

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ТАРАСА
ШЕВЧЕНКА
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КАФЕДРА ІНФОРМАЦІЙНИХ СИСТЕМ ТА ТЕХНОЛОГІЙ

Лабораторна робота №2
з дисципліни « **Технології програмування**»

Варіант №21

Виконав:

студент гр. ІР-21

Анна Шмідт

Перевірив:

Пирог Микола
Володимирович

Зараховано від

____.____.____

(підпис
викладача)

Київ-2025

ЛАБОРАТОРНА РОБОТА №2

JavaScript. Data types, typeof, type casting, arrays, date, object

Мета: навчитися працювати з масивами та датами у JavaScript, закріпити навички перевірки типів і валідації введених даних.

ЗАВДАННЯ

- У всіх завданнях необхідно:
- перевіряти правильність введення (валідація даних);
 - виконувати перевірку типів (typeof, isNaN тощо);
 - у разі помилки виводити повідомлення.

21	Перевірити, чи сьогодні день народження користувача.	Користувач вводить 10 імен із клавіатури. Створити масив введених імен. Об'єднати всі елементи в один рядок через кому.
----	--	---

Скрін коду

```
function runTasks() {
  let continueProgram = true;

  // Головний цикл програми
  while (continueProgram) {

    // --- Завдання 1: Перевірити, чи сьогодні день народження користувача (ДД.ММ.РРРР). ---
    console.log("\n--- Завдання 1: Перевірка дня народження ---");

    // Змінений запит дати народження у користувача
    let birthdayInput = prompt("Завдання 1: Введіть дату свого народження у форматі ДД.ММ.РРРР (наприклад, 15.05.1990):");

    // 1.1 Валідація і перевірка введення
    if (birthdayInput === null || birthdayInput.trim() === "") {
      console.error("Помилка введення: Ви нічого не ввели або скасували. Перехід до наступного завдання.");
    } else {
      let birthday;
      let isValidDate = false;

      // Спроба парсингу дати у форматі ДД.ММ.РРРР
      const parts = birthdayInput.split('.');

      // Перевірка на коректну кількість частин (ДД, ММ, РРРР)
      if (parts.length === 3) {
        // Отримуємо день, місяць, рік
        const day = parseInt(parts[0], 10);
        const month = parseInt(parts[1], 10); // Місяць: 1-12
        const year = parseInt(parts[2], 10);

        // Перевірка, чи всі частини є числами та чи місяць/день у розумних межах
        if (!isNaN(day) && !isNaN(month) && !isNaN(year) && month >= 1 && month <= 12) {
          // Створюємо об'єкт Date. У JavaScript місяці йдуть з 0 до 11, тому (month - 1)
          birthday = new Date(year, month - 1, day);

          // 1.2 Фінальна перевірка коректності дати (валідація дати)
          // Додатково перевіряємо, чи створена дата відповідає введеним Дню та Місяцю.
          // Це потрібно для відловлювання неіснуючих дат, наприклад, 31.02.2000.
          if (birthday.getFullYear() === year && birthday.getMonth() === month - 1 && birthday.getDate() === day) {
            isValidDate = true;
          }
        }
      }
    }
  }
}
```



```
--- Завдання 1: Перевірка дня народження ---
Завдання 1: Введіть дату свого народження у форматі ДД.ММ.РРРР (наприклад, 15.05.1990
):15.10.20000
🎉 Вітаємо! Сьогодні, 15.10, Ваш День Народження! 🎉

--- Завдання 2: Введення та обробка 10 імен ---
Завдання 2: Введіть ім'я №1 з 10:q
Завдання 2: Введіть ім'я №2 з 10:w
Завдання 2: Введіть ім'я №3 з 10:e
Завдання 2: Введіть ім'я №4 з 10:r
Завдання 2: Введіть ім'я №5 з 10:t
Завдання 2: Введіть ім'я №6 з 10:y
Завдання 2: Введіть ім'я №7 з 10:u
Завдання 2: Введіть ім'я №8 з 10:i
Завдання 2: Введіть ім'я №9 з 10:o
Завдання 2: Введіть ім'я №10 з 10:p

Успішно введено всі 10 імен:
Об'єднаний рядок імен: q, w, e, r, t, y, u, i, o, p
Бажаєте обрати знову (введіть 'так') чи завершити (введіть 'ні')?
```

Контрольні питання:

- 1. Які типи даних існують у JavaScript і які з них належать до примітивних? Відповідь:** Існує 8 типів. **Примітивні (7):** String, Number, BigInt, Boolean, Undefined, Null, Symbol. **Непримітивний (1):** Object.
- 2. Чим відрізняється значення null від undefined у плані використання та перевірки? Відповідь:** **undefined** призначається системою, якщо змінній не було присвоєно значення. **null** призначається розробником свідомо, означаючи порожнє посилання або відсутність значення.
- 3. Який результат дасть вираз: typeof NaN? Чому саме такий? Відповідь:** Результат: "number". **Чому:** NaN (Not a Number) є спеціальним числовим значенням і належить до примітивного типу Number.
- 4. Яке значення поверне Number(" ") і чому? Відповідь:** Результат: 0. **Чому:** Функція Number() видаляє пробіли, перетворюючи " " на порожній рядок (""), який конвертується в 0.

5. Який результат дає `Boolean("0")` і чим він відрізняється від `Boolean(0)`? Відповідь: `Boolean("0")` → `true` (непорожній рядок).
`Boolean(0)` → `false` (число 0 є "falsy").

6. У чому полягає різниця між масивом і об'єктом у JavaScript, якщо обидва є об'єктами? Відповідь: Об'єкт зберігає неупорядковані дані за довільними ключами. Масив зберігає упорядковані дані за числовими індексами (з 0) і має властивість `length`.

7. Який результат виконання коду? Поясніть, у чому різниця між `slice` і `splice`.

```
let arr = [1, 2, 3];  
console.log(arr.slice(1, 2));  
console.log(arr.splice(1, 2));  
console.log(arr);
```

Відповідь: Результат: `[2]`, `[2, 3]`, `[1]`. Різниця: `slice` не змінює оригінальний масив; `splice` змінює оригінальний масив, повертаючи видалені елементи.

8. Як за допомогою `Math.random()` отримати випадкове ціле число в діапазоні від -50 до 50 включно? Відповідь:

```
const result = Math.floor(Math.random() * 101) - 50;
```

9. Який результат обчислень наступного коду? Чому саме так?

```
let d1 = new Date("2025-01-01");  
let d2 = new Date("2025-01-02");  
console.log((d2 - d1) / (1000 * 60 * 60 * 24));
```

Відповідь: Результат: `1`. Чому: Різниця дат повертається у мілісекундах. Ділення на кількість мілісекунд у добі ($1000 \times 60 \times 60 \times 24$) обчислює кількість діб між датами.

10. Який результат дасть наступний код? Чому змінилось значення в обох змінних?

```
let obj = { a: 1 };  
let copy = obj;  
copy.a = 5;
```

```
console.log(obj.a)
```

Відповідь: Результат: **5**. **Чому:** Об'єкти передаються за посиланням. obj та сору вказують на **один і той же** об'єкт у пам'яті, тому зміна через одну змінну відображається через іншу.