

N° Test	Fichier JS	Fonction testée	Lignes de code testées	Résultat attendu	Comment vérifier le résultat attendu	Problème possible
1	sendXHR.js	sendXHR()	Toutes	Cette fonction doit effectuer des requêtes type GET et POST à l'API. Elle suit le principe des API fetch, et donc hérite de leurs avantages et inconvénients	Appeler la fonction avec altération d'un ou des deux premiers paramètres requis. La gestion d'erreur a été écrite dans la fonction : code réponse > 400	Code > 400. Problème de sécurité CORS si les autres fonctions génèrent des erreurs dans leur propre exécution
2	updatePanierHeader.js	updatePanierHeader()	4 – 17	Actualise automatiquement le nombre d'article et le prix total pour les pages de site web. Effectuée en haut dans l'élément HEADER	Faire varier le nombre des articles et suivre l'actualisation dans toutes les pages.	Si la LOCALSTORAGE ne donne pas de valeurs cohérentes alors filtration de redirection altérée
3		updatePanierHeader()	22 – 37	Autorise les redirections vers le panier en cliquant sur l'icône panier ou prix dans le HEADER. Condition : LOCALSTORAGE non vide	1- LOCALSTORAGE non vide, s'assurer que le clic de l'icône depuis n'importe quelle page redirige vers la page panier 2- LOCALSTORAGE vide : vérifier que l'accès au panier est bloqué partout	Si la LOCALSTORAGE ne donne pas de valeurs cohérentes alors erreur filtration de redirection altérée
4	animationImage.js	imageAnimation()	5 – 18	Au survol de la souris, les images produites de la page vitrine doivent modifier leur opacité et leur échelles	Survoler avec la souris chacune des images. Et s'assurer du fonctionnement	Cette fonction s'exécute bien sur Firefox mais pas sur Chrome. Vérifier la portabilité. Fonction non vitale dans tous les cas
5	index.js	document.AddEventListener()	32 – 40	Cette fonction doit, dès le début de la construction de la DOM, appeler plusieurs autres fonctions : sendXHR() puis en cas de succès, afficherVitrine() et imageAnimation()	Cette fonction s'appelle à l'écoute d'un événement. En de succès de toute la chaîne de fonction, nous devons voir la page s'afficher correctement les articles	Si la DOM n'initie pas son chargement, alors l'événement déclencheur de ce bloc d'instructions ne démarrera pas
		afficherVitrine()	8 – 28	Cette fonction doit afficher toutes les caméras retournées par l'API. Elle s'exécute comme un succès à la suite d'une requête GET. Elle récupère les données reçues de la requête lancée au tout début du chargement de la page HTML	Appeler la fonction en lui passant les données reçues de l'API. Elle doit afficher à l'écran	Pas de données reçues de l'API (code ≥ 400) = pas d'affichage produit. Erreur dans le HTML et CSS. Erreur dans l'élément DIV d'accueil
6	index_filtration.js	requeteFiltration()	8 – 46	Filtration des données de prix reçues par l'API. Filtrations selon PRIX et NOM	Utiliser les changements de 'FORM' pour voir les différentes filtrations. Cette fonction exécute une nouvelle GET (avec sendXHR()) à chaque changement du menu	Pas de réponse du serveur. Pas de données reçues du serveur. Tous les avantages et les inconvénients des API fetch

N° Test	Fichier JS	Fonction testée	Lignes de code testées	Résultat attendu	Comment vérifier le résultat attendu	Problème possible
7	produit.js	document. AddEventListener()	20 – 33	Récupérer par un GET ,toutes les données relatives à une produit et ensuite les transmettre à une chaîne de fonctions pour affichage et manipulations	Cette fonction s'appelle à l'écoute d'un évènement. En cas de succès de toute la chaîne de fonction, nous devons voir la page s'afficher correctement.	Si la DOM n'initie pas son chargement, alors l'évènement déclencheur de ce bloc d'instructions ne démarrera pas
8	produit_afficher.js	afficherData()	1 – 48	Construit le HTML pour afficher les données pour un produit spécifique. Les données proviennent d'un GET	Appeler la fonction en lui passant les données reçus de l'API. Elle doit afficher à l'écran	Pas de données reçues de l'API (code ≥ 400) = pas d'affichage produit. Erreur dans le HTML et CSS . Erreur dans l'élément DIV d'accueil
9	produit_ajouterPanier.js	ajouterAuPanier()	8 – 55	Crée un tableau pour chaque article choisi ainsi qu'une clé spécifique et stocke l'information dans la LOCALSTORAGE	Faire CONSOLE.LOG / CONSOLE.TABLE pour vérifier le contenu de la LOCALSTORAGE	L'article n'est pas ajouté. L'article n'est pas actualisé.
