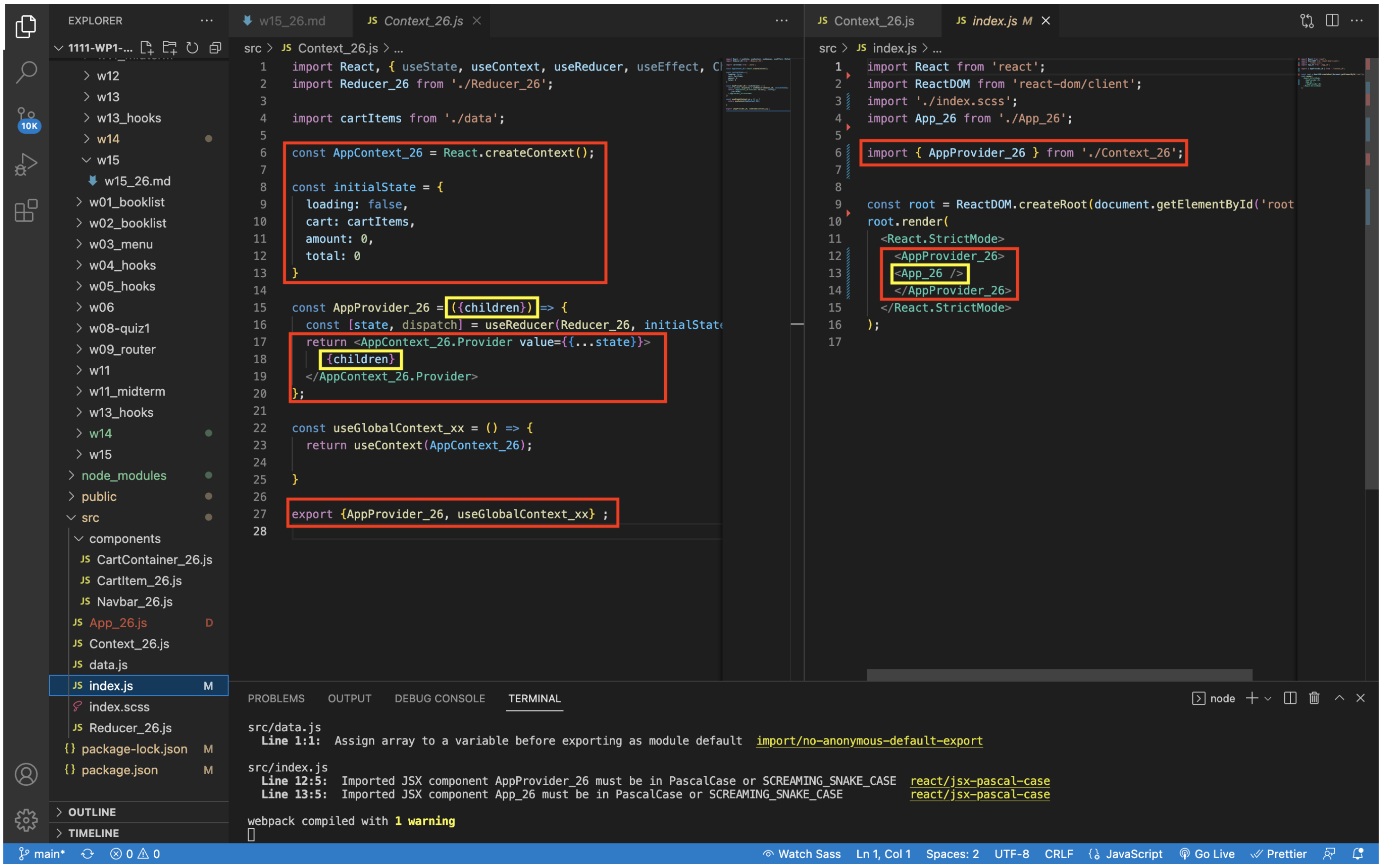
W15-P1: Setup AppProvider_xx for App_xx and get state data

