

Fichier JS	Lignes de code testées	Fonction testée	Résultat attendu	Comment vérifier le résultat attendu	Problème possible
page_accueil.js	2 à 23	creerCarteProduit	La fonction doit créer une structure HTML pour dans le contenant "liste_produit" de la page d'accueil	Appeler la fonction sur la page d'accueil et observer la création de la carte produit	le contenant pourrait ne pas exister, il n'y aurait pas de création
page_accueil.js	29 à 42	afficherNomEtPrixNounours	La fonction prend en argument un tableau d'objets au format [{"colors": "", "id": "", "name": "", "price": "", "imageUrl": "", "description": ""}...] et pour chaque objet appelle la fonction creerCarteProduit et doit ajouter le nom et le prix de l'objet dans les éléments HTML ayant pour classe respectivement "carte_text_titre" et "carte_text_prix"	appeler la fonction avec en argument un tableau d'objet correspondant au format et observer le rendu sur le navigateur	objet non conforme au format donc pas de donnée à afficher
page_accueil.js	44 à 49	afficherPhotoLienNounours	La fonction prend en argument un tableau d'objets au format [{"colors": "", "id": "", "name": "", "price": "", "imageUrl": "", "description": ""}...] et pour chaque objet doit ajouter l'uri de l'image de l'objet dans l'élément HTML ayant pour classe "carte_image" et ainsi que l'id sous la forme "/pagesHTML/page_produit.html?id= + l'id en question	appeler la fonction avec en argument un objet correspondant au format et observer le rendu sur le navigateur	objet non conforme au format donc pas de donnée à afficher
page_accueil.js	61 à 65	afficherListeProduits	La fonction prend en argument un tableau d'objets au format [{"colors": "", "id": "", "name": "", "price": "", "imageUrl": "", "description": ""}...] et appelle les fonctions afficherNomEtPrixNounours ainsi que afficherPhotoLienNounours avec ce même argument	appeler la fonction avec une liste d'article conforme à celle de l'API et observer l'affichage sur le navigateur	liste d'article non conforme au format de l'API, donc pas de création de liste d'article
page_accueil.js	66 à 70	fillProducts	La fonction demande à l'API la liste des produits et applique la réponse à la fonction afficherListeProduits	appeler la fonction et placer un console.log du retour de l'API	adresse de l'API incorrecte et donc pas de communication possible avec l'API
page_produit.js	1 à 15	adressellUnique	la fonction doit récupérer le paramètre "id" de l'URL et retourner une URL de type 'http://localhost:3000/api/eddies' + idProduit	appeler la fonction et placer un console.log du résultat	pas de paramètre à récupérer dans l'URL et donc adresse incomplète pour la requête à l'API
page_produit.js	23 à 27	afficherNom	La fonction prend en argument un objet au format {"colors": "", "id": "", "name": "", "price": "", "imageUrl": "", "description": ""} et doit ajouter le nom de cet objet à l'élément HTML ayant pour classe 'titre_ourson'	appeler la fonction avec en argument un objet correspondant au format et observer le rendu sur le navigateur	objet non conforme au format donc pas de donnée à afficher
page_produit.js	29 à 33	afficherPrix	La fonction prend en argument un objet au format {"colors": "", "id": "", "name": "", "price": "", "imageUrl": "", "description": ""} et doit ajouter le prix de cet objet à l'élément HTML ayant pour classe 'prix_ourson'	appeler la fonction avec en argument un objet correspondant au format et observer le rendu sur le navigateur	objet non conforme au format donc pas de donnée à afficher
page_produit.js	35 à 39	afficherDescription	La fonction prend en argument un objet au format {"colors": "", "id": "", "name": "", "price": "", "imageUrl": "", "description": ""} et doit ajouter la description de cet objet à l'élément HTML ayant pour classe 'description_ourson'	appeler la fonction avec en argument un objet correspondant au format et observer le rendu sur le navigateur	objet non conforme au format donc pas de donnée à afficher
page_produit.js	41 à 44	afficherPhotoDesNounours	La fonction prend en argument un objet au format {"colors": "", "id": "", "name": "", "price": "", "imageUrl": "", "description": ""} et doit ajouter l'uri de l'image de cet objet à l'élément HTML ayant pour classe 'image_ourson'	appeler la fonction avec en argument un objet correspondant au format et observer le rendu sur le navigateur	objet non conforme au format donc pas de donnée à afficher
page_produit.js	46 à 54	afficherStyleNounours	La fonction prend en argument un objet au format {"colors": "", "id": "", "name": "", "price": "", "imageUrl": "", "description": ""} et doit ajouter le style de cet objet à l'élément HTML ayant pour classe 'option'	appeler la fonction avec en argument un objet correspondant au format et observer le rendu sur le navigateur	objet non conforme au format donc pas de donnée à afficher
ajout_panier.js	1 à 14	recuperationId	La fonction doit récupérer le paramètre id de L'URL et doit le retourner	appeler la fonction et placer un console.log pour observer l'id récupéré	l'id produit n'existe pas donc il n'y a pas de retour de la fonction
ajout_panier.js	23 à 31	verificationQuantiy	La fonction prend un argument et doit vérifier que c'est un nombre compris entre 1 et 100. Si oui, elle retourne "true" sinon "false"	appeler la fonction avec un argument et placer un console.log pour capter le retour	les valeurs de retour peuvent être inversées
