

Лабораторна робота №1

Тема:

Основи JavaScript. Функції та Структури даних.

Коротка довідка

JavaScript (JS) — це невибаглива до ресурсів скриптова мова програмування код якої інтерпретується та компілюється під час виконання. JS було розроблено у Netscape як універсальний інструмент зв'язку різних частин веб сайтів. Сьогодні використання як скриптової мови для веб-сторінок надалі залишається однією із основних сфер застосування JS. Проте, окрім цього, мова зайшла широке застосування у таких галузях як:

- розробка серверної частини (Node.js)
- десктопних застосунків (Electron.js)
- мобільних застосунків (ReactNative, ApacheCordova, Ionic, Framework 7)
- і т д.

JS є реалізацією стандарту ECMAScript для скриптових мов програмування. Найновішою версією стандарту є 9-е видання або ж ECMAScript 2018.

Налаштування середовища розробки

- 1) Як середовище розробки рекомендується використовувати візуальний редактор/IDE VS Code <https://code.visualstudio.com>.
- 2) Завантажуємо із сайту <https://nodejs.org>, та інсталуємо останню версію **Node.js** для вашої операційної системи.
- 3) Перевіряємо установку виконавши у терміналі команди
 - a. **node -v**
 - b. **npm -v**
- 4) Завантажуємо із сайту <https://git-scm.com/downloads>, та інсталуємо останню версію **git** для вашої операційної системи;
- 5) Створюємо директорію **lab1** у якому знаходитимуться вихідні коди лабораторної роботи.
- 6) Запускаємо команду '**npm init**':
 - a. *package name:* (code) > lab1
 - b. *version* (1.0.0)
 - c. *description:* > Lab1
 - d. *entry point:* > main.js
 - e. *test command:*
 - f. *git repository:*
 - g. *keywords:*
 - h. *author:* > {yourname}
 - i. *license:* (ISC) > MIT
- 7) Запускаємо команду у терміналі (або Git Bash для Windows) '**git init**'.
- 8) Створюємо файл **.gitignore** у якому прописуємо стрічку **node_modules/**.
- 9) Виконуємо команду '**./node_modules/.bin/eslint -init**':
 - a. *How would you like to use ESLint?* > To check syntax, find problems, and enforce code style
 - b. *What type of modules does your project use?* > JavaScript modules (import/export)

- c. *Which framework does your project use?* > None of these
- d. *Where does your code run?* > <a> (both)
- e. *How would you like to define a style for your project?* > Use a popular style guide
- f. *Which style guide do you want to follow?* > Standard
- g. *What format do you want your config file to be in?* > JSON

10) У файл **main.js** записуємо

```
'use strict'  
  
console.log('Hello Node + JS')
```

- 11) Виконуємо команду **'node .\main.js'**. Якщо на екрані є помилки та попередження виправляємо їх та запускаємо команду повторно.
- 12) Виконуємо команду **'npm eslint main.js'**. Якщо вивід не порожній уважно читаємо та виправляємо помилки.
- 13) Виконуємо команду **'git status'**.
- 14) Виконуємо команду **'git add .'**.
- 15) Виконуємо команду **'git commit -m "Initial commit"'**.

Рекомендації щодо виконання

Завдання 1-6 повинні знаходитися у різних файлах та містити у собі виклики із наборами тестових даних

Завдання 7 має бути реалізоване у вигляді Node.js модулів та включатися з допомогою `require` у головному файлі програми.

Використані джерела

- 1. <https://developer.mozilla.org/uk/docs/Web/JavaScript>
- 2. <https://habr.com/ru/post/434686/>
- 3. Фленаген Д. JavaScript. Подробное руководство – СПб.: Питер, 2013. — 1080 с.

Варіанти робіт

Варіант 1

1. Реалізувати розбиття стрічки на слова у масив. Допустим є написання функцій.
2. Написати функцію, яка поверне усі перестановки слова
3. Написати функцію, яка порахує факторіал натурального числа
4. Знайти елемент із найбільшою частотою повторень
5. Написати функцію, яка поверне кількість днів для заданого місяця та року.
6. Написати реалізацію швидкого сортування масиву
7. Реалізувати з допомогою прототипів або класів такі інформаційні об'єкти:
 - a. Пасажир
 - b. Квиток
 - c. Потяг
 - d. Продані квитки

Реалізувати такі можливості системи:

- a. Додавання нового пасажир в колекцію
 - b. Редагування пасажир в колекції
 - c. Видалення пасажир з колекції
 - d. Пошук одного пасажир в колекції
 - e. Додавання потяга в колекцію
 - f. Редагування потяга в колекції
 - g. Видалення пасажир з колекції
 - h. Пошук одного потяга в колекції
 - i. Покупка пасажиром квитка на потяг
 - j. Зміна квитка із одного потяга на інший
 - k. Скасування покупки квитка
 - l. Пошук потяга на який продали найбільше/найменше квитків
- Усі колекції реалізувати у вигляді масивів `Array`