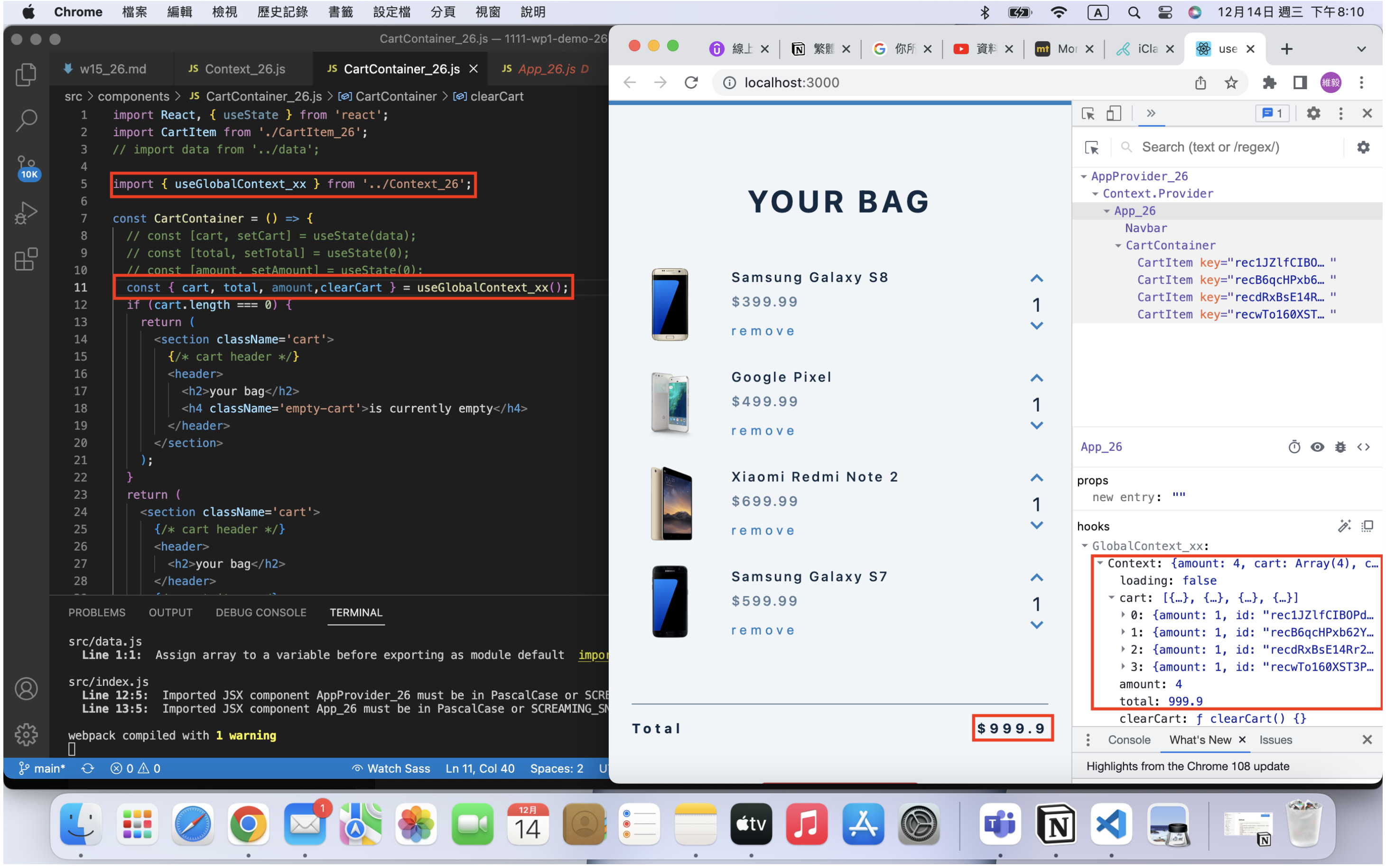


```
import React, { useState, useContext, useReducer, useEffect, Cl
import Reducer_26 from './Reducer_26';

import cartItems from './data';

const AppContext_26 = React.createContext();

const initialState = {
  loading: false,
  cart: cartItems,
  amount: 0,
  total: 0
}

const AppProvider_26 = ({children}) => {
  const [state, dispatch] = useReducer(Reducer_26, initialState
  return <AppContext_26.Provider value={{...state}}>
    {children}
  </AppContext_26.Provider>
};

const useGlobalContext_xx = () => {
  return useContext(AppContext_26);

}

export {AppProvider_26, useGlobalContext_xx} ;
```

```
import React from 'react';
import ReactDOM from 'react-dom/client';
import './index.scss';
import App_26 from './App_26';

import { AppProvider_26 } from './Context_26';


const root = ReactDOM.createRoot(document.getElementById('root
root.render(
  <React.StrictMode>
    <AppProvider_26>
    <App_26 />
    </AppProvider_26>
  </React.StrictMode>
);
```

```
src/data.js
  Line 1:1:  Assign array to a variable before exporting as module default  import/no-anonymous-default-export

src/index.js
  Line 12:5:  Imported JSX component AppProvider_26 must be in PascalCase or SCREAMING_SNAKE_CASE  react/jsx-pascal-case
  Line 13:5:  Imported JSX component App_26 must be in PascalCase or SCREAMING_SNAKE_CASE          react/jsx-pascal-case

webpack compiled with 1 warning
```

