



# Java Starter

УМОВНІ КОНСТРУКЦІЇ

# Java Starter

Після уроку обов'язково



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

Доступ можна отримати через керівництво вашого навчального центру



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

## Умовні конструкції

# Умовні конструкції

## Selection Statements

Оператор **розгалуження** (умовний оператор) – конструкція мови програмування, що забезпечує виконання певної команди (набору команд) тільки за умови істинності деякого логічного виразу.

**ЯКЩО:** (*Умова задовольняє істинності*)

**ТО:** Виконати цю серію інструкцій

**ІНАКШЕ:** Виконати цю серію інструкцій

# Умовні конструкції

## Selection Statements

Умовний оператор: `if ... else`

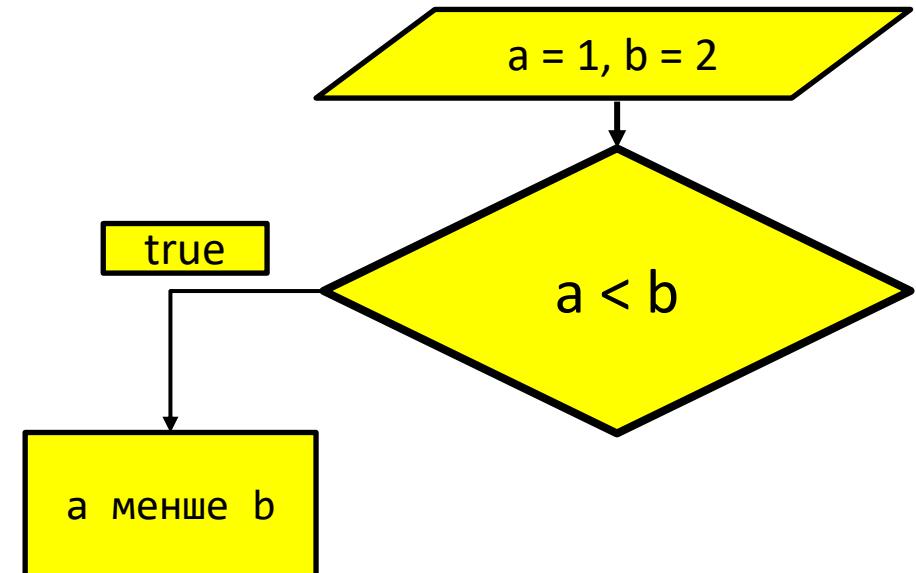
Тернарний оператор: `... ? ... : ...`

Оператор багатозначного вибору: `switch - case`

## Умовний оператор

Реалізує виконання певних команд за умови, що логічний вираз, який використовується, в умові задовольняє істинності. Результатом обчислення логічного виразу має бути **true** або **false**.

```
int a = 1, b = 2;  
  
// ЯКЩО: умова задовольняє істинності  
// ТО: виконуємо тіло умовної конструкції  
if (a < b) {  
    System.out.println("а менше b"); // Гілка 1  
}
```