Варіант 2

1. Реалізувати розбиття стрічки на слова у масив. Допустим є написання функцій.
2. Написати функцію, яка посортує слово у алфавітному порядку
3. Написати функцію, яка найбільший спільний дільник двох додатних чисел
4. Написати функцію, яка перемішає масив випадковим чином
5. Написати функцію, яка поверне кількість днів для заданого місяця та року.
6. Написати реалізацію сортування масиву об'єктів `User` за значенням поля `name`
7. Реалізувати з допомогою прототипів або класів такі інформаційні об'єкти:
 - a. Порт
 - b. Корабель
 - c. Пристань
 - d. Кораблі на пристані

Реалізувати такі можливості системи:

- a. Додавання нового порту в колекцію
 - b. Редагування порту в колекції
 - c. Видалення порту з колекції
 - d. Пошук одного порту в колекції
 - e. Додавання корабля в колекцію
 - f. Редагування корабля в колекції
 - g. Видалення корабля з колекції
 - h. Пошук одного корабля в колекції
 - i. Додавання пристані до порту
 - j. Видалення пристані із порту
 - k. Прибуття/Відбуття корабля від пристані
 - l. Пошук усіх кораблів на пристані
- Усі колекції реалізувати у вигляді масивів `Array`

Варіант 3

1. Реалізувати розбиття стрічки на слова у масив. Допустим є написання функцій.
2. Написати функцію, яка поверне знайде чи входить шукана стрічка у задану
3. Написати функцію, яка порахує факторіал натурального числа
4. Реалізувати алгоритм бінарного пошуку
5. Написати функцію, яка знайде максимальну дату у масиві
6. Написати реалізацію сортування бульбашкою масиву
7. Реалізувати з допомогою прототипів або класів такі інформаційні об'єкти:
 - a. Замовник
 - b. Проект
 - c. Виконавець
 - d. Проекти на виконанні

Реалізувати такі можливості системи:

- a. Додавання нового замовника в колекцію
 - b. Редагування замовника в колекції
 - c. Видалення замовника з колекції
 - d. Пошук одного замовника в колекції
 - e. Додавання проекту в колекцію
 - f. Редагування проекту в колекції
 - g. Видалення проекту з колекції
 - h. Додавання виконавця в колекцію
 - i. Редагування виконавця в колекції
 - j. Видалення виконавця з колекції
 - k. Пошук усіх проектів замовника
 - l. Пошук усіх проектів виконавця
- Усі колекції реалізувати у вигляді асоціативних масивів `Array`

Варіант 4

1. Написати програму для генерування випадкового шістнадцяткового числа. Допустим є написання функцій.
2. Написати функцію, яка вставить стрічку у відповідну позицію іншої стрічки
3. Написати функцію, яка посортує масив вставкою
4. Знайти елемент із найбільшою частотою повторень
5. Написати функцію, яка поверне кількість днів, яка пройшла від початку року.
6. Написати реалізацію рекурсивного обходу дерева
7. Реалізувати з допомогою прототипів або класів такі інформаційні об'єкти:
 - a. Космічна станція
 - b. Планета
 - c. Вантаж
 - d. Доставлені вантажі

Реалізувати такі можливості системи:

- a. Додавання нової станції колекцію
 - b. Редагування станції в колекції
 - c. Видалення станції з колекції
 - d. Пошук однієї станції в колекції
 - e. Додавання планети в колекцію
 - f. Редагування планети в колекції
 - g. Видалення планети з колекції
 - h. Пошук планети в колекції
 - i. Додавання вантажу в колекцію
 - j. Редагування вантажу в колекції
 - k. Видалення вантажу з колекції
 - l. Пошук усіх станцій планети
- Усі колекції реалізувати у вигляді масивів `Array`

Варіант 5

1. Реалізувати злиття масиву слів у стрічку. Допустим є написання функцій.
2. Написати функцію, яка першу літеру кожного слова речення переведе у верхній регістр
3. Написати функцію, яка порахує перші n числа Фібоначі
4. Написати функцію, яка видалить усі дублі із масиву стрічок
5. Написати функцію, яка порахує різницю між датами у днях/тижнях/секундах.
6. Написати реалізацію сортування злиттям масиву
7. Реалізувати з допомогою прототипів або класів такі інформаційні об'єкти:
 - a. Магазин
 - b. Товар
 - c. Склад
 - d. Товари на складі

Реалізувати такі можливості системи:

- a. Додавання нового магазину в колекцію
 - b. Редагування магазину в колекції
 - c. Видалення магазину з колекції
 - d. Пошук одного магазину в колекції
 - e. Додавання товару в колекцію
 - f. Редагування товару в колекції
 - g. Видалення товару з колекції
 - h. Пошук одного товару в колекції
 - i. Додавання складу в колекцію
 - j. Редагування складу в колекції
 - k. Видалення складу з колекції
 - l. Пошук складу в колекції
 - m. Доставка товару на склад
 - n. Видалення товару на склад
 - o. Трансфер товару із складу на склад
 - p. Відвантаження товару в магазин із складу
- Усі колекції реалізувати у вигляді асоціативних масивів `Array`