10		afficherConfirmation()	58 – 84	Affiche un message de confirmation du produit ajouté ou actualisé au clic du bouton ajouter au panier	Faire CONSOLE.LOG pour s'assurer que LOCALSTORAGE ne soit pas vide	Le message de confirmation d'ajout ne s'affiche pas
11	produit_capteurs.js	capterOptionLentilles()	1 – 7	capter les options de lentilles faites par le client et tout changement	En cliquant sur le menu déroulant le changement des options de lentilles doit être capter. Utiliser CONSOLE.LOG() pour vérifier	Le HTML sous-jacent n'a pas été construit correctement
12		capterNombreArticleChoisi()	11 -17	capter le nombre d'article(s) choisi par le client et tout changement	En cliquant sur le menu déroulant le changement de la quantité d'article(s) doit être possible. Utilise CONSOLE.LOG() pour vérifier	Le HTML sous-jacent n'a pas été construit correctement
13		capterPrixTotal()	21 – 28	capter le prix total et tout changement	Le prix total doit s'afficher dynamiquement en suivant le changement du nombre d'articles. Utilise CONSOLE.LOG() pour vérifier.	Le HTML sous-jacent n'a pas été construit correctement. Aussi problème de multiplication entre STRING et INTEGER surtout après un INNERHTML
14	produit_creationOptions.js	creationOptionLentilles()	2 – 11	Le menu déroulant doit être cliquable, afficher autoriser et les différentes options de lentilles.	Utilise CONSOLE.LOG() pour vérifier.	Menu déroulant est figé
15		creationOptionQuantite()	15 – 24	Le menu déroulant doit être cliquable, afficher autoriser et les différentes options de quantité d'article.	Utilise CONSOLE.LOG() pour vérifier.	Menu déroulant est figé

N° Test	Fichier JS	Fonction testée	Lignes de code testées	Résultat attendu	Comment vérifier le résultat attendu	Problème possible
16	panier.js	afficherPanier()	10 – 39	Balaie LOCALSTORAGE, récupère et affiche correctement tous les éléments considérés comme article choisi par un client	Les articles choisis doivent s'afficher	Si la LOCALSTORAGE ne donne pas de valeurs cohérentes alors l'affichage sera altérée
17		window.addEventListener()	44 – 52	Fonction centrale pour l'affichage du panier. Au chargement terminé, elle appelle différentes fonctions pour assurer l'affichage complet de la page	Cette fonction s'appelle à l'écoute d'un événement. En cas de succès de toute la chaîne de fonctions, nous devons voir la page s'afficher correctement.	Si la DOM ne termine pas son chargement, alors l'événement déclencheur de ce bloc d'instructions ne démarrera pas. Une ou plusieurs fonctions présente(nt) des anomalies
18	panier_ajusterQuantite.js	ajusterQuantite()	6 – 42	Pour chaque article présent dans le panier, elle permet d'ajuster leur quantité et de le supprimer. Si LOCALSTORAGE est vide, renvoie à l'accueil	Utilisez CONSOLE.LOG() pour suivre l'actualisation d'un article, et la confirmation de la suppression	Tous les articles soient modifiés ou supprimés en même temps. Aucun article n'écoute cette fonction
19	panier_confirmerCommande.js	creerObjectContact()	10 – 19	Un objet contenant les informations du client	Utilisez CONSOLE.LOG() pour vérifier la création de l'objet	Cette fonction utilisant les données du formulaire de contact, et en subira les anomalies
20		creerTableauProduct()	24 – 37	Effectue des boucles dans LOCALSTORAGE pour récupérer les nombre d'ID de tous les articles présent, même en répétitions	CONSOLE.LOG() pour afficher les ID des articles et compter leur nombre	Ne compte pas les répétitions d'une même référence. Si la LOCALSTORAGE ne donne pas de valeurs cohérentes le calcul sera altérée. Dépend d'une requête POST et subira toutes les conséquences.
