***Завдання до уроку №7. String***

***Варіант 1***

**Завдання 1**

Створіть програму, яка отримує від користувача послідовність чисел, розділених комами. Перевірте, чи введена послідовність складається лише з чисел. Виведіть на екран лише числа, які є парними. Для цього використовуйте методи split(), equals(), String.valueOf(), if, та цикли.

**Завдання 2**

Ви отримали таємне повідомлення від свого друга, але воно написане в закодованій формі. Вам необхідно розкодувати це повідомлення, використовуючи методи роботи з рядками. Вам відомо, що:

* Кожна літера повідомлення була змінена на її велику форму, наприклад, 'a' стало 'A', 'b' стало 'B' і так далі.
* Всі пробіли були замінені на знак плюса '+'.
* Кожна літера повідомлення була замінена на номер її позиції в латинському алфавіті (нумерація починається з 1). Наприклад, 'A' відповідає 1, 'B' відповідає 2 і так далі.

Отже, вам потрібно створити програму, яка розкодує це повідомлення і виводить його в розшифрованому вигляді.

Приклад:

Вхідне повідомлення: "1+2+3+4+5"

Розкодоване повідомлення: "ABCDE"

Завдання вимагає використання методів indexOf, substring, toLowerCase і toUpperCase для роботи з рядками та обробки символів.

**Завдання 3. Повторення**

Створіть консольну програму для управління списком контактів. Кожен контакт повинен мати ім'я та номер телефону.

Створіть масив, який буде зберігати контакти. Почніть з пустого списку.

Реалізуйте основне меню для користувача, де буде доступні такі опції:

* Перегляд списку контактів.
* Додавання нового контакту.
* Видалення контакту за іменем.
* Пошук контакту за іменем.
* Вихід з програми.

Після вибору опції "Додавання нового контакту", користувач має ввести ім'я та номер телефону контакту, який потім додається до списку контактів.

Після вибору опції "Видалення контакту за іменем", користувач вводить ім'я контакту, який повинен бути видалений зі списку.

***Варіант 2***

**Завдання 1**

Розробіть текстовий калькулятор, який приймає математичний вираз у вигляді рядка, розділеного операціями додавання, віднімання, множення та ділення, і обчислює результат. Використовуйте split(), equals(), String.valueOf(), if, цикли та інші необхідні операції для обробки виразу та розрахунку результату.

**Завдання 2**

Напишіть програму, яка аналізує введений користувачем текст для пошуку ключових слів на певну тему. Користувач буде вводити текст, і програма повинна знаходити всі ключові слова в цьому тексті та виводити їх кількість. Тему та ключові слова визначіть самостійно.

Під час розв'язання цього завдання ви можете використовувати методи indexOf, lastIndexOf, substring, toLowerCase або toUpperCase, щоб зробити пошук ключових слів нечутливим до регістру та зручним для користувача.

Наприклад, якщо тема - "природа", а ключові слова - "ліс", "річка", "гори", програма повинна знайти та підрахувати ці слова в тексті користувача незалежно від їх регістру (великі чи малі літери).

**Завдання 3. Повторення**

Розробіть консольну програму для управління книжковою колекцією. Кожна книга повинна мати такі характеристики:

* Назва книги (рядок)
* Автор книги (рядок)

Ваша програма повинна підтримувати наступні операції:

* Додавання книги: Користувач має мати можливість додавати нові книги до колекції. Потрібно використовувати масив для збереження книг.
* Видалення книги: Користувач має мати можливість видаляти книги з колекції за допомогою назви книги або імені автора. При видаленні книги з масиву, інші книги мають зсунутися, щоб не залишати порожніх слотів у масиві.
* Перегляд колекції: Користувач має мати можливість переглядати всі книги в колекції.

***Варіант 3***

**Завдання 1**

Шифрування рядка за допомогою простого алгоритму. Задача полягає в тому, щоб створити програму, яка шифрує введений рядок тексту за допомогою простого алгоритму. Ось як це працює: Користувач вводить рядок тексту і цільове числове значення, яке використовуватиметься для шифрування. Програма переводить це числове значення в рядок, використовуючи метод String.valueOf(). Рядок тексту розбивається на окремі слова, використовуючи метод split(). Для кожного слова програма використовує метод equals() або equalsIgnoreCase(), щоб порівняти його з числовим рядком, який ми отримали раніше. Якщо слово рівне числовому рядку, то воно замінюється на зашифрований символьний рядок (наприклад, "\*" або "#"). Зашифрований рядок повертається користувачу.

**Завдання 2**

Створіть програму, яка аналізує текстовий коментар користувача і реагує на наявність певних слів або фраз. Програма має використовувати методи indexOf, substring, toLowerCase або toUpperCase.

Кроки:

* Спрошуйте користувача про його коментар.
* Перетворіть введений коментар на нижній регістр, щоб забезпечити нечутливість до регістру.
* Використовуючи метод indexOf та substring, знайдіть позиції та витягніть підстроки коментаря, що містять певні ключові слова або фрази. Наприклад, можливі ключові слова: "привіт", "котик", "піцца".
* Використовуйте if або switch для реагування на знайдені ключові слова або фрази та виведення відповідних повідомлень. Наприклад, якщо в коментарі знайдено "привіт", виведіть "Привіт! Як я можу вам допомогти?".

Перевірте, чи є в коментарі інші ключові слова або фрази та виведіть відповідні повідомлення.

**Завдання 3. Повторення**

Міні Інформаційна Система для Клубу Футбольних Уболівальників

Розробіть міні інформаційну систему для клубу футбольних уболівальників, де вони можуть вести записи про свої улюблені футбольні команди та матчі. Ваша програма повинна включати такі функції:

Додавання команд та їх характеристик:

Користувач повинен мати можливість додати нову футбольну команду, включаючи її назву, країну. Всі дані про команди повинні зберігатися в масиві.

Додавання результатів матчів:

Користувач повинен мати можливість додати результати останнього матчу для обраної команди. Він повинен ввести ім'я команди, її оцінку та оцінку суперника.

Результати матчів повинні також зберігатися в масиві.

Видалення команди:

Користувач повинен мати можливість видалити команду зі списку футбольних команд. Після видалення всі результати матчів з цією командою також повинні бути видалені.

Виведення інформації:

Користувач повинен мати можливість переглянути список всіх зареєстрованих команд та їх характеристики.

Користувач повинен також мати можливість переглядати результати останнього матчу для обраної команди.

Меню інтерфейсу:

Створіть текстовий консольний інтерфейс, де користувач може вибирати різні опції за допомогою меню.