

Fichier JS	Lignes de code testées	Fonction testée	Résultat attendu	Comment vérifier le résultat attendu	Problème possible
panier.js	178 à 217	validLastName()	La fonction doit retourner 'true' si l'input est rempli selon plusieurs critères en même temps : au moins 2 caractères et écrit en Majuscule et 'false' si ces critères ne sont pas respectés.	On saisit le nom dans l'input (nom saisi selon les critères demandés), et on observe la valeur retournée, avec un console.log par exemple	La valeur retournée pourrait être erronée (true au lieu de false; et false au lieu de true).
panier.js	221 à 249	validAdresse()	La fonction doit retourner le message "le format d'adresse n'est pas valide" si l'input n'est pas rempli selon les critères de sa RegEx "^[0-9]+[.,[a-z]+/, "g" et 'false' si elle reçoit tout autre type de caractères ou nombres.	On saisit des mots dans l'input (un chiffre, suivi d'une virgule ensuite un nom de rue en minuscule et on observe la valeur retournée, avec un console.log par exemple.	La valeur retournée pourrait être erronée (true au lieu de false; et false au lieu de true).
panier.js	118 à 140	validEmail()	La fonction doit retourner 'true' si l'input est rempli selon les critères de sa RegEx "^[a-zA-Z0-9.-_]+@[a-zA-Z0-9.-_]+.[a-z]{2,10}\$", "g" et 'false' si elle reçoit tout autre type de caractères ou nombres.	On saisit des mots dans l'input (des adresses email utilisant des minuscule ou majuscule allant de a à z et/ou des chiffres de 0 à 9, des points, des _ , des - , des espaces, elles ne doivent contenir qu'un seul @, ensuite le nom de domaine avec les même critère, obligatoirement un point puis les initiales de fin) et on observe la valeur retournée, avec un console.log par exemple.	La valeur retournée pourrait être erronée (true au lieu de false; et false au lieu de true).
index.js	3 à 8	Fetch(API/Furniture)	la deuxième fonction then(furniture) doit récupérer tous les objets présents dans l'API	On fait un console.log(furniture) et on vérifie dans la console si tous les objets du tableau ont été récupérés.	Problème de permission, de réseau
index.js	3 à 24	Fetch(API/Furniture)	La première fonction then(response) nous permet de vérifier qu'on a bien une réponse, avant de pouvoir aller plus en avant et exploiter ce retour.	Fait un console.log(response) pour vérifier la présence d'une réponse en json	Problème de permission, de réseau, la fonction catch récupère l'erreur et l'affiche dans le console.log(erreur)
produit.js	13 à 15	Fetch(furniture/id)	Cette première fonction then(response) doit nous retourner une réponse qui correspond au produit choisi	On vérifie dans console.log(response) que l'id de l'objet sélectionné correspond bien à l'objet présent.	Il peut y avoir un problème de réseau ou de permission, ou alors on peut avoir un mauvais id
produit.js	13 à 18	Fetch(furniture/id)	Cette deuxième fonction then(furniture) doit nous retourner les éléments de l'objet sélectionner	On vérifie dans console.log(furniture) que les éléments présents, les options correspondent bien à l'objet sélectionné	Il peut y avoir des options qui manquent ou d'autres éléments de l'objet qui manquent
produit.js	57 à 86	function ajouterArticle() 'condition panier vide'	Cette fonction doit ajouter l'article sélectionné au panier	On peut regarder dans la partie inspecteur/application de la page web si un article a bel et bien été ajouter dans le localStorage	L'article peut être illisible dans le local storage si on oublie de faire un JSON.stringify(panier) au moment de l'envoi dans le panier. A ce moment-là on aura un format [object ,object].
utils.js	8 à 19	function showQuantity()	cette fonction doit calculer le nombre d'article total envoyé au panier	la quantité totale d'articles ajoutés doit être présente sur le site web, à côté du panier	La somme calculée n'est pas exacte, erreur dans la formule de calcul, ou par exemple, si on ne fait pas un "parseInt" on aura une concaténation des chiffres au lieu d'avoir un calcul.