# index.html / home.js

Test de la réponse de l’API

La fonction « displayTeddies() » sert à obtenir les informations d’un type de produit présent sur l’API, dans notre cas les oursons en peluche.

On va générer sur la page un contenant pour chaque ourson présent dans la base de données. On affichera :

* L’image de l’ourson ;
* Son nom ;
* Un lien sur tout le contenant qui permettra d’accéder à la fiche détaillée de l’ourson.

# product.html / product.js

Test de la réponse de l’API pour un ID de produit et ajout dans le localStorage des données utiles pour le panier

La fonction « init() » initie le chargement des données pour la page produit, elle inclut tout ce qui doit se passer sur la page produit.

* Dans un premier temps, la récupération des données du produit grâce à son ID stocké dans l’URL de la page, déclenché lors du clic sur la page précédente index.html.
  + Avec l’utilisation d’une boucle pour faire apparaitre le choix des différents coloris.
* Dans un second temps, le déclenchement de la fonction « tedItheCart() » au clic sur le bouton ajouter au panier. Cette fonction permet l’ajout des données voulues au localStorage : l’id, le nom, le prix, la couleur, la quantité et enfin le total (prix \* quantité). Avec une différenciation lors de l’ajout du premier item au panier.
  + Et enfin une alerte confirmant l’ajout au panier.

# cart.html / cart.js

Test de la récupération des données dans le localStorage, envoi du formulaire complété et des id produits.

La fonction « cartResume() » permet l’affichage des données dans le localStorage, le nom, le coloris, le prix et enfin la quantité. On utilise une boucle qui permet de générer une boite pour chaque item dans le panier et de faire l’addition du montant de chaque ourson dans le panier, pour afficher le total une fois la boucle terminée.

La fonction « sendingData() » est une requête de type « post » qui permet d’envoyer des données à l’API, à savoir les données dans le formulaire, nom, prénom, email, adresse et ville ainsi que les id des produits dans le panier. Avec l’envoi une fois ces étapes terminées vers une page de confirmation de commande.

# order.html / order.js

Affichage des données dans l’URL, et nettoyage du localStorage en arrivant sur la page.

La fonction « thankYou() » déclenche le nettoyage du localStorage, ce qui permet d’avoir une page panier ou il n’y a plus d’items après confirmation de la commande, et dans un second temps, d’afficher un message contenant l’ID unique de la commande et le total de la commande.