

ORINOCO

Plan de test unitaires

Fichier	Fonction à tester	Paramètre d'entrée	Cas de test	Résultat attendu
main.js	createProductBloc 34/48	camera : objet : article camera comme indiqué dans la documentation container : HTMLElement : conteneur qui va afficher tous les produits	On passe un objet camera ainsi que le conteneur	Dans le nombre de childNodes du conteneur, on doit avoir un élément de plus.
product.js	displayProductSelected 70/115	cameraDescription : objet repreant les informations du produit sélectionné	On passe un objet camera en paramètre	Le produit doit s'afficher sur la page dans les éléments correspondants.
	addElementToCart 22/115	cameraDescription : objet repreant les informations du produit sélectionné	L'utilisateur ne personnalise pas le produit L'utilisateur sélectionne une lentille	Une alerte apparaît en indiquant à l'utilisateur de bien vouloir sélectionner une lentille. On vérifie que le nombre d'élément dans le localStorage est resté inchangé. Une alerte apparaît indiquant que le produit a bien été ajouté au panier. On vérifie dans le localStorage qu'un élément a bien été ajouté.
cart.js	displayItemsInCart 36/116	items : Array : liste d'articles container : HTMLElement : conteneur qui va afficher les éléments sélectionnés	On ajoute un article dans le panier depuis la page product	La fonction lance une boucle qui permet de parcourir le tableau contenant la liste des articles qui ont été ajoutés au panier, et les afficher dans un tableau ligne par ligne avec ses caractéristiques.
	displayTotalPrice 11/116	/	Des articles sont présents dans le panier	La fonction permet de rajouter une ligne dans le tableau indiquant le prix total de la commande.

confirmation.js	getTotalPrice 7/116	itemsInCart : tableau renvoyant les articles présents dans le panier	On passe les articles présents dans le panier	Une boucle permet de parcourir les articles ajoutés dans le panier et d'additionner le prix de l'ensemble de la commande.
	confirmOrder 43/116	/	Le formulaire doit être rempli avant de pouvoir valider la commande	Si le formulaire ne respecte pas le format demandé, une alerte apparaît et demande à l'utilisateur de remplir les champs correctement. Si le formulaire est bien rempli, les informations saisies par l'utilisateur sont envoyées via la méthode POST.
	setOrderInStorage 8 /116	orderConfirm	La commande est validée	L'id de commande, le nom et le prix total de la commande sont stockées dans le localStorage.
	orderInfo 11/15	On s'assure d'avoir en localStorage les éléments nécessaires à la validation d'une commande	A l'appel de la fonction, on doit pouvoir visualiser les informations de la commande.	La fonction permet de récupérer les informations du localStorage et d'afficher le numéro de commande généré, le nom du client saisi dans le formulaire et le montant total de la commande.
	function.js getCart 9/49	/	Les produits sont ajoutés dans le panier	Un tableau est stocké dans le localStorage et reprend les données du produit (nom, description, prix, choix à personnaliser).
	displayCartItemsNumber 5/49	/	On récupère les produits ajoutés au panier	Le nombre de produit ajouté est affiché dans le header de chaque page.
	checkInfo 29/49	/	Les Informations saisies manuellement par l'utilisateur ne sont pas au format attendu. Les informations saisies sont au bon format	Une alerte apparaît et demande à l'utilisateur de saisir les champs correctement. L'utilisateur ne peut pas aller plus loin. L'utilisateur peut saisir tous les champs du formulaire.