W15-P2: replace data in CartContainer using global context



Chrome 檔案 編輯 檢視 歷史記錄 書籤 設定檔 分頁 視窗 說明
12月14日 週三 下午8:10
CartContainer_26.js — 1111-wp1-demo-26
w15_26.md
Context_26.js
CartContainer_26.js
App_26.js D
src > components > CartContainer_26.js > CartContainer > clearCart
localhost:3000

```
import React, { useState } from 'react';
import CartItem from './CartItem_26';
// import data from '../data';

import { useGlobalContext_xx } from '../Context_26';

const CartContainer = () => {
  // const [cart, setCart] = useState(data);
  // const [total, setTotal] = useState(0);
  // const [amount, setAmount] = useState(0);
  const { cart, total, amount,clearCart } = useGlobalContext_xx();
  if (cart.length === 0) {
    return (
      <section className='cart'>
        {/* cart header */}
        <header>
          <h2>your bag</h2>
          <h4 className='empty-cart'>is currently empty</h4>
        </header>
      </section>
    );
  }
  return (
    <section className='cart'>
      {/* cart header */}
      <header>
        <h2>your bag</h2>
      </header>
```

```
src/data.js
  Line 1:1:  Assign array to a variable before exporting as module default  impor

src/index.js
  Line 12:5:  Imported JSX component AppProvider_26 must be in PascalCase or SCRE
  Line 13:5:  Imported JSX component App_26 must be in PascalCase or SCREAMING_SN

webpack compiled with 1 warning
```