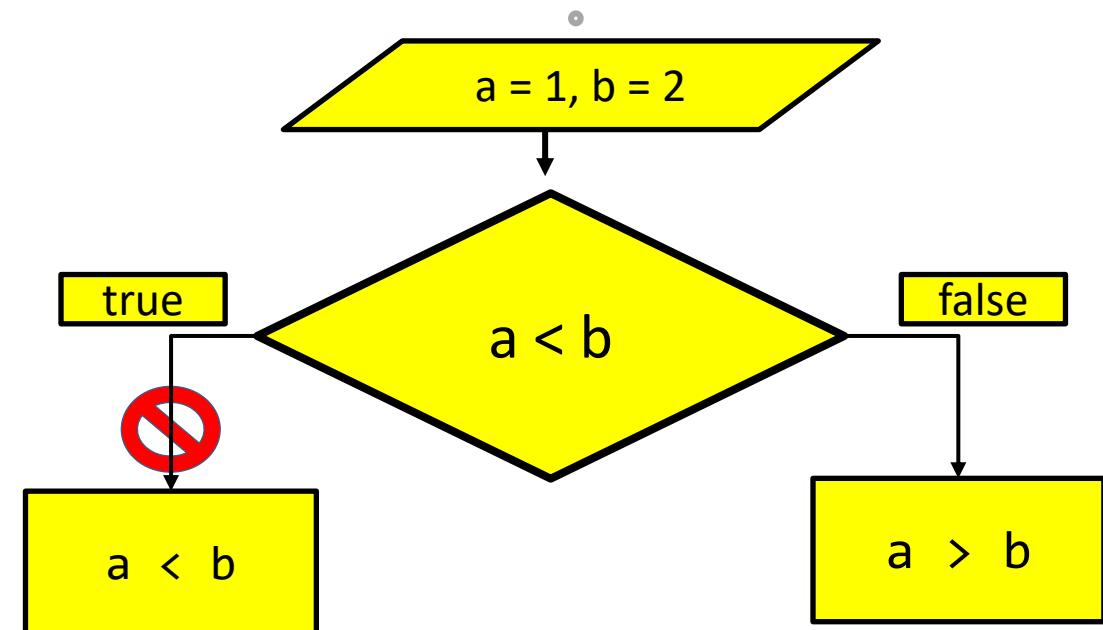


# if-else

## Умовний оператор

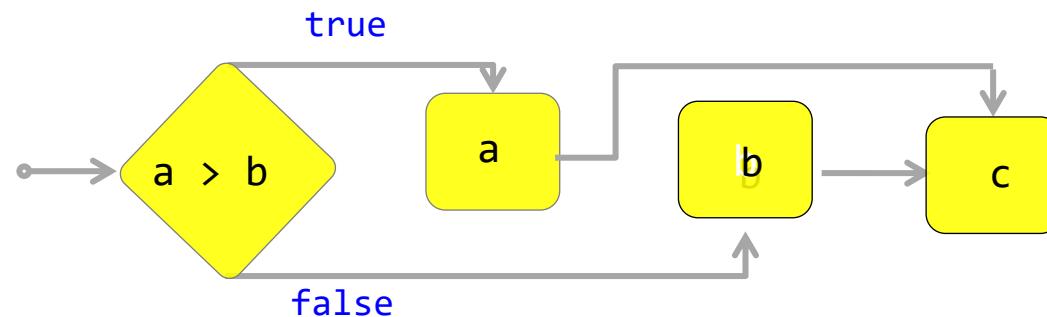
Залежно від умови виконується лише одна із двох серій команд, які входять до команди розгалуження. Якщо умова дотримана, то треба виконати серію 1, а якщо ні, то серію 2.

```
int a = 1, b = 2;  
  
// ЯКЩО: умова задовольняє істинності  
// ТО: виконуємо тіло умовної конструкції  
if (a < b) {  
    System.out.println("a < b"); // Гілка 1  
} else { // ІНАКШЕ: виконуємо тіло блоку else.  
    System.out.println("a > b"); // Гілка 2  
}
```



? :

## Тернарний (третинний) оператор



```
int a = 1, b = 2, c = 0;
```

```
// ЯКЩО: (a > b) ТО: повернути a ІНАКШЕ: повернути b;  
c = (a > b) ? a : b;
```

```
System.out.println(c);
```



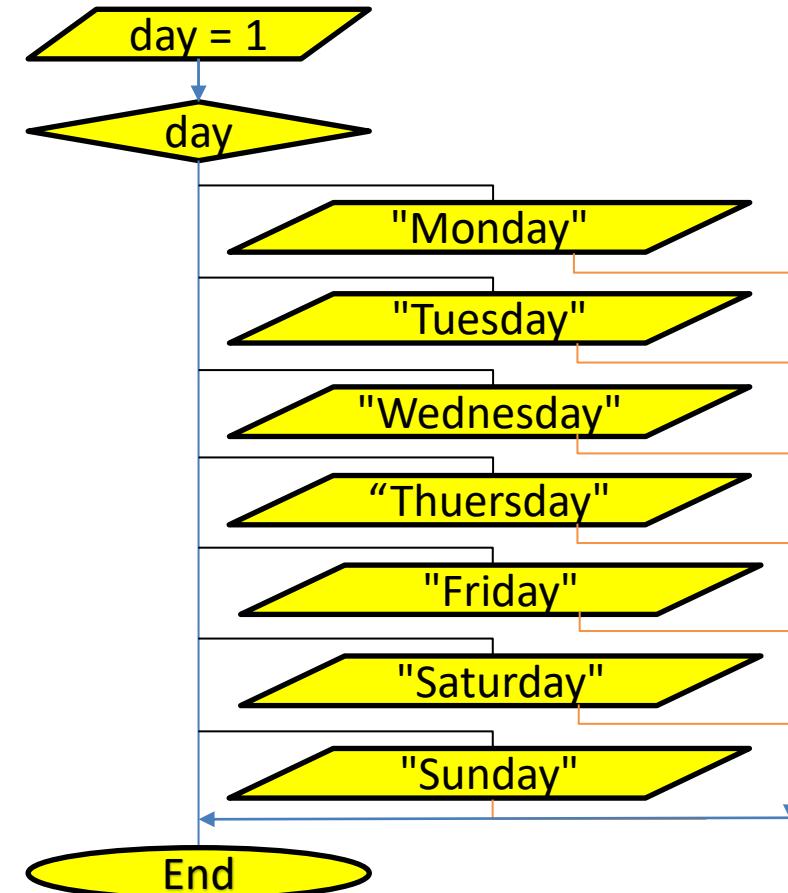
Тернарний оператор обов'язково має повернати значення, інакше буде помилка.

