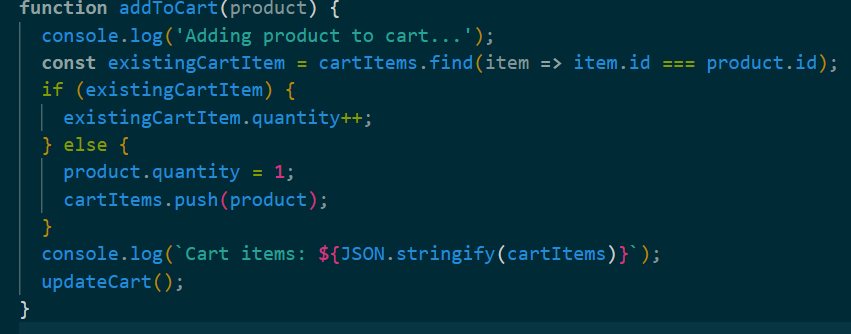
DOCUMENTATIE

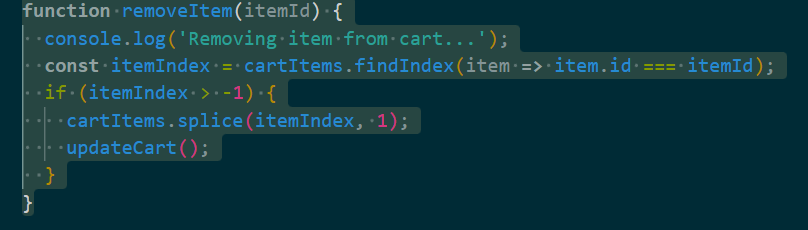
Acesta este un fragment de cod JavaScript care definește o funcție numită addToCart. Această funcție primește un singur argument, product, care reprezintă un obiect ce conține informații despre un produs pe care utilizatorul dorește să îl adauge în coșul de cumpărături.

În interiorul funcției, se verifică dacă produsul există deja în coșul de cumpărături (cartItems) prin căutarea după id. Dacă produsul există deja, cantitatea acestuia este crescută cu 1, altfel se adaugă produsul în coș cu o cantitate inițială de 1.



Acesta este un alt fragment de cod JavaScript care definește o funcție numită removeItem. Această funcție primește un singur argument, itemId, care reprezintă id-ul produsului pe care utilizatorul dorește să îl elimine din coșul de cumpărături.

În interiorul funcției, se caută produsul în coșul de cumpărături (cartItems) prin căutarea după id. Dacă produsul este găsit, se obține indexul său în array-ul cartItems prin intermediul funcției findIndex. Dacă indexul este mai mare decât -1 (adică produsul a fost găsit în array-ul cartItems), se elimină produsul din coș prin intermediul funcției splice și se actualizează vizualizarea coșului prin apelul funcției updateCart.