YOUR BAG

Samsung Galaxy S8
$399.99
remove
1

Google Pixel
$499.99
remove
1

Xiaomi Redmi Note 2
$699.99
remove
1

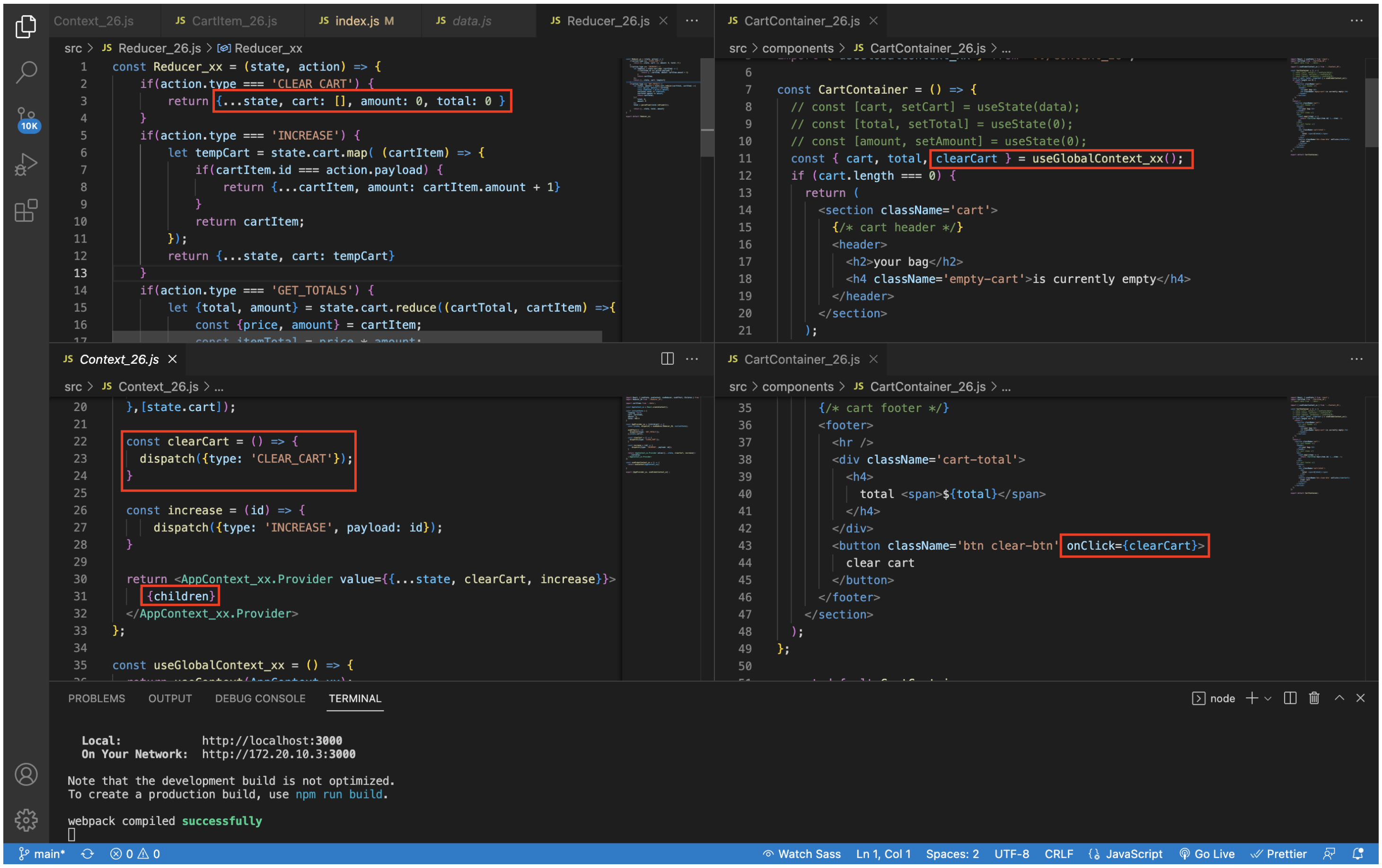
Samsung Galaxy S7
$599.99
remove
1

Total $999.9

```
▾ AppProvider_26
  ▾ Context.Provider
    ▾ App_26
        Navbar
      ▾ CartContainer
          CartItem key="rec1JZlfCIBO… "
          CartItem key="recB6qcHPxb6… "
          CartItem key="recdRxBsE14R… "
          CartItem key="recwTo160XST… "

App_26

props
  new entry: ""

hooks
 ▾ GlobalContext_xx:
   ▾ Context: {amount: 4, cart: Array(4), c…
       loading: false
     ▾ cart: [{…}, {…}, {…}, {…}]
       ▸ 0: {amount: 1, id: "rec1JZlfCIBOPd…
       ▸ 1: {amount: 1, id: "recB6qcHPxb62Y…
       ▸ 2: {amount: 1, id: "recdRxBsE14Rr2…
       ▸ 3: {amount: 1, id: "recwTo160XST3P…
       amount: 4
       total: 999.9
       clearCart: ƒ clearCart() {}
```

Highlights from the Chrome 108 update

W15-P3: put clear cart function in global context



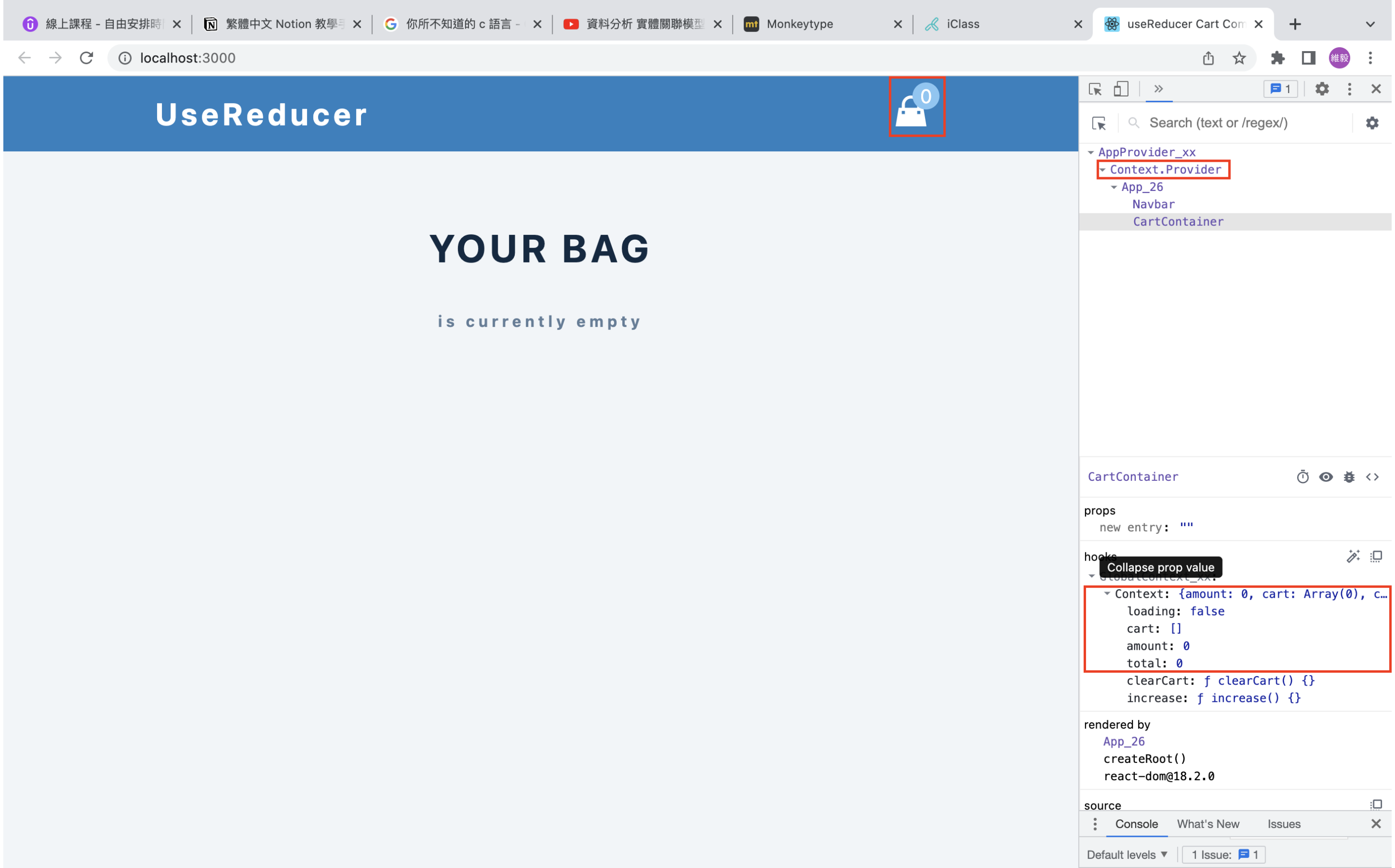
```
const Reducer_xx = (state, action) => {
    if(action.type === 'CLEAR_CART') {
        return {...state, cart: [], amount: 0, total: 0 }
    }
    if(action.type === 'INCREASE') {
        let tempCart = state.cart.map( (cartItem) => {
            if(cartItem.id === action.payload) {
                return {...cartItem, amount: cartItem.amount + 1}
            }
            return cartItem;
        });
        return {...state, cart: tempCart}
    }
    if(action.type === 'GET_TOTALS') {
        let {total, amount} = state.cart.reduce((cartTotal, cartItem) =>{
            const {price, amount} = cartItem;
```

```
const CartContainer = () => {
  // const [cart, setCart] = useState(data);
  // const [total, setTotal] = useState(0);
  // const [amount, setAmount] = useState(0);
  const { cart, total, clearCart } = useGlobalContext_xx();
  if (cart.length === 0) {
    return (
      <section className='cart'>
        {/* cart header */}
        <header>
          <h2>your bag</h2>
          <h4 className='empty-cart'>is currently empty</h4>
        </header>
      </section>
    );
```

```
  },[state.cart]);

  const clearCart = () => {
    dispatch({type: 'CLEAR_CART'});
  }

  const increase = (id) => {
      dispatch({type: 'INCREASE', payload: id});
  }

  return <AppContext_xx.Provider value={{...state, clearCart, increase}}>
      {children}
  </AppContext_xx.Provider>
};

const useGlobalContext_xx = () => {
```

```
        {/* cart footer */}
        <footer>
          <hr />
          <div className='cart-total'>
            <h4>
              total <span>${total}</span>
            </h4>
          </div>
          <button className='btn clear-btn' onClick={clearCart}>
            clear cart
          </button>
        </footer>
      </section>
    );
  };
```

```
  Local:            http://localhost:3000
  On Your Network:  http://172.20.10.3:3000

Note that the development build is not optimized.
To create a production build, use npm run build.

webpack compiled successfully
```