21		capterDataComande()	42 – 46	Balayer et stocker dans LOCALSTORAGE le prix total des articles choisis au moment de cliquer sur la confirmation de commande	Utilise CONSOLE.LOG() pour vérifier.	Dépend d'une requête POST et subira toutes les conséquences.
22		confirmerCommande()	50 – 58	Fonction centrale de panier pour la confirmation de commande. Au clic, elle appelle différentes fonctions dont un POST pour renvoyer à l'API l'objet contact client et le tableau PRODUCT et rediriger vers la page de confirmation	Utilise CONSOLE.LOG() pour vérifier la réponse retour du POST de l'API	Essentiellement problème au niveau de la requête POST. Mauvais format des données envoyés avec le POST
23	panier_supprimerArticle.js	supprimerArticlePanier()	6 -26	Supprimer un article. Supprime un clé et son contenu de la LOCALSTORAGE après demande de confirmation. Si LOCALSTORAGE est vide, renvoie à l'accueil	Utilise CONSOLE.LOG() pour vérifier. Afficher le nombre d'articles au différentiel	Si la LOCALSTORAGE ne donne pas de valeurs cohérentes alors la suppression sera altérée

N° Test	Fichier JS	Fonction testée	Lignes de code testées	Résultat attendu	Comment vérifier le résultat attendu	Problème possible
25	panier_updateAfficherPanier.js	updateAfficherPanier()	3 – 11	Fonction auxiliaire qui aide à actualiser le panier après modification	Toutes les modifications d'article dans le panier doivent être vues à l'affichage	Si la LOCALSTORAGE ne donne pas de valeurs cohérentes alors l'affichage sera altérée
26	panier_validerFormulaire.js	validerFormulaireCommande()	4 – 20	Vérifier que tous les champs requis du formulaire soit correctement remplis, conforme aux formats de REG-EX	Faire des test "à vide" ou oublier un ou plusieurs champs	Formulaire soit validé alors qu'il y ait une ou plusieurs anomalie dans les champs.
27	panier_viderPanier.js	viderPanier()	6 – 12	Vider le panier. Supprimer tous les éléments présents dans LOCALSTORAGE, après demande de confirmation et renvoyer à l'accueil	Utilise CONSOLE.LOG() pour vérifier que LOCALSTORAGE soit vide. Être redirigé vers l'accueil du site	LOCALSTORAGE ne se vide pas. Retour vers accueil non autorisé
28	confirmation.js	allerVersBoutique()	2 – 7	Vider la LOCALSTORAGE et redirige vers la page d'accueil au clic	Cliquer à s'assurer d'être rediriger vers la page d'accueil. CONSOLE.LOG() pour vérifier que LOCALSTORAGE soit vide	Mauvaise ou absence de redirection. LOCALSTORAGE non vidée
29		allerVersNavigation()	12 – 17	Vider la LOCALSTORAGE et redirige vers https://www.google.com	Cliquer à s'assurer d'être rediriger vers la page https://www.google.com CONSOLE.LOG() pour vérifier que LOCALSTORAGE soit vide	Mauvaise ou absence de redirection. LOCALSTORAGE non vidée
30		compteArebours	26 – 38	Décompte, affiche le temps en seconde et redirige vers l'accueil au temps = 0s	Le temps doit-être afficher et la redirection effective à l'issue	Pas d'arrêt du décompte. Pas de redirection
31		afficherDonneesComande()	42 – 47	Récupère les données de commande de LOCALSTORAGE et les affiche dans la page de confirmation	Les données de commande stockés dans LOCALSTORAGE sont récupérées.	Les données ne sont pas affichées. LOCALSTORAGE est défaillant. L'API n'a pas répondu favorablement.
32		window.addEventListener()	51 – 57	Exécute les fonctions "filles" quand le chargement du HTML est terminé	Cette fonction s'appelle à l'écoute d'un évènement. En cas de succès de toute la chaîne de fonction, nous devons voir la page s'afficher correctement.	Si le HTML ne termine pas son chargement, alors l'évènement déclencheur de ce bloc d'instructions ne démarrera pas

N° Test	Fichier JS	Fonction testée	Lignes de code testées	Résultat attendu	Comment vérifier le résultat attendu	Problème possible
1	sendXHR.js	sendXHR()	Toutes	Cette fonction doit effectuer des requêtes type GET et POST à l'API. Elle suit le principe des API fetch, et donc hérite de leurs avantages et inconvénients	Appeler la fonction avec altération d'un ou des deux premiers paramètres requis. La gestion d'erreur a été écrite dans la fonction : code réponse > 400	Code > 400. Problème de sécurité CORS si les autres fonctions génèrent des erreurs dans leur propre exécution
2	updatePanierHeader.js	updatePanierHeader()	4 – 17	Actualise automatiquement le nombre d'article et le prix total pour les pages de site web. Effectuée en haut dans l'élément HEADER	Faire varier le nombre des articles et suivre l'actualisation dans toutes les pages.	Si la LOCALSTORAGE ne donne pas de valeurs cohérentes alors filtration de redirection altérée
3		updatePanierHeader()	22 – 37	Autorise les redirections vers le panier en cliquant sur l'icône panier ou prix dans le HEADER. Condition : LOCALSTORAGE non vide	1- LOCALSTORAGE non vide, s'assurer que le clic de l'icône depuis n'importe quelle page redirige vers la page panier 2- LOCALSTORAGE vide : vérifier que l'accès au panier est bloqué partout	Si la LOCALSTORAGE ne donne pas de valeurs cohérentes alors erreur filtration de redirection altérée
4	animationImage.js	imageAnimation()	5 – 18	Au survol de la souris, les images produites de la page vitrine doivent modifier leur opacité et leur échelles	Survoler avec la souris chacune des images. Et s'assurer du fonctionnement	Cette fonction s'exécute bien sur Firefox mais pas sur Chrome. Vérifier la portabilité. Fonction non vitale dans tous les cas
5	index.js	document.AddEventListener()	32 – 40	Cette fonction doit, dès le début de la construction de la DOM, appeler plusieurs autres fonctions : sendXHR() puis en cas de succès, afficherVitrine() et imageAnimation()	Cette fonction s'appelle à l'écoute d'un événement. En de succès de toute la chaîne de fonction, nous devons voir la page s'afficher correctement les articles	Si la DOM n'initie pas son chargement, alors l'événement déclencheur de ce bloc d'instructions ne démarrera pas
		afficherVitrine()	8 – 28	Cette fonction doit afficher toutes les caméras retournées par l'API. Elle s'exécute comme un succès à la suite d'une requête GET. Elle récupère les données reçues de la requête lancée au tout début du chargement de la page HTML	Appeler la fonction en lui passant les données reçues de l'API. Elle doit afficher à l'écran	Pas de données reçues de l'API (code ≥ 400) = pas d'affichage produit. Erreur dans le HTML et CSS. Erreur dans l'élément DIV d'accueil
6	index_filtration.js	requeteFiltration()	8 – 46	Filtration des données de prix reçues par l'API. Filtrations selon PRIX et NOM	Utiliser les changements de 'FORM' pour voir les différentes filtrations. Cette fonction exécute une nouvelle GET (avec sendXHR()) à chaque changement du menu	Pas de réponse du serveur. Pas de données reçues du serveur. Tous les avantages et les inconvénients des API fetch

N° Test	Fichier JS	Fonction testée	Lignes de code testées	Résultat attendu	Comment vérifier le résultat attendu	Problème possible
7	produit.js	document. AddEventListener()	20 – 33	Récupérer par un GET ,toutes les données relatives à une produit et ensuite les transmettre à une chaîne de fonctions pour affichage et manipulations	Cette fonction s'appelle à l'écoute d'un évènement. En cas de succès de toute la chaîne de fonction, nous devons voir la page s'afficher correctement.	Si la DOM n'initie pas son chargement, alors l'évènement déclencheur de ce bloc d'instructions ne démarrera pas
8	produit_afficher.js	afficherData()	1 – 48	Construit le HTML pour afficher les données pour un produit spécifique. Les données proviennent d'un GET	Appeler la fonction en lui passant les données reçus de l'API. Elle doit afficher à l'écran	Pas de données reçues de l'API (code ≥ 400) = pas d'affichage produit. Erreur dans le HTML et CSS . Erreur dans l'élément DIV d'accueil
9	produit_ajouterPanier.js	ajouterAuPanier()	8 – 55	Crée un tableau pour chaque article choisi ainsi qu'une clé spécifique et stocke l'information dans la LOCALSTORAGE	Faire CONSOLE.LOG / CONSOLE.TABLE pour vérifier le contenu de la LOCALSTORAGE	L'article n'est pas ajouté. L'article n'est pas actualisé.
10		afficherConfirmation()	58 – 84	Affiche un message de confirmation du produit ajouté ou actualisé au clic du bouton ajouter au panier	Faire CONSOLE.LOG pour s'assurer que LOCALSTORAGE ne soit pas vide	Le message de confirmation d'ajout ne s'affiche pas
11	produit_capteurs.js	capterOptionLentilles()	1 – 7	capter les options de lentilles faites par le client et tout changement	En cliquant sur le menu déroulant le changement des options de lentilles doit être capter. Utiliser CONSOLE.LOG() pour vérifier	Le HTML sous-jacent n'a pas été construit correctement
12		capterNombreArticleChoisi()	11 -17	capter le nombre d'article(s) choisi par le client et tout changement	En cliquant sur le menu déroulant le changement de la quantité d'article(s) doit être possible. Utilise CONSOLE.LOG() pour vérifier	Le HTML sous-jacent n'a pas été construit correctement
13		capterPrixTotal()	21 – 28	capter le prix total et tout changement	Le prix total doit s'afficher dynamiquement en suivant le changement du nombre d'articles. Utilise CONSOLE.LOG() pour vérifier.	Le HTML sous-jacent n'a pas été construit correctement. Aussi problème de multiplication entre STRING et INTEGER surtout après un INNERHTML
14	produit_creationOptions.js	creationOptionLentilles()	2 – 11	Le menu déroulant doit être cliquable, afficher autoriser et les différentes options de lentilles.	Utilise CONSOLE.LOG() pour vérifier.	Menu déroulant est figé
15		creationOptionQuantite()	15 – 24	Le menu déroulant doit être cliquable, afficher autoriser et les différentes options de quantité d'article.	Utilise CONSOLE.LOG() pour vérifier.	Menu déroulant est figé

N° Test	Fichier JS	Fonction testée	Lignes de code testées	Résultat attendu	Comment vérifier le résultat attendu	Problème possible
16	panier.js	afficherPanier()	10 – 39	Balaie LOCALSTORAGE, récupère et affiche correctement tous les éléments considérés comme article choisi par un client	Les articles choisis doivent s'afficher	Si la LOCALSTORAGE ne donne pas de valeurs cohérentes alors l'affichage sera altérée
17		window.addEventListener()	44 – 52	Fonction centrale pour l'affichage du panier. Au chargement terminé, elle appelle différentes fonctions pour assurer l'affichage complet de la page	Cette fonction s'appelle à l'écoute d'un événement. En cas de succès de toute la chaîne de fonctions, nous devons voir la page s'afficher correctement.	Si la DOM ne termine pas son chargement, alors l'événement déclencheur de ce bloc d'instructions ne démarrera pas. Une ou plusieurs fonctions présente(nt) des anomalies
18	panier_ajusterQuantite.js	ajusterQuantite()	6 – 42	Pour chaque article présent dans le panier, elle permet d'ajuster leur quantité et de le supprimer. Si LOCALSTORAGE est vide, renvoie à l'accueil	Utilisez CONSOLE.LOG() pour suivre l'actualisation d'un article, et la confirmation de la suppression	Tous les articles soient modifiés ou supprimés en même temps. Aucun article n'écoute cette fonction
19	panier_confirmerCommande.js	creerObjectContact()	10 – 19	Un objet contenant les informations du client	Utilisez CONSOLE.LOG() pour vérifier la création de l'objet	Cette fonction utilisant les données du formulaire de contact, et en subira les anomalies
20		creerTableauProduct()	24 – 37	Effectue des boucles dans LOCALSTORAGE pour récupérer les nombre d'ID de tous les articles présent, même en répétitions	CONSOLE.LOG() pour afficher les ID des articles et compter leur nombre	Ne compte pas les répétitions d'une même référence. Si la LOCALSTORAGE ne donne pas de valeurs cohérentes le calcul sera altérée. Dépend d'une requête POST et subira toutes les conséquences.
