

Лабораторна робота №6

Робота з масивами, об'єктами та циклами у JavaScript.

Створення фіскального касового чеку

Мета:

Поглибити розуміння змінних, області видимості, типів даних, операторів та умовних операторів, а також навчитися працювати з масивами та об'єктами, використовувати цикли і функцію `prompt` для введення даних користувачем.

Завдання:

Розробити програму для введення даних про товари, обчислення їх вартості з податком та без податку, а також виведення фіскального касового чеку.

Функціональні вимоги

1. Введення даних про товари:

- Програма повинна запитувати у користувача назву товару.
- Програма повинна запитувати у користувача ціну товару. Ціна може бути введена як з крапкою, так і з комою як роздільником дробової частини.
- Програма повинна запитувати у користувача кількість товарів.

2. Перевірка введених даних:

- Програма повинна перевіряти, щоб ціна товару була більше 0 і була числовим значенням.
- Програма повинна перевіряти, щоб кількість товарів була більше 0 і була числовим значенням.
- Якщо введені дані некоректні, програма повинна припиняти подальшу обробку та виводити повідомлення про помилку.

3. Зберігання даних:

- Програма повинна зберігати введені дані про товари у масиві об'єктів. Кожен об'єкт повинен містити назву товару, ціну товару та кількість товарів.

4. Обчислення вартості:

- Програма повинна обчислювати загальну вартість товарів без ПДВ.
- Програма повинна обчислювати суму ПДВ. Податкова ставка 20%.
- Програма повинна обчислювати загальну вартість товарів з ПДВ.

5. Виведення фіскального касового чеку:

- Програма повинна виводити фіскальний касовий чек, який містить:
 - Назву товару
 - Ціну товару з податком
 - Кількість одиниць товару
 - Сумарну вартість товару з податком
 - Сумарну вартість товару без податку
- Програма повинна виводити загальну вартість товарів без податку.
- Програма повинна виводити суму податку.
- Програма повинна виводити загальну вартість товарів з податком.
- Програма повинна виводити інформацію про касу та касира.
- Програма повинна виводити поточну дату та час.
- Програма повинна виводити QR-код чеку (просто текст).

Нефункціональні вимоги

1. Інтерфейс користувача:

- Введення даних повинно здійснюватися через діалогові вікна (prompt).
- Виведення результатів повинно здійснюватися через консоль (console.log).
- Запитувати користувача чи він бажає додати ще товар за допомогою функції confirm.

2. Обробка чисел:

- Програма повинна коректно обробляти введення чисел з комою як роздільником дробової частини, замінюючи кому на крапку перед перетворенням на число за допомогою **parseFloat**.

3. Форматування чисел:

- Програма повинна формувати числа з плаваючою комою до двох десяткових знаків під час виведення.

Інші вимоги

1. Програма повинна коректно виконувати усі розрахунки.
2. Необхідно використовувати цикли (**while**, **for**).
3. Необхідно створити хоча б одну функцію і використати її.
4. Використовувати відповідні оголошення змінних (**let**, **const**).
5. Перевіряти введені дані та перетворювати у відповідний формат за потреби.

Додаткові вимоги (не обов'язкові)

1. Додати у функціонал діалогове вікно зі загальною вартістю з ПДВ після завершення додавання товарів. У полі вводу необхідно ввести суму, яка сплачується. Якщо сума платежу, наприклад, 360 грн, а клієнт дає касиру 400 грн, то у чеку має бути інформація про суму готівкою і решту.
2. Перевірка на натискання кнопки "Cancel" (скасувати) у діалоговому вікні.
3. Якщо не введено жодного товару, то необхідно вивести текст "Жодного товару не було додано. Чек не може бути сформований."
4. Програма повинна перевіряти тип введених даних і повертати помилку на невірний тип даних.
5. Зробити послідовну нумерацію доданих товарів.

Приклад роботи програми

1. Користувач вводить назву товару: "Хліб".
2. Користувач вводить ціну товару з податком: "20,5".
3. Користувач вводить кількість товарів: "2".
4. Користувач підтверджує бажання додати ще один товар.
5. Користувач вводить назву товару: "Молоко".
6. Користувач вводить ціну товару з податком: "15.3".
7. Користувач вводить кількість товарів: "3".
8. Користувач відмовляється додавати ще один товар.
9. Програма виводить фіскальний касовий чек з усіма обчисленнями та інформацією.

Що таке ПДВ?

ПДВ – це податок, який додається до ціни товару чи послуги. Наприклад, якщо книга коштує 100 грн, а ставка ПДВ – 20%, то Ви заплатите 120 грн. Ці додаткові 20 грн – це і є ПДВ.

Якщо потрібно розрахувати вартість товару без ПДВ (відома ціна з ПДВ та ставка ПДВ), ділимо ціну з ПДВ на (1 + ставка ПДВ у десятковому вираженні).

Наприклад, якщо маємо ціну з ПДВ (120 грн) і ставку ПДВ (20%), тоді сума без ПДВ = $120 \text{ грн} / (1 + 0,2) = 120 \text{ грн} / 1,2 = 100 \text{ грн}$.