ajout_panier.js	42 à 49	isInCart	La fonction prend un argument et cherche à savoir si c'est une clé du localStorage. Si oui, la fonction retourne "true" sinon elle retourne "false"	préremplir le localStorage et le passer en argument dans la fonction. On capture le retour via un console.log	
ajout_panier.js	51 à 60	isColorAlreadyChosen	La fonction prend en arguments une couleur String et un idProduit String. Pour tout les éléments du localStorage qui ont pour clé IdProduit, on vérifie si la couleur est présente. Si oui la fonction retourne "true", sinon elle retourne "false".	remplir le localStorage avec idproduit: {"color": ""}, appeler la fonction avec un id et une couleur et capter le résultat avec un console.log	si l'idproduit n'est pas dans le localStorage, il ne peut y avoir de comparaison
page_panier.js	1 à 7	creerElement	la fonction prend en paramètre un id de contenant HTML, une balise HTML, une à trois classes tous de type String et crée un élément HTML enfant dans le contenant avec les classes renseignées	Dans une page HTML, appeler la fonction avec un contenant, une balise et une classe et observer la création de l'enfant sur le navigateur	si les arguments sont null, aucune création ne se fait
page_panier.js	9 à 16	afficherPanierVide	la fonction récupère le premier élément du localStorage et si il est égal à null, affiche "le panier est vide" dans un élément HTML de classe "paragraph-prix-total"	charger la page panier avec un localStorage vide et capter la réponse de la fonction avec un console.log	si le localStorage contient des éléments même non-conforme, la fonction ne fait rien
page_panier.js	36 à 40	afficherPrixTotal	la fonction prend en paramètre un nombre et l'affiche dans l'élément HTML de class "prix-total"	appeler la fonction avec un argument et observer le résultat dans la page panier.html	si l'argument n'est pas un nombre, le prix affiché est NAN
page_panier.js	42 à 60	remplirStructureArticle	la fonction prend en argument un objetAPI {"colors": "", "id": "", "name": "", "price": "", "imageUrl": "", "description": ""}, un objetPanier {"color": "", "quantity": ""} et nombre. Elle récupère le name, imageUrl et le price de l'objetAPI ainsi que la couleur et la quantity de l'objetPanier pour les ajouter dans les élément HTML correspondants. Elle appelle également la fonction afficherPrixTotal	appeler la fonction et observer le resultat sur la page.html	si l'argument est null ou pas au bon format, rien ne se créer
page_panier.js	62 à 68	recuperationId LocalStorage	la fonction va chercher les id stocké dans le local storage et les retourne dans un tableau	appeler la fonction et capturer la réponse avec un console.log	
valider_panier.js	3 à 13	recuperationPanier	la fonction récupère les clés du localStorage et les retourne dans un tableau suivant leur quantité	préremplir le localStorage avec une clé: {"quantity": integer}. On appelle la fonction et on capture le retour via un console.log	si la quantité dans le localStorage n'existe pas. La fonction retourne une erreur
valider_panier.js	16 à 30	recuperation Coordonnees	la fonction récupère les données du formulaire (nom, prénom, adresse, mail, ville) et les retourne dans un objet = {firstName: prénom, lastName: nom, email: adresseMail, address: adressePostale, city: ville}	appeler la fonction sur la page panier avec un formulaire prérempli et capturer la réponse avec un console.log	si les input n'existe pas, la fonction ne renvoi rien
valider_panier.js	33 à 40	formatageDesInfo PourAPI	la fonction prend en argument deux objets et retourne un String comme suit {"contact": argument1, "products": argument2}	appeler la fonction avec en argument deux objets et capturer la réponse avec un console.log	
valider_panier.js	44 à 53	extraireIdCommande	la fonction prend en argument un objet {"orderid": "", "products": ""} et retourne un tableau [idecommande, prixtotal]	appeler la fonction avec en argument un objet correspondant au retour de l'API et capturer le résultat	si l'argument est nul, la fonction ne peut rien retourner
valider_panier.js	55 à 59	redirectionPage Confirmation	la fonction prend en argument un objet {"orderid": "", "products": ""}, appelle la fonction extraireIdCommande avec cet argument. Elle redirige le navigateur vers la page de confirmation en ajoutant le retour de la fonction comme paramètre	appeler la fonction avec en argument un objet correspondant au retour de l'API et capturer l'URL générée	si l'argument est nul ou non conforme, la fonction ne peut rien retourner
valider_panier.js	88 à 97	checkFormulaire	la fonction prend en argument une classe CSS String et un regex vérificateur. Elle retourne True si la valeur liée à la classe est conforme au regex.	appeler la fonction avec en argument une classe d'un input à vérifier et un regex.	si la classe ou le regex n'est pas existant ou un regex, la fonction renvoi une erreur et ne fait rien
page_confirmation.js	5 à 24	recupererPrixEtId	la fonction récupère 2 paramètres de l'URL (idcommande et total) et les ajoute au élément HTML correspondant de classe "ref-commande" et "prix-total"	appeler la fonction sur une page de confirmation vierge en ajoutant des paramètres. On observe la modification de la page	si les éléments.html n'existe pas, la fonction renvoi une erreur.