

| Fichier JS | Lignes de code testées | Fonction testée | Résultat attendu | Comment vérifier le résultat attendu | Problème possible | Commit | Resultat |
|------------|------------------------|---|--|--|---|---------|----------|
| Index.js | 9 à 18 | Function noname | La fonction doit récupérer les produits | On peut verifier grâce à un console.log | Erreur sur fonction | Init | Ok |
| Index.js | 20 à 43 | Function getProducts | Récupérer les données produits via fetch et toutes les données json | On peut vérifier grâce à l'inspecteur et à console.log | Pas de retour de données | Test 1 | Ok |
| index.js | 36 à 42 | Function error | Afficher une erreur au cas où il y a une erreur | Débrancher l'api et un console.log | Pas d'affichage de l'erreur | Test 2 | Ok |
| Index.js | 28 à 30 | Function products | Recuperation de tous les produits | Grâce à console.log si les données produits sont retourné | Aucune info disponible | Test 3 | ok |
| index.js | 46 à 60 | Function displayProduct | Affichage des produits en créant des templates automatiquement selon les produit grâce aux id des produits, les données récupérés pourront remplir les champs du template | Grâce à console.log et à l'inspecteur | Pas de création de Template | Test 4 | ok |
| produit.js | 5 à 12 | Function general | On créer une fonction no name pour récupérer l'id du produit intégré dans l'uri personnalisé de la page | Grâce à l'inspecteur et console.log | Pas de récupération de l'id | Test 7 | ok |
| produit.js | 14 à 40 | FunctiongetProductid | On a récupéré l'id du produit choisi | Grâce à l'inspecteur et console.log | Pas de récupération de l'id | Test 7 | ok |
| produit.js | 42 à 65 | Function hydrateProduct | On récupère les information du produit choisi pour les afficher sur la page produit | Grâce au console.log et à l'inspecteur en regardant si les infos récupérées correspond au produit choisi et que le produit s'affiche | Le chemin du produit non récupéré et non reconnu | Test 7 | ok |
| produit.js | 58 à 65 | Let colorSelection | On créer un menu déroulant où l'on pourra sélectionner la couleur du produit | En cliquant sur le menu et en choisissant une couleur | Erreur dans l'écriture, couleur non reconnu, | Test 9 | ok |
| produit.js | 71 à 73 | Const ProductId | Récupère l'id du Produit | Grâce au console.log et à l'inspecteur | Pas de récupération de données ou données erronés | Test 10 | ok |
| produit.js | 71 à 73 | Const ProductId | Récupère l'id du Produit sélectionné pour l'envoyer au panier | Grâce au console.log et à l'inspecteur | Pas de récupération de données ou données erronés | Test 10 | ok |
| produit.js | 68 à 69 | Const ProductName | Récupère le nom du Produit pour l'envoyer au panier | Grâce au console.log et à l'inspecteur | Pas de récupération de données ou données erronés | Test 10 | ok |
| produit.js | 77 à 78 | Const ProductColor | Récupère l'id de la couleur du Produit pour l'envoyer au panier | Grâce au console.log et à l'inspecteur | Pas de récupération de données ou données erronés | Test 10 | ok |
| produit.js | 99 à 100 | Const ProductChoice | Récupère la couleur choisi du Produit pour l'envoyer au panier | Grâce au console.log et à l'inspecteur | Pas de récupération de données ou données erronés | Test 10 | ok |
| produit.js | 77 à 78 | Const ProductQuantity | Récupère la quantité commandé pour l'envoyer au panier | Grâce au console.log et à l'inspecteur | Pas de récupération de données ou données erronés | Test 10 | Ko |
| produit.js | 105 à 106 | Const ProductPriceQuantity | Récupère le prix du ou des Produit pour l'envoyer au panier | Grâce au console.log et à l'inspecteur | Pas de récupération de données ou données erronés | Test 10 | ok |
| produit.js | 83 à 84 | Const ProductToCart | Apparition du lien ajout au panier pour l'envoyer le produit au panier | Grâce au console.log et à l'inspecteur | Pas de récupération de données ou données erronés | Test 10 | ok |
| produit.js | 89 à 177 | ProductToCart.addEventListener('click', (event) => {}) | Ajout du produit au panier au clic sur le lien on envoie toutes les données | Grâce au console.log et à l'inspecteur | Erreur dans le code ou doutbon lors de la commande sur le panier | Test 11 | Ko |
| produit.js | 96 à 97 | Const ProductNameChoice | Ajout du nom du produit acheté au panier | Grâce au console.log et à l'inspecteur | Pas de récupération de données ou données erronés | Test 11 | Ok |
| produit.js | 99 à 100 | Const ProductChoice | Ajout de la couleur produit acheté au panier | Grâce au console.log et à l'inspecteur | Pas de récupération de données ou données erronés | Test 11 | Ok |
| produit.js | 102 à 103 | Const ProductQuantityChoice | Ajout de la quantité du produit choisi par l'utilisateur dans le panier | Grâce au console.log et à l'inspecteur | Pas de récupération de données ou données erronés | Test 11 | Ko |
| produit.js | 105 à 106 | Const ProductPriceQuantity | Ajout du prix du produit selon la quantité commandé | Grâce au console.log et à l'inspecteur | Pas de récupération de données ou données erronés | Test 11 | Ok |
