Plan de test

Main.js

Une image contenant texte

Description générée automatiquement

Une image contenant texte

Description générée automatiquement

La fonction DisplayListeAricles : fonctionnera si fetch s’exécute et génère un fichier RESPONSE avec les données attendues.

Product.js

Une image contenant texte

Description générée automatiquement

GetProductById : si recuperationIdProduct a bien généré un ID.

Tester en donnant une valeur à ID directement.

Une image contenant texte

Description générée automatiquement

DisplayProductDetails : si fetch s’exécute et génère un fichier RESPONSE.json avec

Les données demandées.

Panier.js

Une image contenant texte

Description générée automatiquement

articleAjoute a le bon article si :

«  LocalStorage .setItem ( 'Product' ,JSON.stringify ( articlePanier ) ) «  🡺

Dans product.js Ligne 95 a sauvegardé les variables du bon produit.

Une image contenant texte

Description générée automatiquement

afficherArticleSelectionné : si articleAjoute a les variables du bon produit.

* Remplacer i par une valeur de 0 à 4 pour valider un produit précis.
* Remplacer chaque variable ex : articleAjoute[i].name par Arnold.

Une image contenant texte

Description générée automatiquement

L’affichage du montant total est juste: si articleAjoute[i].price est le bon prix 🡺

« articlePanier = JSON.parse ( articleAjoute ) ; » product.js Ligne 89.

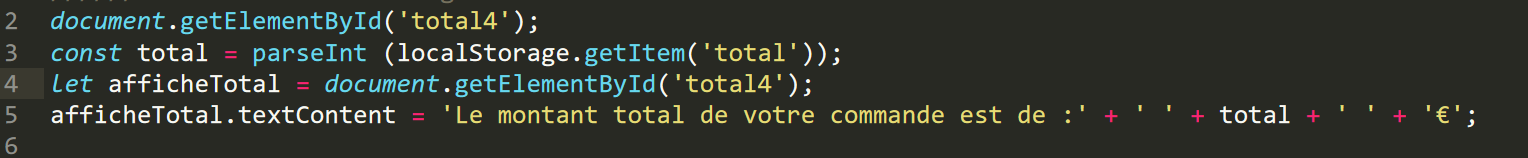
Remplacer articleAjoute[i].price par une valeur quelconque.

Une image contenant texte

Description générée automatiquement

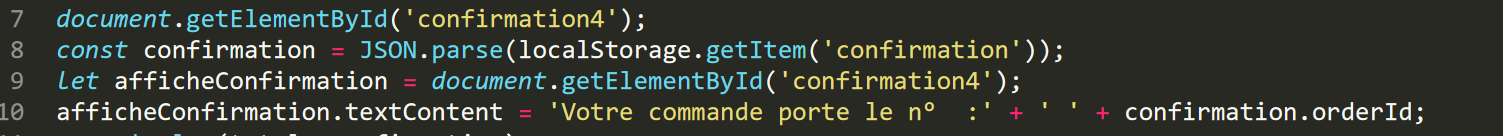
La réceptions du IdOrder(confirmation) : fetch exécutée avec méthode POST.

Confirmation.js



Le prix total si :

« localStorage.setItem("total", montantGlobal ) » dans panier.js Ligne 83 possede le bon montant global.



Confirmation  si :

“localStorage.setItem("confirmation", JSON.stringify(data))” dans panier.js Ligne 128 possede le bon orderID.