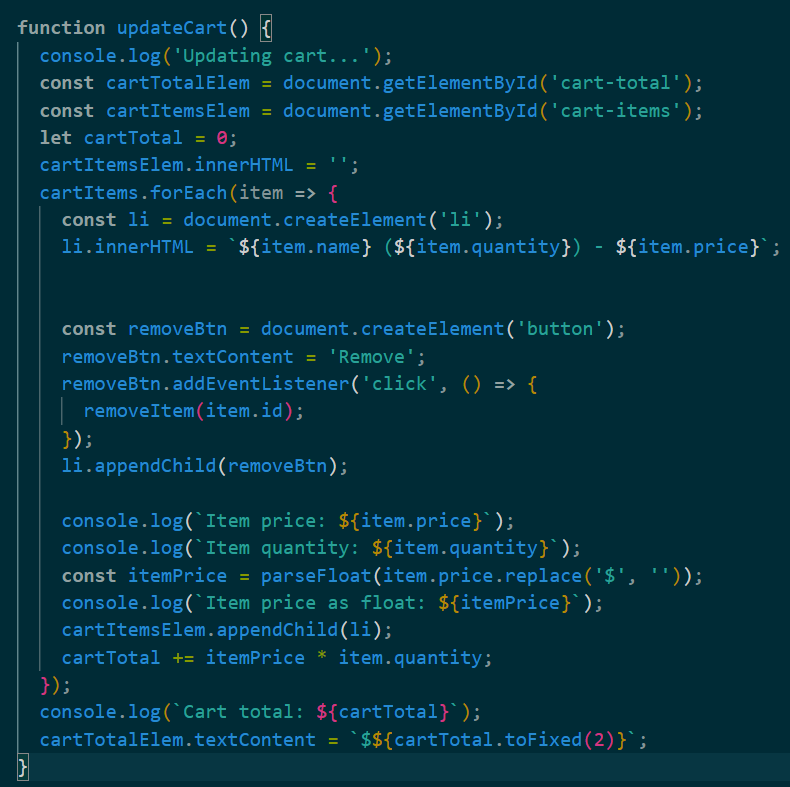


Acesta este un alt fragment de cod JavaScript care definește o funcție numită updateCart. Această funcție nu primește argumente și are rolul de a actualiza vizualizarea coșului de cumpărături pe baza informațiilor din variabila globală cartItems.

În interiorul funcției, se obțin referințe la elementele HTML care afișează numărul total de produse și detaliile despre produsele din coș. De asemenea, se inițializează o variabilă cartTotal care va fi folosită pentru a calcula prețul total al produselor din coș.

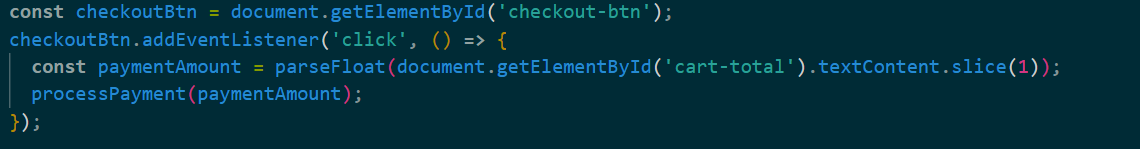
În continuare, se șterge conținutul elementului cartItemsElem (care afișează detaliile despre produse) și se parcurg toate elementele din array-ul cartItems. Pentru fiecare element, se creează un element li care va afișa detaliile produsului (nume, cantitate și preț). De asemenea, se creează un buton "Remove" care va fi asociat cu funcția removeItem și va elimina produsul respectiv din coș. Toate aceste elemente sunt adăugate la elementul cartItemsElem.

În final, se calculează prețul total al produselor din coș, care este afișat în elementul cartTotalElem.



Acest fragment de cod JavaScript selectează elementul HTML cu id-ul checkout-btn și adaugă un eveniment de click care va apela o funcție numită processPayment.

În interiorul funcției processPayment, se obține valoarea totală a coșului de cumpărături (paymentAmount) prin intermediul elementului HTML cu id-ul cart-total. Această valoare este convertită la tipul de date float și apoi este pasată ca argument către funcția processPayment



Acesta este un fragment de cod HTML care conține un formular de contact. Formularul are câmpuri pentru introducerea numelor și prenumelor, adresei de e-mail, adresei de livrare, localității, apartamentului și scării, codului poștal și numărului de telefon. De asemenea, formularul include un buton pentru trimiterea informațiilor introduse, care duce utilizatorul către o pagină denumită checkout.html.

În partea de sus a formularului, se află un element iframe care afișează o hartă Google a localizării din Brăila.

Formularul include și un element div cu id-ul error\_message, care poate fi utilizat pentru a afișa mesaje de eroare în cazul în care utilizatorii nu completează toate câmpurile obligatorii. De asemenea, formularul include și un atribut onsubmit care este setat să apeleze o funcție numită validate\_form, care este definită în altă parte.

