



JavaScript для початківців

Змінні та типи даних





Охріменко Дмитро
MCT



_okhrimenko



dmitriy.okhrimenko

co-founder
CyberBionic Systematics

co-founder
ITVDN



Навчання з подальшим стажуванням та допомогою у працевлаштуванні

The image displays three career roadmaps for students, each showing a progression of skills and technologies. The roadmaps are color-coded: Java Developer (orange), C#/.NET Full-stack Developer (blue/green), and Web Full-stack Developer (blue/green). Each roadmap starts with a 'Start' node and branches out into different paths. Icons indicate whether a course is for both frontend and backend (blue/green circle) or just one (orange/red circle).

Java Developer Roadmap for Students

- Start
- Java Starter
- Git
- Java Essential
- Java Professional
- MySQL
- JDBC & H
- Web
- Android Professional
- Android Essential
- Android Design Support Library
- Android User Interface
- SQLite: basic games & Android
- Apkintertype Android Application
- Unit testing in Android
- Android Dev

C#/.NET Full-stack Developer Roadmap for Students

- Start
- Unity Starter
- Unity Essential
- Unity Advanced
- Game Developer
- CR/.NET
- HTML5 & CSS3
- JavaScript Essential
- ASP.NET MVC Fundamentals
- ASP.NET WEB API 2
- Entity Framework 6
- ASP.NET Core Backend Developer
- ASP.NET Core Web API 2
- ASP.NET Core MVC 6
- UWP Developer
- UWP
- UWP Community toolkit

Web Full-stack Developer Roadmap for Students

- Start
- HTML5 & CSS3
- JavaScript Essential
- Git
- JavaScript Advanced
- JavaScript (ES6, ES7, ES8, ES9)
- jQuery
- JavaScript Patterns
- Bootstrap 5
- Frontend Developer
- Angular Developer
- TypeScript
- Angular CLI
- Angular Advanced
- React Developer
- React Essential
- React Advanced
- Vue.js Developer
- Vue.js
- Python Developer
- Python Starter
- Python Essential
- Python Advanced
- Django
- PHP Developer
- PHP Starter
- PHP Essential
- PHP Professional
- YII 2
- Symfony
- Laravel
- Ruby Developer
- Ruby Starter
- Ruby Essential
- Ruby On Rails
- Node.js Developer
- Node.js
- Mongo DB
- Backend

Умовні позначення:
 ○ Обов'язкові до вивчення курси
 ○ Не обов'язкові курси

План

- Що таке змінна та константа
- Типи даних
- Оператори
- Умовні конструкції

Змінні

Змінна – іменована область пам'яті, яка зберігає у собі певне значення, що можна змінити.

let – ключове слово для створення змінних.

= - оператор присвоювання, з його допомогою можна встановити чи змінити значення змінної.

```
let message;  
message = "Hello world";
```



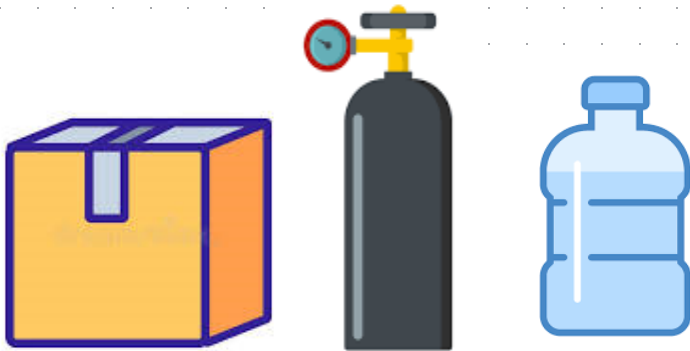
Змінні можна визначити за допомогою ключового слова **var**, яке вважається застарілим. Існують деякі відмінності в роботі змінних, створених за допомогою ключового слова **var**, які будуть розглянуті в наступних курсах.

```
let message;  
message = "Hello world";  
message = "Привіт світ";  
let testMessage = "foo";  
message = testMessage;
```



Типи даних

Тип даних – допустима множина значень. Тип даних описує набір операцій, які можна над цими даними проводити.



JavaScript – **динамічно типізована** мова програмування. Під час створення змінної її тип не вказується, але змінна зв'язується з типом при присвоєнні значення.

Тип даних явно не вказується.

При зміні значення змінної може змінитися і тип даних змінної.

Типи даних

number	для зберігання будь-яких чисел
bigint	цілі числа довільної довжини
string	для зберігання рядків
boolean	логічні значення true чи false
null	для визначення невідомого значення
undefined	для ненаданих значень
object	складні структури даних
symbol	унікальні ідентифікатори

```
let x = 10;  
let x = 10n;  
let x = "hello";  
let x = true;  
let x = null;  
let x = undefined;  
let x = { }  
let x = Symbol("test");
```


Задача 1

Написати програму, яка обчислює середнє арифметичне трьох чисел, введених користувачем, та виводить результат на екран.

Кроки для виконання завдання:

1. Введення трьох чисел:

- Використайте `prompt` для введення трьох чисел користувачем.

2. Перетворення введених даних на числа:

- Перетворіть введені дані з рядків на числа, використовуючи `parseFloat` або `Number`.

3. Обчислення середнього арифметичного:

- Використайте арифметичні оператори для обчислення суми трьох чисел та поділу на 3.

4. Виведення результату:

- Використайте `alert` для виведення середнього арифметичного.

Задача 2

Написати програму, яка обчислює вік людини на основі введеного року народження, порівнює його з поточним роком і виводить повідомлення, чи ця людина досягла повноліття (18 років).

Кроки для виконання завдання:

1. Введення року народження:

- Використайте prompt для введення року народження користувачем.

2. Обчислення віку:

- Збережіть поточний рік у змінну.
- Використайте арифметичний оператор для обчислення віку (різниця між поточним роком і роком народження).

3. Перевірка повноліття:

- Використайте оператори порівняння для перевірки, чи вік більше або дорівнює 18.

4. Виведення результату

Задача 3

Написати програму, яка перетворює температуру з Цельсія у Фаренгейт і навпаки, залежно від вибору користувача.

Кроки для виконання завдання:

1. Вибір користувача:

- Використайте prompt для введення вибору користувачем: перетворити з Цельсія у Фаренгейт або з Фаренгейта у Цельсій.

2. Введення температури:

- Використайте prompt для введення температури, яку потрібно перетворити.

3. Перетворення температури та вивід результату

- Використайте умовні оператори для визначення вибору користувача і застосування відповідної формули для перетворення температури:

- З Цельсія у Фаренгейт: $F = 9/5 * t + 32$
- З Фаренгейта у Цельсій: $C = 5/9 * (t - 32)$

Title