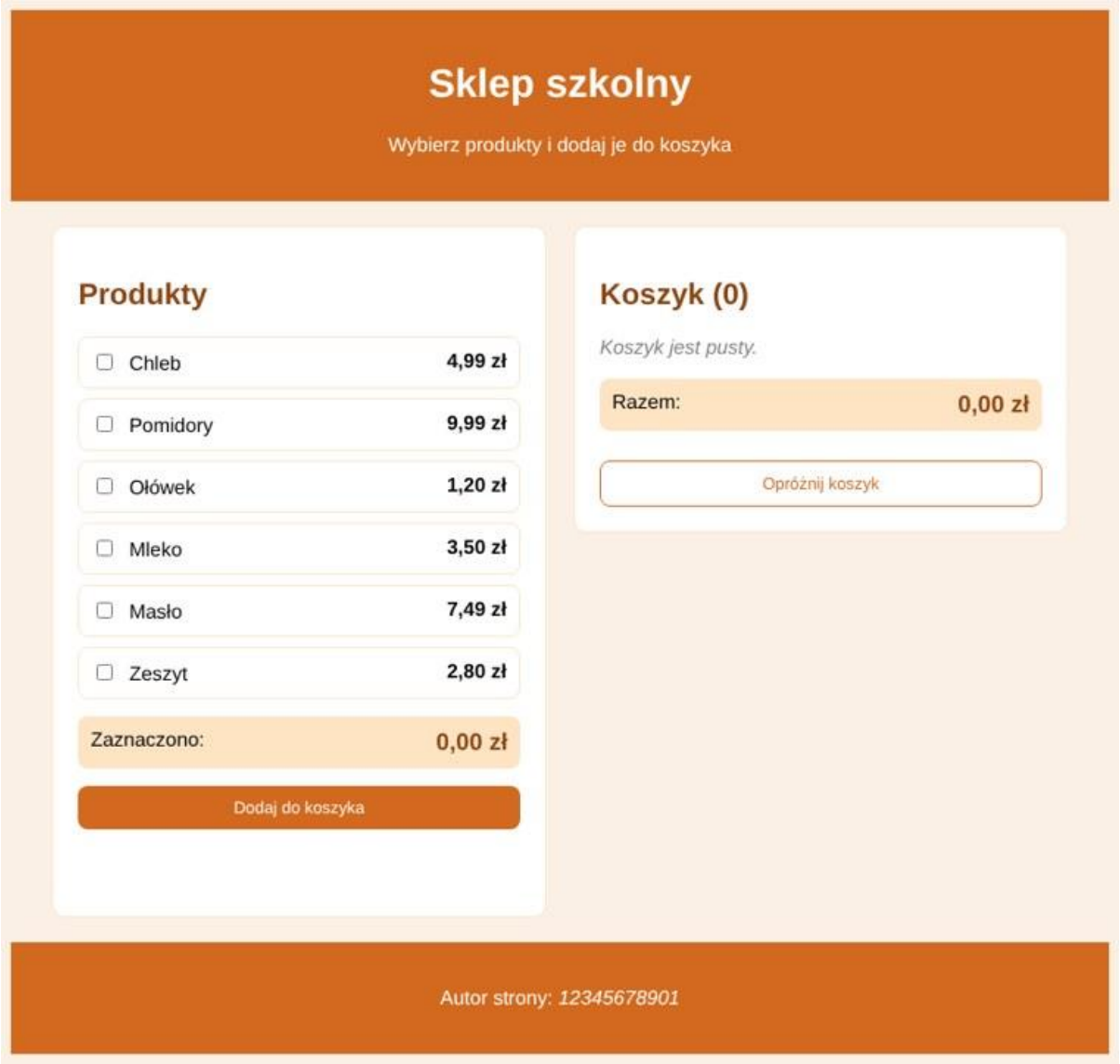
Zadanie 19: DOM – koszyk z produktami

Treść zadania

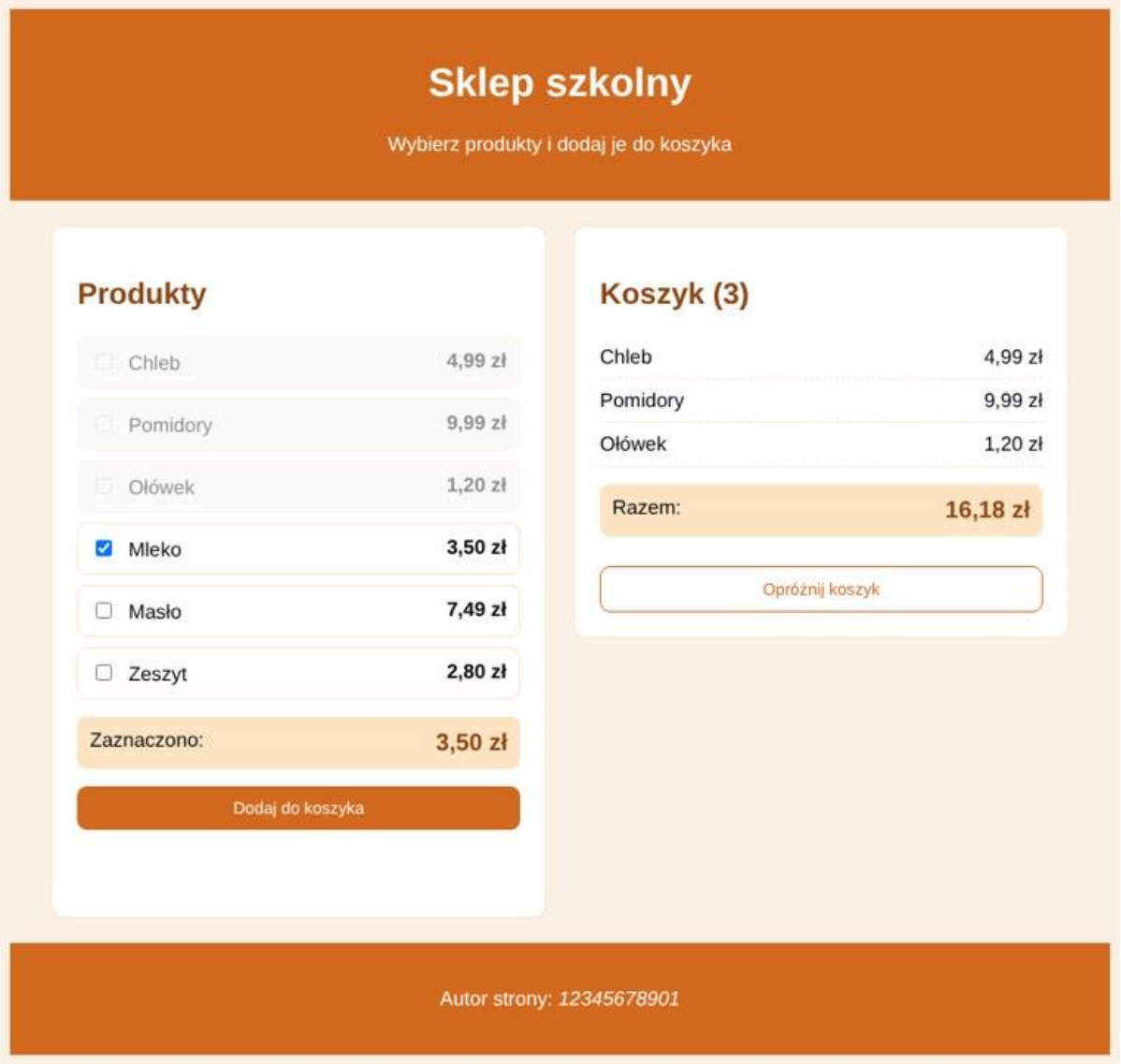
Wykonaj stronę internetową sklepu szkolnego, na której użytkownik wybiera produkty z listy i zatwierdza wybór przyciskiem, dodając je do koszyka. Na bieżąco ma być widoczna suma cen. Wykorzystaj edytor zaznaczający składnię oraz przeglądarkę internetową. Utwórz folder o nazwie zadanie19. Po skończonej pracy wszystkie pliki zapisz w tym folderze.

