

Масиви

№ уроку: 9 Курс: Java Starter

Засоби навчання: Комп'ютер із встановленою IntelliJ IDEA

Огляд, мета та призначення уроку

Розгляд масивів.

Вивчивши матеріал цього заняття, учень зможе:

- Використовувати та створювати масиви.
- Виконувати перебір масивів.

Зміст уроку

1. Одновимірні масиви.
2. Двовимірні масиви.
3. Тривимірні масиви.
4. Зубчасті масиви.

Резюме

- **Масив** – іменований набір однотипних змінних, розташованих у пам'яті безпосередньо одна за одною, доступ до яких здійснюється за індексом.
- **Індекс масиву** – ціле число або значення типу, що призводить до цілого, яке вказує на конкретний елемент масиву.
- Кількість використаних індексів масиву може бути різною. Масиви з одним індексом називають одновимірними, із двома – двовимірними й так далі. Одновимірний масив нестрого відповідає вектору в математиці, двовимірний – матриці. Найчастіше застосовуються масиви з одним або двома індексами, рідше – з трьома, ще більше індексів зустрічається вкрай рідко.
- **Масив масивів** – це масив, елементи якого самі є масивами. Елементи масиву масивів можуть мати різні розміри та вимірювання. Масиви масивів іноді також називаються «не вирівняними масивами».
- Масиви масивів можна змішувати з багатовимірними масивами.
- Масив можна ініціалізувати під час оголошення. У цьому випадку специфікація рангу не потрібна, оскільки вона вже надана за кількістю елементів у списку ініціалізації.
- Під час створення масиву значення за замовчуванням числових елементів масиву задано рівним нулю, а елементи посилального типу мають значення NULL.
- Зубчастий масив є масивом масивів і тому його елементи є посилальними типами й ініціалізуються значенням `null`.
- Індексція масивів починається з нуля: масив з елементами `n` індексується від 0 до `n-1`.
- Елементи масиву можуть бути будь-яких типів, зокрема й типом масиву.
- Типи масиву є посилальними типами, похідними від абстрактного базового класу `Array`.
- Клас `Array` дає змогу використовувати багато інших корисних методів і властивостей для виконання сортування, пошуку та копіювання масивів.
- Масиви можна передавати як аргументи для параметрів методів. **Оскільки масиви є посилальними типами, метод може змінювати значення елементів.**

Закріплення матеріалу

- Що таке одновимірний масив?
- Що таке двовимірний масив?
- Що таке тривимірний масив?
- Що таке зубчастий масив?
- Чи може індекс перевищувати загальну кількість елементів масиву?

- З якого числа починають індексуватись елементи масиву?

Додаткове завдання

Використовуючи IntelliJ IDEA, створіть клас **Arrays**. Створіть масив розмірністю 10 елементів, виведіть на екран усі елементи масиву у зворотному порядку.

Самостійна діяльність учня

Завдання 1

Вивчіть основні конструкції та поняття, розглянуті на уроці.

Завдання 2

Використовуючи IntelliJ IDEA, створіть клас **CustomerArray**. Створити масив розміру N-елементів, заповнити його довільними цілими значеннями (розмір масиву задає користувач). Вивести на екран: найбільше значення масиву, найменше значення масиву, загальну суму всіх елементів, середнє арифметичне всіх елементів, вивести всі непарні значення.

Завдання 3

Використовуючи IntelliJ IDEA, створіть клас **ReversedArray**. Створити метод `myReverse(int [] array)`, який приймає як аргумент масив цілочислових елементів і повертає інвертований масив (елементи масиву у зворотному порядку). Створити метод `int [] subArray (int [] array, int index, int count)`. Метод повертає частину отриманого як аргумент масиву, починаючи з позиції, яка зазначена в аргументі `index`, розмірністю, що відповідає значенню аргументу `count`. Якщо аргумент `count` містить значення більше, ніж кількість елементів, що входять до частини вихідного масиву (від зазначеного індексу `index` до індексу останнього елемента), то під час формування нового масиву розмірністю в `count`, заповнить одиницями ті елементи, які не були скопійовані з вихідного масиву.

Завдання 4

Використовуючи IntelliJ IDEA, створіть клас **UserArray**. Створити метод, який виконуватиме збільшення довжини масиву, переданого як аргумент, на 1 елемент. Елементи масиву мають зберегти своє значення та порядок індексів. Створити метод, який приймає два аргументи, перший аргумент типу `int [] array`, другий аргумент типу `int value`. У тілі методу реалізуйте можливість додавання другого аргументу методу в масив за індексом 0, водночас довжина нового масиву має збільшитися на 1 елемент, а елементи одержуваного масиву як перший аргумент мають скопіюватися в новий масив починаючи з індексу 1.

Завдання 5

Створіть два масиви 3x3, заповніть їх, створіть третій масив порожній. У третій масив внесіть результат додавання перших двох.

Завдання 6

Створіть два масиви 3x3, заповніть їх, створіть третій масив порожній. У третій масив внесіть результат додавання перших двох, використовуючи методи.

Рекомендовані ресурси

Масиви у JAVA

<https://docs.oracle.com/javase/tutorial/java/nutsandbolts/arrays.html>

<https://docs.oracle.com/javase/specs/jls/se7/html/jls-10.html>