| produit.js | 109 à 115 | Let ProductOptions | Création d'un objet contenant les données addProductNameChoice, asProductChoice, addProductQuantityChoice, addProductPriceQuantity, addProductId Creation du local storage verification s'il est vide. On va lire la key produit via getitem, convertir les données js présentent dans le localstorage en json grâce à json.parse(localStorage.getItem('product')). On va push les produits à ajouter et faire un « localStorage.setItem('product', JSON.stringify(localStorage.getItem('product'))) » pour les ajouter dans le localstorage | Grâce au console.log et à l'inspecteur | Pas de récupération de données ou données erronés | Test 11 | Ok |
| produit.js | 124 à 176 | Let localStorageInit | | Grâce à l'application localStorage | Erreur dans le code ou pas de récupération de données ou données erronés | Test 12 | Ok |
| produit.js | 129 à 138 | Const messageConfirm | Ajout d'un message de confirmation avec deux boutons pour soit être redirigé vers le panier ou vers l'accueil | Lors de la commande et de l'ajout au panier | Erreur dans le code | Test 14 | Ok |
| Panier.js | 124 à 176 | Let products = JSON.parse(localStorage.getItem('product')) | On récupère les données du local storage en vérifiant si il est vide et en convertissant les données en json (voir ligne 25 du plan de test) | Grâce à l'application localStorage | Erreur dans le code ou pas de récupération de données ou données erronés | Test 15 | Ok |
| Panier.js | 10 à 27 | Const selectionContenuPanier | On sélectionne le template html pour le preparer à recevoir les données du panier | Grâce à l'application localStorage | Erreur dans le code html ou pas de récupération de données ou données erronés | Test 16 | Ok |
| Panier.js | 28 à 39 | Const selectionContenuPanier | Affichage de l'array avec les produits dans le panier qui sont normalement reconnu | Grâce à l'application localStorage | Erreur dans le code html ou pas de récupération de données ou données erronés | Test 17 | Ok |
| Panier.js | 24 à 39 | For (j = 0; j < localStorageInit.length; j++) | Affichage en boucle des produits selon la quantité reconnu | Grâce à l'application localStorage et la page panier et console.log | Ne s'affiche pas correctement à l'écran | Test 18 | Ko |
| Panier.js | 42 à 44 | Let suppressionProduitPanier | Possibilité de supprimer un produit du panier en ajoutant un lien effacer | Console.log | Bouton qui ne s'affiche pas | Test 19 | ok |
| Panier.js | 47 à 53 | for (let k = 0; k < suppressionProduitPanier.length; k++) | Possibilité de supprimer un produit du panier | Grâce au localStorage mais aussi voir sur l'écran sur l'écran du panier le produit s'efface correctement | Ne s'efface pas ou bouton qui ne marche pas | Test 20 | Ok |
| Panier.js | 55 à 73 | for (let k = 0; k < suppressionProduitPanier.length; k++) | Filtre les produits à supprimer selon selection se met à jour afin de créer un nouvel array | Grâce au localStorage mais aussi voir sur l'écran sur l'écran du panier le produit s'efface correctement | Ne s'efface pas ou bouton qui ne marche pas ou efface tout | Test 21 | ok |
| Panier.js | 75 à 79 | const viderProduitPanier | creation du lien vider le panier | Grâce au console.log | Bouton qui ne s'affiche pas ou qui ne répond pas | Test 22 | ok |
| Panier.js | 84 à 85 | Const viderProduitPanierLink | Recuperation des infos len vider panier | Grâce à console.log et à l'inspecteur | Erreur dans le code html ou pas de récupération de données ou données erronés | Test 23 | ok |
| Panier.js | 90 à 100 | viderProduitPanierLink.addEventListener('click', (event) => {}) | Supprimer la clé du local storage pour vider le panier, vider le local storage et afficher un message de confirmation du panier vide | Grâce à console.log et à l'inspecteur | Erreur dans le code html ou pas de récupération de données ou données erronés | Test 23 | ok |
| Panier.js | 102 à 105 | Let panierRemplirPrixTotal = [] | Calcul total du panier en récupèrent les prix des produits grâce à for (let i = 0; i < localStorageInit.length; i++) | Grâce à console.log et à l'inspecteur | Erreur dans le code html ou pas de récupération de données ou données erronés | Test 24 | ok |
| Panier.js | 109 à 115 | panierRemplirPrixTotal.push(panierRemplirPrix) | On va utiliser push pour rajouter les elements dans le panier afin de pouvoir faire le calcul du prix total | Grâce à console.log et à l'inspecteur | Erreur dans le code html ou pas de récupération de données ou données erronés ou quantité negative commandé | Test 25 | ok |
| Panier.js | 116 à 117 | panierRemplirPrixTotal = panierRemplirPrixTotal.map(x) => parseFloat(x)); | On transforme les prix du format euros au format nombre pour le calcul | Grâce à console.log et à l'inspecteur | Erreur dans le code html ou pas de récupération de données ou données erronés | Test 26 | ok |
| Panier.js | 119 à 121 | Const reducer couplé à const prixTotalPanier | On va utiliser reducer pour cumuler le total de tous les produits sélectionnés et du coup obtenir le résultat du prix total du panier | Grâce à console.log et à l'inspecteur | Erreur dans le code html ou pas de récupération de données ou données erronés | Test 26 | ok |
| Panier.js | 128 à 130 | Const afficherPrixTotalPanier | Affichage du prix total du panier | Regarder si le total s'affiche bien sur l'écran | Erreur dans le code html ou pas de récupération de données ou données erronés | Test 26 | ok |

| | | |
|--|--|--|
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |