в package.json внутри "scripts" внесите "dev": "concurrently \"npm start\" \"npm run server\""; 3) в консоли npm run dev.; 2) в package.json внутри "scripts" внесите "dev": "concurrently \"npm start\" \"npm run server\""; 3) в консоли npm run dev.;
2. Запускаем воттак :

"npx json-server --watch public/db.json --port=3001"

3)Подключили json server запрос делаем к нему в App:

useEffect(() => {

  fetch("http://localhost:3001/pizzas")

.then((response)=>response.json()).then(json=>{

  dispatch(setPizzasActionCreator(json))

console.log(json.pizzas)

})

}, []

)