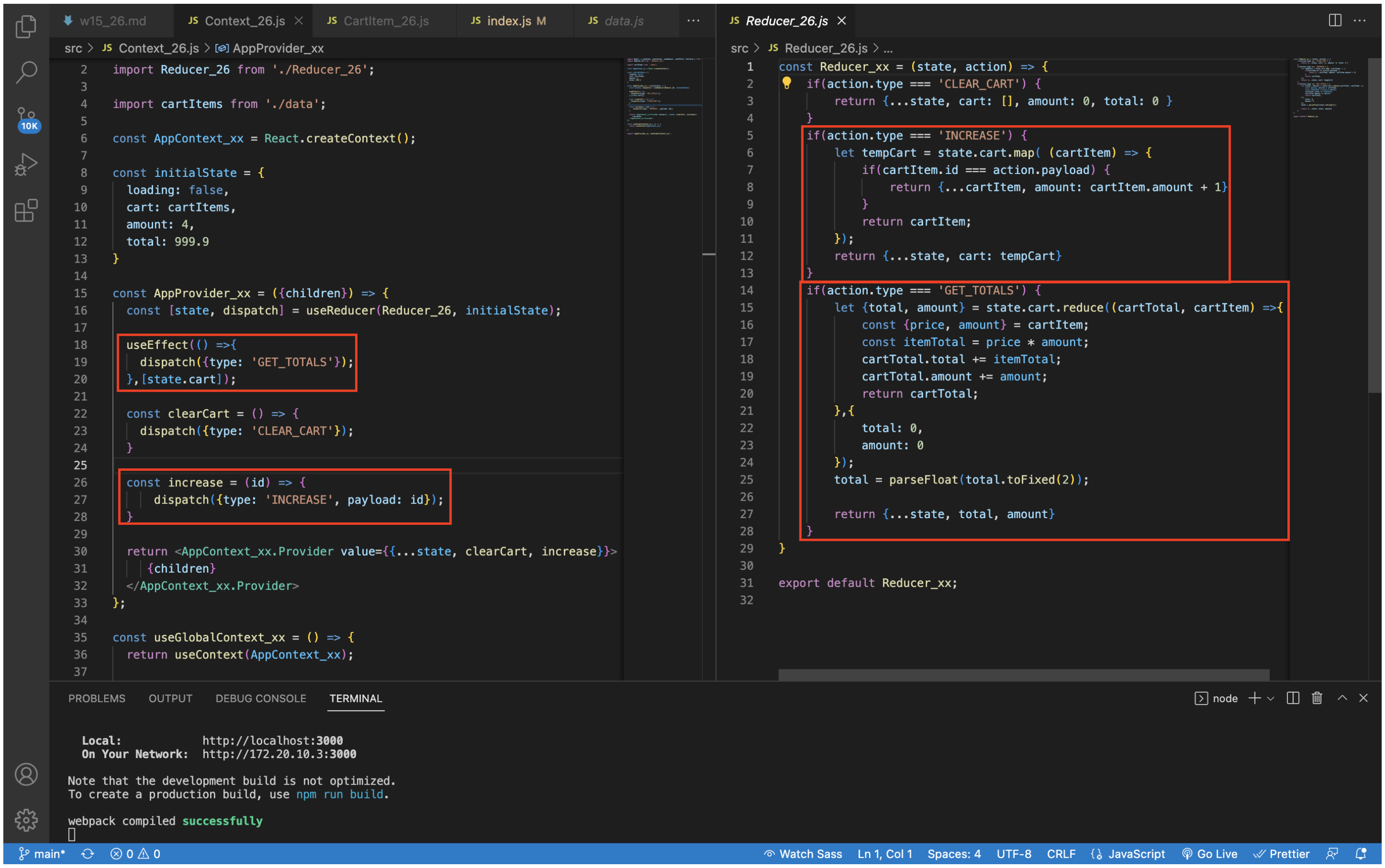
YOUR BAG

is currently empty

```
▾ AppProvider_xx
  ▾ Context.Provider
    ▾ App_26
        Navbar
        CartContainer
```

```
CartContainer

props
  new entry: ""

hooks
  ▾ GlobalContext_xx:
    ▾ Context: {amount: 0, cart: Array(0), c…
        loading: false
        cart: []
        amount: 0
        total: 0
        clearCart: ƒ clearCart() {}
        increase: ƒ increase() {}

rendered by
  App_26
  createRoot()
  react-dom@18.2.0

source
```

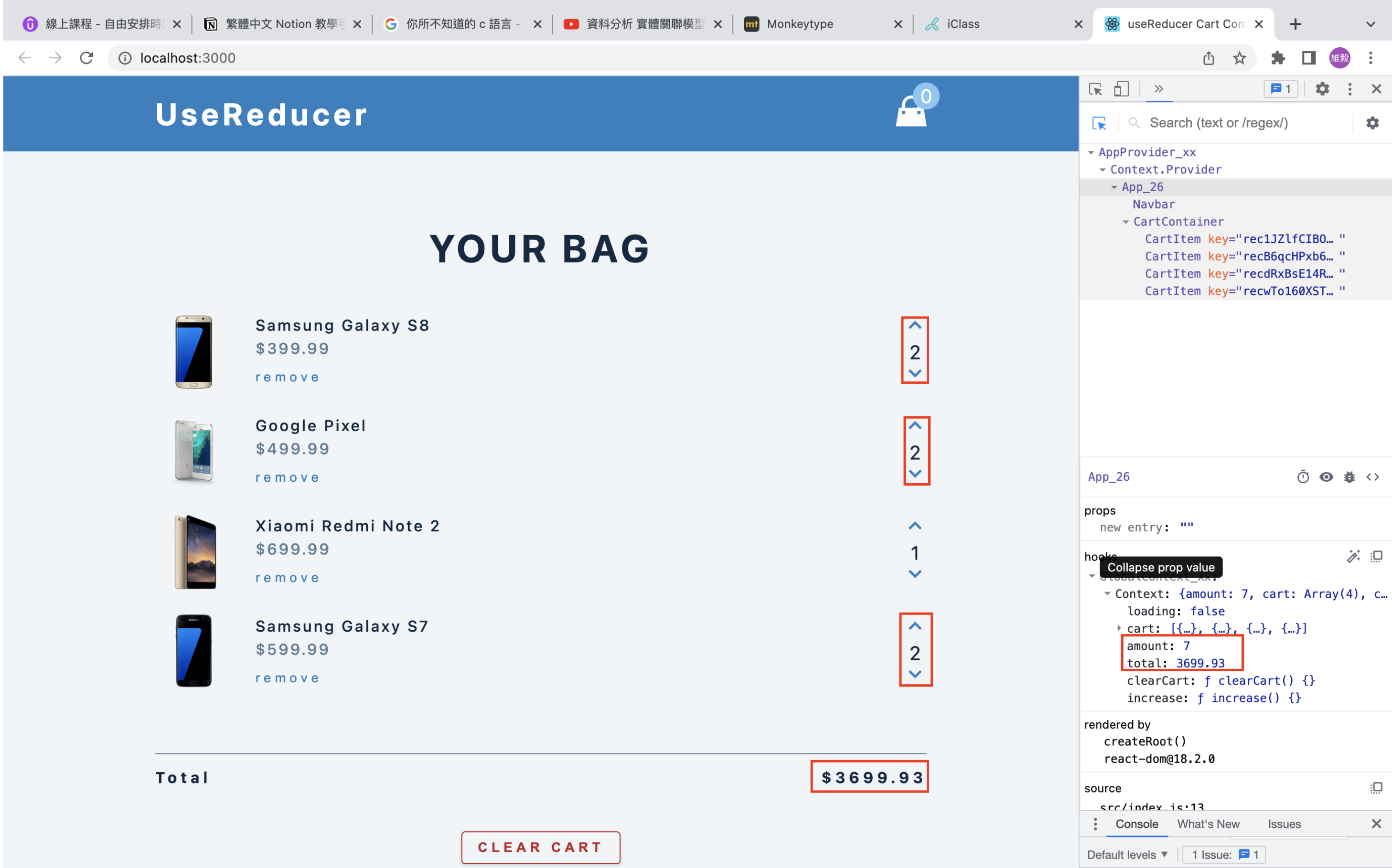
W15-P4: provide increase and get total function in global context



```
import Reducer_26 from './Reducer_26';

import cartItems from './data';

const AppContext_xx = React.createContext();

const initialState = {
  loading: false,
  cart: cartItems,
  amount: 4,
  total: 999.9
}

const AppProvider_xx = ({children}) => {
  const [state, dispatch] = useReducer(Reducer_26, initialState);

  useEffect(() =>{
    dispatch({type: 'GET_TOTALS'});
  },[state.cart]);

  const clearCart = () => {
    dispatch({type: 'CLEAR_CART'});
  }

  const increase = (id) => {
      dispatch({type: 'INCREASE', payload: id});
  }

  return <AppContext_xx.Provider value={{...state, clearCart, increase}}>
      {children}
    </AppContext_xx.Provider>
};

const useGlobalContext_xx = () => {
  return useContext(AppContext_xx);

