

Fichier JS	Lignes de code testées	Fonction testée	Résultat attendu	Comment vérifier le résultat attendu	Problème possible
index.js	4 à 6	Fetch : localhost:3000/api/teddies .then((res) => res.json()) =>	Ici la promesse initier par Fetch attend une réponse du serveur qui signifie que nous communiquons avec lui. Dans les outils de développement aller voir dans l'onglet réseau si le résultat de la méthode GET est 200	Poser une condition : if(response.ok){console.log('communication avec le serveur')} ou dans les outils de développement et aller voir dans l'onglet réseau si le résultat de la méthode GET est 200	Le problème possible serait qu'il n'y ait pas communication avec le serveur
index.js	4 à 6	Fetch : localhost:3000/api/teddies .then((res) => res.json()).then((data) =>	Ici la promesse initier par Fetch va nous permettre de recueillir et d'afficher le tableau des données stocker dans le serveur.	Ici un console.log(data) à l'intérieur de la fonction data devrait nous renvoyer le tableau complet existant dans le serveur, dans ce cas là le tableau contenant les 5 articles. Ici les 5 oursours en peluches ayant Sids différents.	Le problème possible serait que le tableau ne soit pas renvoyer
produit.js	9 à 19	Fonction searchId	Le résultat attendu est la récupération de l'id de du produit sélectionné dans l'URL de la page html de l'oursoun consulté	un console.log(id) permet de vérifié si on bien récupéré le bon id produit	Le problème serait que l'id de l'url ne soit pas récupéré
produit.js	84 à 92	fonction addCart	Le tableau(le produit sélectionné) est ajouté au localStorage('cartId')	Aller voir dans le localStorage des outils de développement si le produit a été ajouté ou effectuer un console.log(localStorage.getItem('cartId')) dans panier.js	le produit pourrait ne pas être ajouté au localStorage('cartId')
produit.js	103 à 114	fonction addCartId	L'id du produit sélectionné est ajouté au localStorage('ids')	Aller voir dans le localStorage des outils de développement si le produit a été ajouté ou effectuer un console.log(localStorage.getItem('ids')) dans panier.js	l'id pourrait ne pas être ajouté au localStorage('ids')
produit.js	125 à 135	fonction addCartPrice	L'id du produit sélectionné est ajouté au localStorage('prices')	Aller voir dans le localStorage des outils de développement si le produit a été ajouté ou effectuer un console.log(localStorage.getItem('prices')) dans panier.js	l'id pourrait ne pas être ajouté au localStorage('prices')
panier.js	9 à 16	OP.forEach(item)	Les produits sélectionnés seront affichés les uns après les autres	console.log(item) dans la fonction	Les différents items(oursours) pourraient ne pas avoir avoir été sélectionnés
panier.js	87 à 97	function montantTotal()	Effectue la somme des prix de chaque produit	console.log(total) dans la fonction est au besoin une calculatrice pour la vérification	Le montant total pourrait être erroné
confirmation.js	5 à 8	function product_id)	Les ids sont collectés et sont ensuite ajoutés à l'aide de pusher dans le tableau products	console.log(_id) dans la fonction	L'id pourrait ne pas être ajouté au au tableau products dans le localStorage('formulaire ') avant l'envoi au serveur
confirmation.js	160 à 172	function formulaireFetch()	L'envoi de la requête POST au serveur doit retourner le numéro de confirmation de commande : _idOrder	Poser une condition : if(response.ok){console.log('communication avec le serveur')} ou dans les outils de développement et aller voir dans l'onglet réseau si le résultat de la méthode GET est 200. <b>Au préalable s'assurer qu'on envoie le bon format de requête au serveur pour qu'il puisse renvoyer sa réponse. S'il y a communication avec le serveur et que le bon format POST est envoyé, le serveur renvoie le IdOrder dans un tableau dans lequel se trouve également un récapitulatif de la commande</b>	la requête post pourrait ne pas être reçu par le serveur
confirmation.js	187 à 197	function saveIdOrderInformation()	Sauvegarde des information renvoyé par le serveur dont le numéro de confirmation de commande	Aller voir dans le localStorage des outils de développement si le produit a été ajouté ou effectuer un console.log(localStorage.getItem('orderIdInformation')) dans confirmationClear.js	le numéro de commande orderId pourrait ne pas être stocké
confirmationClear.js	3 à 14	function closeAll()	une fois le bouton merci et à bientôt cliqué, les données d'achat sont effacées via localStorage.removeItem('Keys').	Aller voir dans le localStorage des outils de développement si les clefs et valeurs ont été effacées	le stockage des données de l'achat pourrait ne pas être effacés du localStorage et en cas de retour sur le site l'ancien panier serait toujours présent

---