21		capterDataComande()	42 – 46	Balayer et stocker dans LOCALSTORAGE le prix total des articles choisis au moment de cliquer sur la confirmation de commande	Utilise CONSOLE.LOG() pour vérifier.	Dépend d'une requête POST et subira toutes les conséquences.
22		confirmerCommande()	50 – 58	Fonction centrale de panier pour la confirmation de commande. Au clic, elle appelle différentes fonctions dont un POST pour renvoyer à l'API l'objet contact client et le tableau PRODUCT et rediriger vers la page de confirmation	Utilise CONSOLE.LOG() pour vérifier la réponse retour du POST de l'API	Essentiellement problème au niveau de la requête POST. Mauvais format des données envoyés avec le POST
23	panier_supprimerArticle.js	supprimerArticlePanier()	6 -26	Supprimer un article. Supprime un clé et son contenu de la LOCALSTORAGE après demande de confirmation. Si LOCALSTORAGE est vide, renvoie à l'accueil	Utilise CONSOLE.LOG() pour vérifier. Afficher le nombre d'articles au différentiel	Si la LOCALSTORAGE ne donne pas de valeurs cohérentes alors la suppression sera altérée

N° Test	Fichier JS	Fonction testée	Lignes de code testées	Résultat attendu	Comment vérifier le résultat attendu	Problème possible
25	panier_updateAfficherPanier.js	updateAfficherPanier()	3 – 11	Fonction auxiliaire qui aide à actualiser le panier après modification	Toutes les modifications d'article dans le panier doivent être vues à l'affichage	Si la LOCALSTORAGE ne donne pas de valeurs cohérentes alors l'affichage sera altérée
26	panier_validerFormulaire.js	validerFormulaireCommande()	4 – 20	Vérifier que tous les champs requis du formulaire soit correctement remplis, conforme aux formats de REG-EX	Faire des test "à vide" ou oublier un ou plusieurs champs	Formulaire soit validé alors qu'il y ait une ou plusieurs anomalie dans les champs.
27	panier_viderPanier.js	viderPanier()	6 – 12	Vider le panier. Supprimer tous les éléments présents dans LOCALSTORAGE, après demande de confirmation et renvoyer à l'accueil	Utilise CONSOLE.LOG() pour vérifier que LOCALSTORAGE soit vide. Être redirigé vers l'accueil du site	LOCALSTORAGE ne se vide pas. Retour vers accueil non autorisé
28	confirmation.js	allerVersBoutique()	2 – 7	Vider la LOCALSTORAGE et redirige vers la page d'accueil au clic	Cliquer à s'assurer d'être rediriger vers la page d'accueil. CONSOLE.LOG() pour vérifier que LOCALSTORAGE soit vide	Mauvaise ou absence de redirection. LOCALSTORAGE non vidée
29		allerVersNavigation()	12 – 17	Vider la LOCALSTORAGE et redirige vers https://www.google.com	Cliquer à s'assurer d'être rediriger vers la page https://www.google.com CONSOLE.LOG() pour vérifier que LOCALSTORAGE soit vide	Mauvaise ou absence de redirection. LOCALSTORAGE non vidée
30		compteArebours	26 – 38	Décompte, affiche le temps en seconde et redirige vers l'accueil au temps = 0s	Le temps doit-être afficher et la redirection effective à l'issue	Pas d'arrêt du décompte. Pas de redirection
31		afficherDonneesComande()	42 – 47	Récupère les données de commande de LOCALSTORAGE et les affiche dans la page de confirmation	Les données de commande stockés dans LOCALSTORAGE sont récupérées.	Les données ne sont pas affichées. LOCALSTORAGE est défaillant. L'API n'a pas répondu favorablement.
32		window.addEventListener()	51 – 57	Exécute les fonctions "filles" quand le chargement du HTML est terminé	Cette fonction s'appelle à l'écoute d'un évènement. En cas de succès de toute la chaîne de fonction, nous devons voir la page s'afficher correctement.	Si le HTML ne termine pas son chargement, alors l'évènement déclencheur de ce bloc d'instructions ne démarrera pas