Witryna internetowa

Wygląd strony po otwarciu przedstawia ilustracja 1, a po dodaniu trzech produktów do koszyka i zaznaczeniu czwartego – ilustracja 2.



Ilustracja 1. Strona sklep.html po otwarciu



Ilustracja 2. Strona po dodaniu produktów do koszyka i zaznaczeniu produktu „Mleko”

Cechy strony sklep.html:

- Zapisana w języku HTML5
- Zadeklarowany polski język zawartości witryny
- Jawnie zastosowany standard kodowania polskich znaków UTF-8
- Tytuł strony „Sklep szkolny”
- Arkusz stylów w pliku o nazwie styl.css prawidłowo połączony z kodem strony
- Skrypt w pliku o nazwie skrypt.js, połączony ze stroną znacznikiem `<script>` umieszczonym na końcu bloku `<body>`
- Podział strony na bloki: nagłówkowy, główny i stopkę, za pomocą znaczników `<header>`, `<main>` i `<footer>`. Blok główny zawiera dwa bloki sekcji umieszczone obok siebie
- **Zawartość bloku nagłówkowego:** nagłówek pierwszego stopnia o treści „Sklep szkolny” oraz paragraf o treści „Wybierz produkty i dodaj je do koszyka”
- **Zawartość pierwszej sekcji:** nagłówek drugiego stopnia „Produkty”; pusty blok `<div>` o identyfikatorze `lista-produktow` (produkty wstawia skrypt); blok `<div>` formatowany klasą `podsumowanie`, zawierający znacznik `<span>` o treści „Zaznaczono:” oraz znacznik

 o identyfikatorze suma-zaznaczonych i treści „0,00 zł”; przycisk (id="dodaj") o treści „Dodaj do koszyka”; pusty paragraf formatowany klasą komunikat o identyfikatorze komunikat

- **Zawartość drugiej sekcji:** nagłówek drugiego stopnia o treści „Koszyk (0)”, w którym cyfra 0 jest zapisana w znaczniku o identyfikatorze liczba-w-koszyku; paragraf formatowany klasą pusty o identyfikatorze pusty-koszyk i treści „Koszyk jest pusty.”; pusta lista nieuporządkowana <ul> o identyfikatorze koszyk; blok formatowany klasą podsumowanie, zawierający „Razem:” oraz o identyfikatorze suma-koszyka i treści „0,00 zł”; przycisk formatowany klasą wtorny (id="oproznij") o treści „Opróżnij koszyk”
- **Zawartość bloku stopki:** paragraf o treści „Autor strony: ”, dalej wstawiony numer zdającego, zapisany za pomocą znacznika semantycznego oznaczającego tekst uwypuklony, formatowany domyślnie jako pochylony

Styl CSS witryny internetowej

Styl CSS zdefiniowany jest w całości w zewnętrznym pliku o nazwie styl.css. Cechy formatowania CSS działające na stronie:

