

Лабораторна робота №1 (2 год. Максимальна оцінка 3 бали)

Тема: Операції над стрічками символів.

Мета: Засвоєння основних операцій над стрічками: пошук, заміна, отримання під рядка.

ТЕОРЕТИЧНІ ВІДОМОСТІ

Рядки

У JavaScript текстові дані зберігаються у вигляді рядків. Не існує окремого типу для одного символу.

Внутрішній формат для рядків завжди UTF-16, він не прив'язаний до кодування сторінки.

Лапки

Рядки можуть бути включені в одинарні лапки, подвійні лапки або зворотні знаки.

```
let single = 'одинарні-лапки';  
let double = "подвійні-лапки";  
let backticks = `зворотні-лапки`;
```

Одинарні та подвійні лапки по суті однакові. Однак зворотні лапки дозволяють нам вставляти будь-який вираз у рядок, загортаючи його у `${...}`:

```
function sum(a, b) {  
    return a + b;  
}  
  
alert(`1 + 2 = ${sum(1, 2)}.`); // 1 + 2 = 3.
```

Зміна регістру

Методи `toLowerCase()` та `toUpperCase()` змінюють регістр символів.

Пошук підрядка

Існує декілька способів для пошуку підрядка.

```
str.indexOf(substr, pos)
```

Сучасніший метод

```
str.includes(substr, pos)
```

повертає true/false в залежності від того чи є substr в рядку str.

Він шукає підрядок substr в рядку str, починаючи з позиції pos, і повертає позицію, де знаходиться збіг, або -1 якщо збігів не було знайдено.

Отримання підрядка

В JavaScript є 3 метода для отримання підрядка: substring, substr та slice.

```
str.slice(start [, end])
```

Повертає частину рядка починаючи від start до (але не включно) end.

```
str.substring(start [, end])
```

Повертає частину рядка між start та end (не включаючи end)...

Цей метод майже такий самий що і slice, але він дозволяє задати start більше ніж end (у цьому випадку він просто міняє значення start і end місцями).

```
str.substr(start [, length])
```

Повертає частину рядка з позиції start, із заданою довжиною length.

На відміну від попередніх методів, цей дозволяє вказати довжину length замість кінцевої позиції.

З додатковими методами можна ознайомитися за вказівкою

Методи https://www.w3schools.com/js/js_string_methods.asp

Пошук в рядку https://www.w3schools.com/js/js_string_search.asp

ЗАВДАННЯ ДЛЯ САМОСТІЙНОГО ВИКОНАННЯ.

Завдання 1. Перший великий.

Напишіть функцію ucFirst(str), яка повертає рядок str з першим символом у верхньому регістрі, наприклад:

```
ucFirst("василь") == "Василь";
```

Завдання 2. Перевірки на паразити.

Напишіть функцію checkSpam(str), яка повертає true, якщо str рядок містить слова паразити: «типу», «якби б це», «коротше», «таки», «ну», «ось», «взагалі-то», «значить», інакше false.

Функція має бути нечутливою до регістру.

Завдання 3. Три крапки.

Створіть функцію *truncate(str, maxlen)*, яка перевіряє довжину *str* і, якщо вона перевищує *maxlen* – замінює кінець *str* символом трьох крапок "...", щоб його довжина була рівною *maxlen*.

Результатом функції повинен бути урізаний (якщо потребується) рядок.

Завдання 4. Зрізи.

Дано рядок `let str = "JavaScript is awesome!";`

Отримати:

- Отримати перші 5 символів рядка
- Видалити перший і останній символ рядка
- Отримати останні 4 символи рядка
- Розділити рядок на дві частини: до першого пробілу та після
- Отримати все, крім перших 4 символів

Завдання 5. Анаграми.

Анаграмами називаються слова, утворені шляхом взаємної перестановки букв. В англійській мові, наприклад, анаграмами є слова «live» і «evil», а українською «автор» - «товар». Напишіть програму, яка буде запитувати у користувача два слова, визначати, чи є вони анаграмами, і виводити на екран відповідь.

Завдання 6. Зняти дублювання.

Вводиться рядок. Потрібно видалити з нього повторювані символи і всі пропуски.

Вхід:

aa

a a b b c dd e

Вивід:

a

abcde