Фіскальний касовий чек

Назва товару	Ціна з ПДВ	Кількість	Вартість	Вартість без ПДВ
Хліб	20.00	1	20.00	16.67
Молоко	35.00	2	70.00	58.33
Яйце	4.50	6	27.00	22.50

Загальна вартість без ПДВ: 97.50 грн

Сума ПДВ: 19.50 грн

Загальна вартість з ПДВ: 117.00 грн

Кількість артикулів: 3

Сума готівкою: 200.00 грн

Решта: 83.00 грн

Каса: 0001

Касир: <тут Ваше ПІБ>

Дата: 09/10/2024

Час: 23:03:42

QR-код чеку: [QR-код]

Структура архіву:

```
lab6/
├── index.html
└── script.js
```

Матеріали та довідники:

- ✓ [w3schools.com](https://www.w3schools.com) – JavaScript (англійською мовою)
- ✓ w3schoolsua.github.io – JavaScript (українською мовою)
- ✓ [MDN Web Docs – JavaScript](https://developer.mozilla.org/en-US/docs/Web/JavaScript) (англійською мовою)

Рекомендовані редактори коду:

- ✓ VSCode
- ✓ PSPad
- ✓ Notepad++
- ✓ Sublime Text (платний)
- ✓ WebStorm, PHPStorm, PyCharm (платний, [безкоштовна ліцензія для навчання](#))

Вимоги до звіту:

Після виконання завдання, студент повинен надати звіт, який включає:

1. Короткий опис виконаної роботи:

- Опис мети лабораторної роботи.
- Короткий опис завдання та його виконання.

2. Скріншоти коду з коментарями та результатів виконання завдання.

3. Висновок:

- Короткий підсумок виконаної роботи.
- Висновки щодо досягнення мети лабораторної роботи.
- Відповіді на питання, які могли виникнути під час виконання роботи.

4. Архів з файлами лабораторної роботи:

- Повинен містити HTML, JS файли.
- Архівуйте файли у форматі ZIP або RAR.

Контрольні запитання для самоконтролю:

Загальні питання:

1. Як вивести повідомлення у консоль браузера?
2. Як вивести повідомлення у вікно **alert**?
3. Як запитати у користувача введення даних через **prompt**?
4. Як перетворити рядок на число у JavaScript?
5. Що таке область видимості змінних? Яка різниця між глобальною та локальною областю видимості?
6. Що таке функція у JavaScript? Як оголосити та викликати функцію?
7. Як працює оператор **typeof** і для чого він використовується?
8. Як за допомогою JavaScript замінити одні символи на інші?
9. Що таке **parseFloat** та **parseInt**?
10. Що таке **toString** та **toFixed**?
11. Що таке **isNaN** та **isInteger**?
12. Як перевірити, чи введене число більше 0?
13. Як перевірити, чи введене значення не є **null**?
14. Як виводити поточну дату та час у консоль?
16. Як використовувати методи **toLocaleDateString** та **toLocaleTimeString** для форматування дати та часу?

Змінні та типи даних:

1. Що таке змінні у JavaScript і як вони оголошуються?
2. Поясніть різницю між **let**, **const** та **var**
3. Як присвоїти значення змінній?
4. Які типи даних існують у JavaScript?
5. Наведіть приклади використання різних типів даних
6. Як можна дізнатися тип даних змінної?

Оператори:

1. Що таке оператор у JavaScript?
2. Які арифметичні оператори підтримуються у JavaScript?
3. Що таке оператор присвоєння? Які є види операторів присвоєння?
4. Які логічні оператори підтримуються у JavaScript? Наведіть приклади.
5. Що таке оператор порівняння? Які є види операторів порівняння?
6. Як працює оператор залишку від ділення (%)?
7. Що таке унарний оператор? Наведіть приклади.
8. Що таке тернарний оператор? Як він використовується?
9. Що таке умовний оператор? Наведіть приклади.

Умовні оператори:

1. Що таке умовний оператор у JavaScript?
2. Як працює оператор **if**?
3. Як працює оператор **if...else**?
4. Як працює оператор **if...else if...else**?
5. Що таке оператор **switch**? Як він використовується?
6. Яка різниця між операторами **==** і **===**?
7. Як використовувати логічні оператори **&&** (AND) та **||** (OR) у умовних виразах?
8. Що таке оператор **!** (NOT) і як він використовується?

Цикли:

1. Як працюють цикли у JavaScript?
2. Які цикли є у JavaScript?
3. Поясніть різницю між циклами **for**, **while** та **do...while**.
4. Наведіть приклади використання кожного з цих циклів.
5. Як працює цикл **for**?
6. Як працює цикл **while**?
7. Як працює цикл **do...while**? Як він відрізняється від циклу **while**?
8. Як працює цикл **for...in**? Як він використовується для ітерації по властивостях об'єкта?
9. Як працює цикл **for...of**? Як він використовується для ітерації по ітерабельних об'єктах (масивах, рядках, картах тощо)?
10. Як використовувати оператори **continue** та **break** у циклах?
11. Як уникнути нескінченних циклів?
12. Поясніть, як методи масивів, такі як **forEach**, **map**, **filter**, **reduce**, можуть бути використані для ітерації по масиву.

Масиви:

1. Що таке масиви у JavaScript і як з ними працювати?
2. Як додати елемент до масиву?
3. Як видалити елемент з масиву?
4. Як отримати доступ до елементів масиву?
5. Як перебирати елементи масиву?
6. Як порахувати довжину масиву?
7. Які методи можна використовувати для сортування масиву?
8. Як об'єднати два масиви?

Об'єкти:

1. Що таке об'єкти у JavaScript і як з ними працювати?
2. Як створити об'єкт?
3. Як додати властивість до об'єкта?

4. Як видалити властивість з об'єкта?
5. Як отримати доступ до властивостей об'єкта?
6. Як перебирати властивості об'єкта?
7. Як перевірити наявність властивості в об'єкті?
8. Що таке методи об'єкта? Як додати метод до об'єкта? Як викликати метод об'єкта?