# switch-case

## Оператор багатозначного вибору (перемикач)

Конструкція перемикача `switch-case` має кілька гілок.

```
String day = "1";  
  
switch (day) { // (day) – вираз-селектор.  
  
    case "1": { // "1" – постійний вираз. -  
        System.out.println("Monday"); // Гілка 1  
        break;  
    }  
    case "2": { // "2" – постійний вираз.  
        System.out.println("Tuesday"); // Гілка 2  
        break;  
    }  
}
```

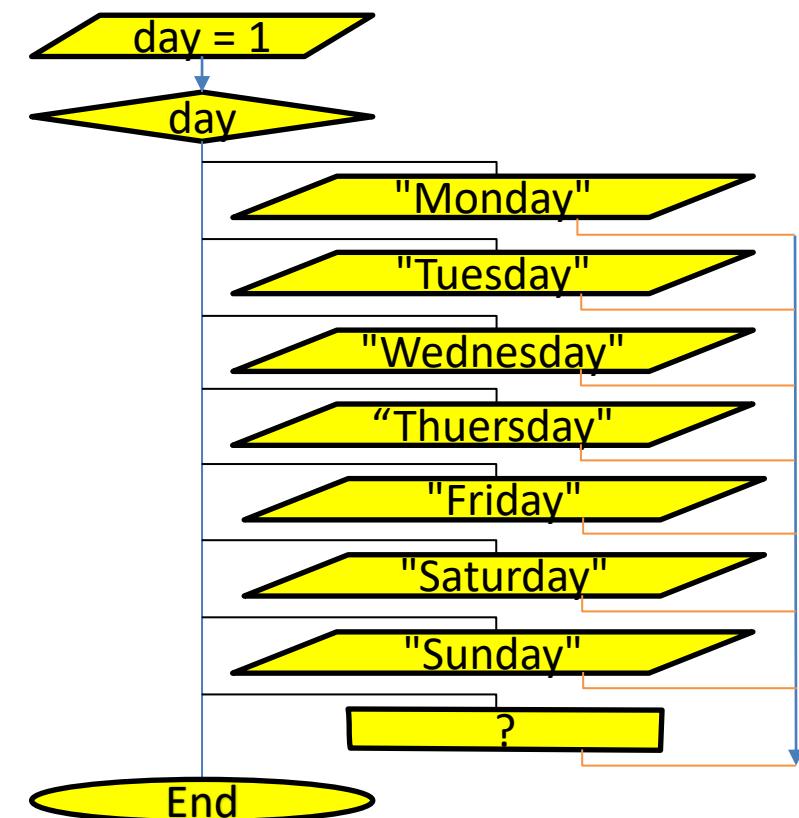


# switch-case

## Оператор багатозначного вибору (перемикач)

У конструкції перемикача **switch-case** може бути блок **default**.

```
String day = "1";  
  
switch (day) {  
    case "1": { // "1" – постійний вираз. -  
        System.out.println("Monday"); // Гілка 1  
        break;  
    }  
    case "2": { // "2" – постійний вираз.  
        System.out.println("Tuesday"); // Гілка 2  
        break;  
    }  
    default: {  
        System.out.println(?); // Гілка 3  
        break;  
    }  
}
```



## Q&A

Інформаційний відеосервіс для розробників програмного забезпечення

