

ПОРІВНЯННЯ ПРОДУКТИВНОСТІ СТРУКТУР ДАНИХ MAP ТА OBJECT У ВЕБ-ДОДАТКАХ НА JAVASCRIPT

Вінницький національний технічний університет

Анотація

У роботі досліджуються реалізації таблиць відповідності ключ-значення Object та Map на JavaScript. Вказано переваги та недоліки даних структур. Визначається важливість продуктивності веб-додатку. Створена програма для вимірювання швидкості виконання операцій у структурах Object та Map. Порівнюється час виконання операцій додавання, читання та видалення елементів у даних структурах.

Ключові слова: JavaScript, Map, Object, продуктивність, веб-додаток, структури даних.

Abstract

The report examines implementations of key-value mapping tables Object and Map in JavaScript. Reviews the advantages and disadvantages of the structures. Determines the importance of performance in web applications. A program is created to measure the speed of operation execution in Object and Map structures. Compares the execution time of adding, reading and deleting operations in the structures.

Keywords: JavaScript, Map, Object, efficiency, web application, data structures.

Вступ

У JavaScript є дві реалізації таблиць відповідності ключ-значення: Object та Map. Обидві реалізації дозволяють встановлювати ключі до значень, отримувати значення та видаляти їх. Є декілька відмінностей даних структур. Object дозволяє тільки ключі String або Symbol, коли Map дозволяє будь-які ключі. Порядок ключів в Object існує, але складний та жоден механізм не може перебрати всі елементи. Map має порядок за часом встановлення ключів. Для отримання кількості елементів в Object потрібно порахувати довжину масиву ключів, що повертається з Object.keys(), а кількість елементів у Map можна отримати за допомогою властивості size [1, 2].

Отже, для реалізації таблиць відповідності можна використовувати Object та Map. Мова JavaScript стала одною з найпопулярніших мов програмування. Продуктивність роботи програми важлива для швидкої відповіді програми без перенавантаження процесору та надмірного споживання енергії [3].

Порівняння продуктивності структур Object та Map створить можливість зменшення часу обробки інформації. Створена програма, що вимірює час виконання додавання, читання та видалення елементів в Object та Map.

Результати дослідження

Для порівняння продуктивності роботи Object та Map потрібно вказати кількість створених ключів та кількість ітерацій. Для кожної структури створюються однакові випадкові ключі, вимірюється час встановлення ключів до випадкових значень, отримання значень із ключів та видалення ключів із структур. Для кожної операції обчислюється середній час виконання за одну ітерацію та виводиться у таблицю.

На рисунку 1 зображено продуктивність роботи Object та Map із малою кількістю ключів. Кожна операція виконується незначно довше в Object ніж Map, тому потрібно перевірити для більшої кількості ключів.

500	- кількість ключів.		
10000	- кількість ітерацій.		
Почати			
	Додавання	Читання	Видалення
Map	0.020 мс	0.000 мс	0.011 мс
Object	0.024 мс	0.005 мс	0.013 мс

Рисунок 1 - Середній час виконання операцій додавання, читання та видалення в Object та Map для 500 ключів

На рисунку 2 зображено продуктивність роботи структур із більшою кількістю ключів. Структура Map виконує операції додавання та видалення ключів значно швидше ніж Object, а операція читання приблизно однакова.

50000	- кількість ключів.		
100	- кількість ітерацій.		
Почати			
	Додавання	Читання	Видалення
Map	4.826 мс	2.006 мс	3.508 мс
Object	7.192 мс	2.181 мс	5.971 мс

Рисунок 2 - Середній час виконання операцій додавання, читання та видалення в Object та Map для 50000 ключів

Висновки

Модифікація структури Map виконується швидше ніж структури Object. При збільшенні кількості ключів зростає різниця часу виконання операцій додавання та видалення ключів між Object та Map. Час виконання операції читання майже не відрізняється в Object та Map. Отже, при малій кількості ключів можна використовувати обидві дані структури без зниження продуктивності, а при роботі з великою кількістю ключів використання структури Map підвищить продуктивність веб-додатку.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Map - JavaScript | MDN [Електронний Ресурс] - Режим доступу: https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Map (дата звернення: 05.01.2026).
2. ECMAScript® 2026 Language Specification [Електронний Ресурс] - Режим доступу: <https://tc39.es/ecma262> (дата звернення: 05.01.2026).
3. Selakovic, Marija. Performance issues and optimizations in JavaScript. [Електронний Ресурс] / Marija Selakovic, Michael Pradel // Proceedings of the 38th International Conference on Software Engineering. - Texas, Austin: ACM, 2016. - С. 61-72. <https://dl.acm.org/doi/pdf/10.1145/2884781.2884829> (дата звернення: 05.01.2026).

Ремовський Юрій Андрійович - студент групи 2ПКТ-246, факультет інтелектуальних інформаційних технологій та автоматизації, Вінницький національний технічний університет, м. Вінниця, email: removskiy.edu@gmail.com

Богач Ілона Віталіївна – к.т.н., доцент, професор кафедри автоматизації та інтелектуальних інформаційних технологій, факультет інтелектуальних інформаційних технологій та автоматизації, Вінницький національний технічний університет, м.Вінниця, e-mail: ilona.bogach@gmail.com

Removskiy Yuriy Andriyovych - student of group 2PCT-24b, Faculty of Intellectual Information Technologies and Automation, Vinnytsia National Technical University, Vinnytsia, email: removskiy.edu@gmail.com

Bogach Ilona Vitaliivna – PhD, Associate Professor of Automation and Intelligent Information Technologies, Faculty of Intelligent Information Technology and Automation, Vinnytsia National Technical University, Vinnytsia, e-mail: ilona.bogach@gmail.com