



# Java Starter

Змінні та типи даних



# Java Starter

Після уроку обов'язково



Повторіть цей урок у відео форматі на  
[ITVDN.com](http://itvdn.com)

Доступ можна отримати через керівництво  
вашого навчального центру



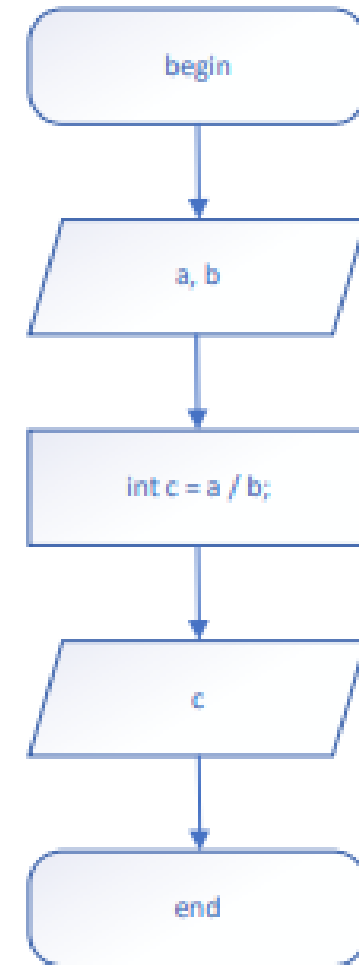
Перевірте, як Ви засвоїли даний матеріал  
на [TestProvider.com](http://testprovider.com)

## Змінні та типи даних

# Змінна

## Алгоритми. Лінійний

**Лінійний алгоритм (наслідування)** утворюється командами, що виконуються одноразово в тій послідовності, в якій вони записані.



# Змінна

## Арифметичні оператори

+ Додавання додає два значення  $x + y$

- Віднімання одного значення з іншого  $x - y$

\* Множення множить два значення  $x * y$

/ Ділення ділить одне значення на інше  $x / y$

% Ділення із виведенням залишку повертає залишок від поділу  $x \% y$

++ Інкремент збільшує значення змінної на 1  $++x$

-- Декремент зменшує значення змінної на 1  $--x$

# Змінні

## Правила іменування



Імена змінних повинні бути зрозумілі і передавати зміст значення, що в них зберігається.

`int myVariable = 2;`

Перший символ ідентифікатора

Допустимі символи	Недопустимі символи
A-Z, a-z, _ , \$	0-9, . , , , ? , : , * , > , < , = , & , # , ! і т.п.

Інші символи ідентифікатора

Допустимі символи	Недопустимі символи
A-Z, a-z, _ , \$, 0-9	@ , . , , , ? , : , * , > , < , = , & , # , ! і т.п.

# Змінні

## Правила іменування

- 1) В ідентифікаторах допустимо використовувати символи алфавіту, нижнього підкреслення та символ долар:

`myVariable, my_Variable, _MyVariable, my$Variable, $myVariable`

- 2) Використання цифр недопустимо тільки на першій позиції:

`myVariable1, my1Variable, 1MyVariable`

- 3) Не можна використовувати як ідентифікатори зарезервовані ключові слова:

~~`extends, false, import, intMyVar`~~

- 4) Мова JAVA чутлива до регістру, тому якщо написати змінні у різних регістрах – це будуть різні змінні:

`myVariable, MyVariable, myvariable`

# Ключові слова

## Keywords

**Ключові слова** — це заздалегідь визначені зарезервовані ідентифікатори, що мають спеціальні значення для компілятора.

abstract	assert	boolean	break	byte
case	catch	char	class	const
continue	default	do	double	else
enum	extends	final	finally	float
for	goto	if	implements	import
instanceof	int	interface	long	native
new	package	private	protected	public
return	short	static	strictfp	super
switch	synchronized	this	throw	throws
transient	try	void	volatile	while



**Ключові слова не можуть бути використані у якості ідентифікаторів.**



# Змінні

## Угоди щодо найменування

Специфікація мови JAVA рекомендує дотримуватись певних правил (casing conventions) під час створення ідентифікаторів.