- Domyślnie, dla wszystkich selektorów: krój czcionki „Verdana”, w przypadku braku sans-serif
- Dla ciała strony: kolor tła Linen
- Dla bloku nagłówkowego i stopki: kolor tła Chocolate, kolor czcionki White, wyrównanie tekstu do środka, marginesy wewnętrzne 20 px
- Dla bloku głównego: wyrównanie tekstu do środka
- Dla bloków sekcji: model pudełkowy border-box, wyświetlanie jako element liniowo-blokowy (inline-block), wyrównanie do góry (vertical-align: top), szerokość 45%, marginesy zewnętrzne górny i dolny 20 px, lewy i prawy 1%, marginesy wewnętrzne 20 px, wyrównanie tekstu do lewej strony, kolor tła White, obramowanie linią ciągłą o grubości 1 px i kolorze Bisque, zaokrąglenie rogów 10 px
- Dla nagłówka drugiego stopnia: kolor czcionki SaddleBrown
- Dla klasy produkt: wyświetlanie blokowe, marginesy zewnętrzne dolne 8 px, marginesy wewnętrzne 10 px, obramowanie linią ciągłą o grubości 1 px i kolorze Bisque, zaokrąglenie rogów 8 px, kursor pointer
- Gdy kursor myszy znajdzie się na elemencie klasy produkt, kolor tła zmienia się na Linen
- Dla klasy nazwa: margines zewnętrzny lewy 10 px
- Dla klasy cena: położenie po prawej stronie (float: right), czcionka pogrubiona
- Dla klasy w-koszyku: przezroczystość 0.45, kursor not-allowed, kolor tła WhiteSmoke
- Dla klasy podsumowanie: ukrycie nadmiaru (overflow: hidden), marginesy zewnętrzne górny i dolny 14 px, lewy i prawy 0, marginesy wewnętrzne 10 px, kolor tła Bisque, zaokrąglenie rogów 8 px
- Dla znaczników w elemencie klasy podsumowanie (selektor potomka): położenie po prawej stronie (float: right), rozmiar czcionki 120%, kolor czcionki SaddleBrown
- Dla elementu o identyfikatorze koszyk: marginesy zewnętrzne i wewnętrzne 0, brak znaczników listy (list-style-type: none)
- Dla znaczników <li> w elemencie o identyfikatorze koszyk: ukrycie nadmiaru (overflow: hidden), marginesy wewnętrzne górny i dolny 8 px, lewy i prawy 0, obramowanie dolne linią kreskowaną o grubości 1 px i kolorze Bisque

- Dla ostatniego znacznika `<span>` w każdym `<li>` w elemencie o identyfikatorze koszyk (pseudoklasa `:last-child`): położenie po prawej stronie (`float: right`)
- Dla klasy pusty: kolor czcionki Gray, czcionka pochylona
- Dla klasy komunikat: minimalna wysokość 1.5em, kolor czcionki FireBrick, rozmiar czcionki 90%
- Dla selektora przycisku: szerokość 100%, marginesy wewnętrzne 10 px, kolor tła Chocolate, kolor czcionki White, brak obramowania, zaokrąglenie rogów 8 px, kursor pointer
- Gdy kursor myszy znajdzie się na przycisku, kolor tła zmienia się na SaddleBrown
- Dla klasy wtorny: margines zewnętrzny górny 10 px, kolor tła White, kolor czcionki Chocolate, obramowanie linią ciągłą o grubości 1 px i kolorze Chocolate. Gdy kursor myszy znajdzie się na elemencie klasy wtorny, kolor tła zmienia się na Bisque

Uwaga: nie należy zmieniać nazw klas ani identyfikatorów, gdyż jest to uwarunkowane projektem całej witryny. Klasy produkt, nazwa, cena i w-koszyku nie występują w kodzie HTML. Nadaje je skrypt.

Skrypt działający po stronie klienta

Wymagania dotyczące skryptu:

- Zapisany w języku JavaScript w pliku skrypt.js
- Należy stosować znaczące nazewnictwo zmiennych i funkcji w języku polskim lub angielskim
- Wszystkie polecenia są wykonywane w odpowiedzi na działanie użytkownika (zaznaczenie produktu, kliknięcie przycisku). Po otwarciu strony koszyk jest pusty
- Sumy cen są wyświetlane w formacie „4,99 zł” (dwa miejsca po przecinku, przecinek jako separator, jednostka „zł” po spacji)

Dane produktów znajdują się w tablicy obiektów, tak jak w poprzednim zadaniu. Przepisz ją na początek skryptu:

```
const products = [
    { name: "Chleb", price: 4.99 },
    { name: "Pomidory", price: 9.99 },
    { name: "Ołówek", price: 1.2 },
    { name: "Mleko", price: 3.5 },
    { name: "Masło", price: 7.49 },
    { name: "Zeszyt", price: 2.8 },
];
```

Korzystając z poznanych metod manipulacji DOM JavaScript, wykonaj poniższe polecenia.