```

```
const Reducer_xx = (state, action) => {
    if(action.type === 'CLEAR_CART') {
        return {...state, cart: [], amount: 0, total: 0 }
    }
    if(action.type === 'INCREASE') {
        let tempCart = state.cart.map( (cartItem) => {
            if(cartItem.id === action.payload) {
                return {...cartItem, amount: cartItem.amount + 1}
            }
            return cartItem;
        });
        return {...state, cart: tempCart}
    }
    if(action.type === 'GET_TOTALS') {
        let {total, amount} = state.cart.reduce((cartTotal, cartItem) =>{
            const {price, amount} = cartItem;
            const itemTotal = price * amount;
            cartTotal.total += itemTotal;
            cartTotal.amount += amount;
            return cartTotal;
        },{
            total: 0,
            amount: 0
        });
        total = parseFloat(total.toFixed(2));

        return {...state, total, amount}
    }
}

export default Reducer_xx;

```

```
  Local:            http://localhost:3000
  On Your Network:  http://172.20.10.3:3000

Note that the development build is not optimized.
To create a production build, use npm run build.

webpack compiled successfully
```



UseReducer

0

YOUR BAG

Samsung Galaxy S8
$399.99
remove
2

Google Pixel
$499.99
remove
2

Xiaomi Redmi Note 2
$699.99
remove
1

Samsung Galaxy S7
$599.99
remove
2

Total $3699.93

CLEAR CART

```
AppProvider_xx
  Context.Provider
    App_26
      Navbar
      CartContainer
        CartItem key="rec1JZlfCIBO… "
        CartItem key="recB6qcHPxb6… "
        CartItem key="recdRxBsE14R… "
        CartItem key="recwTo160XST… "

App_26
props
  new entry: ""
hooks
  GlobalContext_xx:
    Context: {amount: 7, cart: Array(4), c…
      loading: false
      cart: [{…}, {…}, {…}, {…}]
      amount: 7
      total: 3699.93
      clearCart: ƒ clearCart() {}
      increase: ƒ increase() {}
rendered by
  createRoot()
  react-dom@18.2.0
source
  src/index.js:13
```

W15-P5:

```
851f3c0 WeiYiHuangfelix Thu Dec 15 15:54:53 2022 +0800  W15-P4: provide increase and get total function
416fbad WeiYiHuangfelix Thu Dec 15 15:54:26 2022 +0800  W15-P3: put clear cart function in global conte
45e439e WeiYiHuangfelix Thu Dec 15 15:53:52 2022 +0800  W15-P2: replace data in CartContainer using glo
239a35b WeiYiHuangfelix Thu Dec 15 15:52:56 2022 +0800  W15-P1: Setup AppProvider_xx for App_xx and get
```