Стиль	Опис	Приклад
Pascal casing	Кожне слово в ідентифікаторі починається з великої літери	MyMethod, Remove
Camel casing	Кожне слово, крім першого, в ідентифікаторі починається з великої літери	myCount, totalDiscount
Uppercase	Ідентифікатор складається з букв, написаних у верхньому реєстрі; таким стилем оголошуються константи	IO, XML



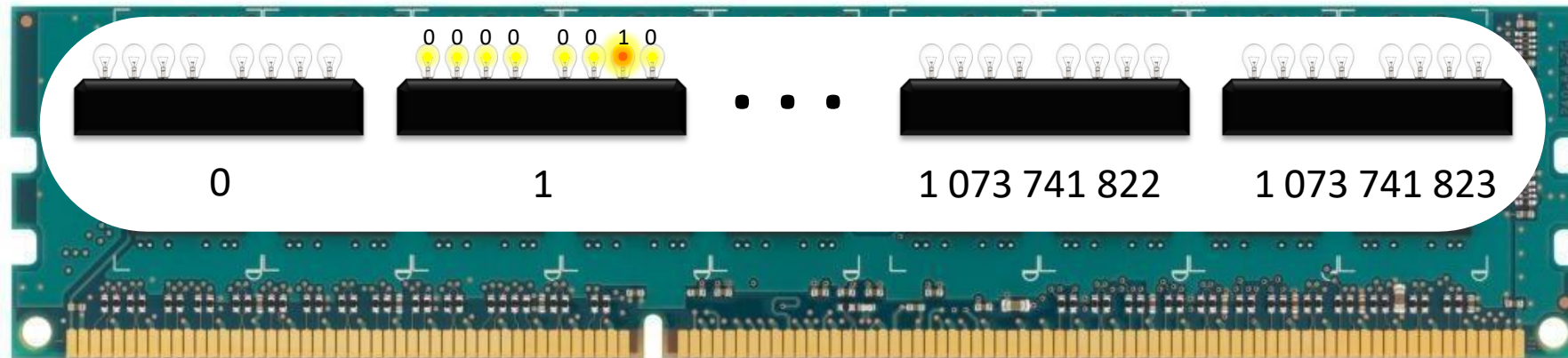
**Не рекомендується користуватися угорською нотацією та починати ідентифікатори із символу нижнього підкреслення, долара.**

# Константа

## Constant

**Константа** – це іменована область пам'яті, яка зберігає у собі деяке значення, яке не можна змінити.

```
final byte A = 2; // 0000 0010 b
```



**Спроба надати константі нове значення,  
призводить до помилки рівня компіляції.**

# Перетворення значення типу

## Casting або Type conversion

**Кастинг** – це перетворення значення змінної одного типу на значення іншого типу.

### Явний кастинг (explicit)

Перетворення виконуються лише у разі явної вказівки (у круглих дужках) типу, на який необхідно перетворити.

### Неявний кастинг (implicit)

Перетворення виконуються автоматично без втрати точності та урізання частини вихідного значення числа.

# Неявний кастинг

## Безпечний

Неявне перетворення типу (безпечне) – перетворення значення меншого типу на значення більшого або цілого в дійсне.

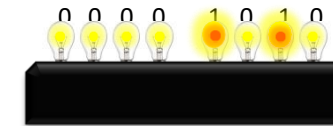
`byte a = 10;`



`short b = 0;`



`b = a;`



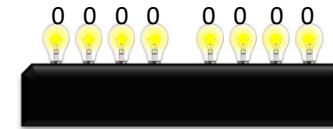
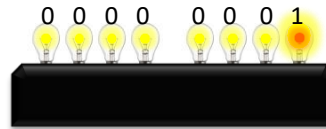
Комірка заповнюється нульовими значеннями або, якщо число було негативним, – одиницями.

# Явний кастинг

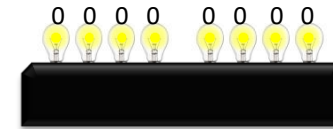
## Небезпечний

Явне перетворення типу (небезпечне) – перетворення значення більшого типу на значення меншого, або дійсного в ціле.

`short a = 256;`

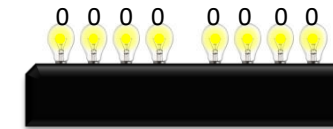


`byte b = 0;`



`b = (byte)a;`

Оператор явного перетворення  
значення типу.



**Явний кастинг вважається небезпечним, оскільки може відбутися втрата точності чи урізання числа.**

# Рядки

## Форматований виведення

Маркер підстановки  
↓  
`System.out.printf("Це число %1$s", 1);`

↑  
Елемент підстановки

Позиція елементу підстановки, що починається з одиниці (1,2,3,... і т. д.), вказівка більшої позиції призведе до помилки.

```
System.out.printf("Це числа за порядком %1$s та %2$s ", 1, 2);
```

```
System.out.printf("Виведення навпаки %2$s та %1$s ", 1, 2);
```

# Java Starter

Q&A

# Інформаційний відеосервіс для розробників програмного забезпечення