1. Wygeneruj w elemencie `#lista-produktow` listę produktów na podstawie tablicy `products`. Każdy produkt ma być elementem `<label>` z klasą „produkt”, zawierającym pole wyboru `checkbox`, `<span>` z klasą „nazwa” (nazwa produktu) i `<span>` z klasą „cena” (cena w formacie `4,99 zł`).

2. Pokazuj **na bieżąco** sumę cen zaznaczonych produktów w elemencie `#suma-zaznaczonych`. Suma ma się zmieniać przy każdym zaznaczeniu i odznaczeniu produktu.

3. Po kliknięciu „Dodaj do koszyka” dodaj zaznaczone produkty do listy #koszyk (jeden `<li>` na produkt zawierający dwa elementy `<span>`: nazwę i cenę). Jeśli nic nie zaznaczono, wyświetl w elemencie `#komunikat` tekst „Zaznacz co najmniej jeden produkt.”.

4. Produkty dodane do koszyka odznacz, zablokuj ich pole wyboru (`disabled`) i nadaj etykiecie klasę „w-koszyku”, aby nie dało się ich dodać drugi raz. Wyzeruj sumę zaznaczonych.

5. Po każdej zmianie koszyka zaktualizuj liczbę produktów w nagłówku koszyka (`#liczba-w-koszyku`) oraz sumę w `#suma-koszyka`. Gdy koszyk jest pusty, pokaż napis „Koszyk jest pusty.”, w przeciwnym razie ukryj go.

6. Przycisk „Opróżnij koszyk” ma usuwać wszystkie produkty z koszyka, odblokowywać pola wyboru i zdejmować klasę „w-koszyku”.

Tabela 1. Wybrane pola i metody modelu DOM języka JavaScript

Wyszukiwanie i zmiana elementów	Dodatkowo dopuszczalne
`document.getElementById(id)`	`document.createElement(element), append()`
`document.querySelectorAll(CSSselector)`	`element.checked, element.disabled`
`element.innerHTML = "wartość"`	`element.textContent = "wartość"`
`element.style.property_name = "wartość"`	`element.classList.add(klasa), remove(klasa)`
`onclick, onchange`	`element.addEventListener(zdarzenie, funkcja)`
`toFixed(2), replace(a, b)`	`tablica.reduce(), map(), forEach()`

UWAGA: po zakończeniu pracy sprawdź działanie strony w przeglądarce. W folderze powinny znajdować się pliki: sklep.html, styl.css, skrypt.js.
Ocenie podlegać będą 3 rezultaty: zawartość witryny internetowej, styl CSS witryny internetowej, skrypt.

Rozwiązanie

Pytania teoretyczne

Pytanie 1. Na czym polega delegacja zdarzeń i dlaczego tutaj ma sens? Co się stanie z nasłuchami, gdy produkty będą dodawane do strony dynamicznie?

Pytanie 2. Czym różni się zdarzenie `change` od zdarzenia `click` w polu wyboru `checkbox`? Które lepiej nadaje się do tego zadania?

Pytanie 3. Co zwróci wyrażenie `document.querySelectorAll('input:checked')`?

- **A.** tablicę (`Array`) zaznaczonych pól
- **B.** żywą kolekcję `HTMLCollection` zaznaczonych pól
- **C.** statyczną listę `NodeList` zaznaczonych pól
- **D.** pierwsze zaznaczone pole

Pytanie 4. Dlaczego użyto `textContent`, a nie `innerHTML`, do wstawiania nazw produktów? Co mogłoby się stać, gdyby nazwy pochodziły z pola tekstowego wypełnianego przez użytkownika?

Pytanie 5. Dlaczego stan koszyka trzymamy w tablicy `koszyk`, a nie odczytujemy go z elementów strony? Jaka jest zaleta funkcji `odswiezKoszyk()`?

Pytanie 6. Pole wyboru ma atrybut `disabled`. Czy zdarzenia kliknięcia w takie pole zostaną wywołane? Czy zablokowane pole jest wysyłane w formularzu? Jak zmieni się wynik zapytania `input:checked` po zablokowaniu zaznaczonego pola?

Odpowiedzi do pytań teoretycznych