



LATVIJAS  
UNIVERSITĀTE  
ANNO 1919

NACIONĀLAIS  
ATTĪSTĪBAS  
PLĀNS 2020



EIROPAS SAVIENĪBA  
Eiropas Sociālais  
fonds

IEGULDĪJUMS TAVĀ NĀKOTNĒ

# Java programmēšanas pamati

4. nodarbība

# SAZAROŠANĀS OPERATORI

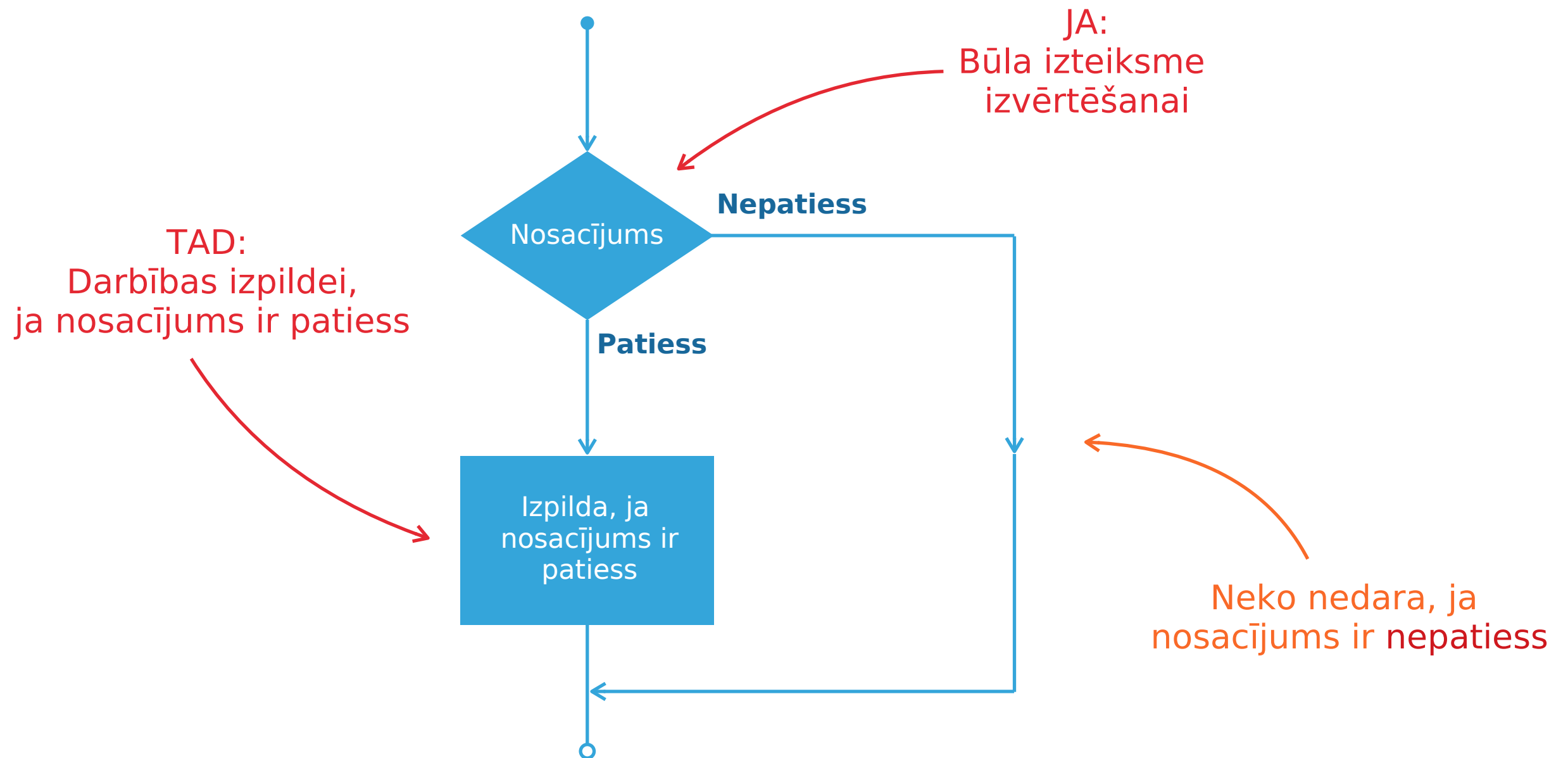


# NOSACĪJUMA OPERATORI

# NOSACĪJUMA IZTEIKSME

- ▶ **Kontrolē** koda izpildi, nosakot noteiktus nosacījumus
  - ▶ Kad nosacījums ir **patiešs** (vienāds ar 'true')
  - ▶ Kad nosacījums ir **nepatiešs** (vienāds ar 'false')
- ▶ Pastāv **divu** veidu nosacījuma izteiksmes:
  - ▶ **If** izteiksme
  - ▶ **Switch** izteiksme

# IF: BLOKSHĒMA



# IF: SINTAKSE

Atslēgas vārds, kas  
norāda uz  
nosacījuma izteiksmi

Mainīgais vai izteiksme  
ar Būla tipa rezultātu

```
if (izteiksme) {  
    //Kods izpildei  
    //Ja izteiksme ir patiesa  
}
```

# IF: PIEMĒRS

## Būla mainīgā izteiksme

```
boolean flag = true;  
  
if (flag) {  
    System.out.print("True");  
}
```

## Iekļautā Būla izteiksme

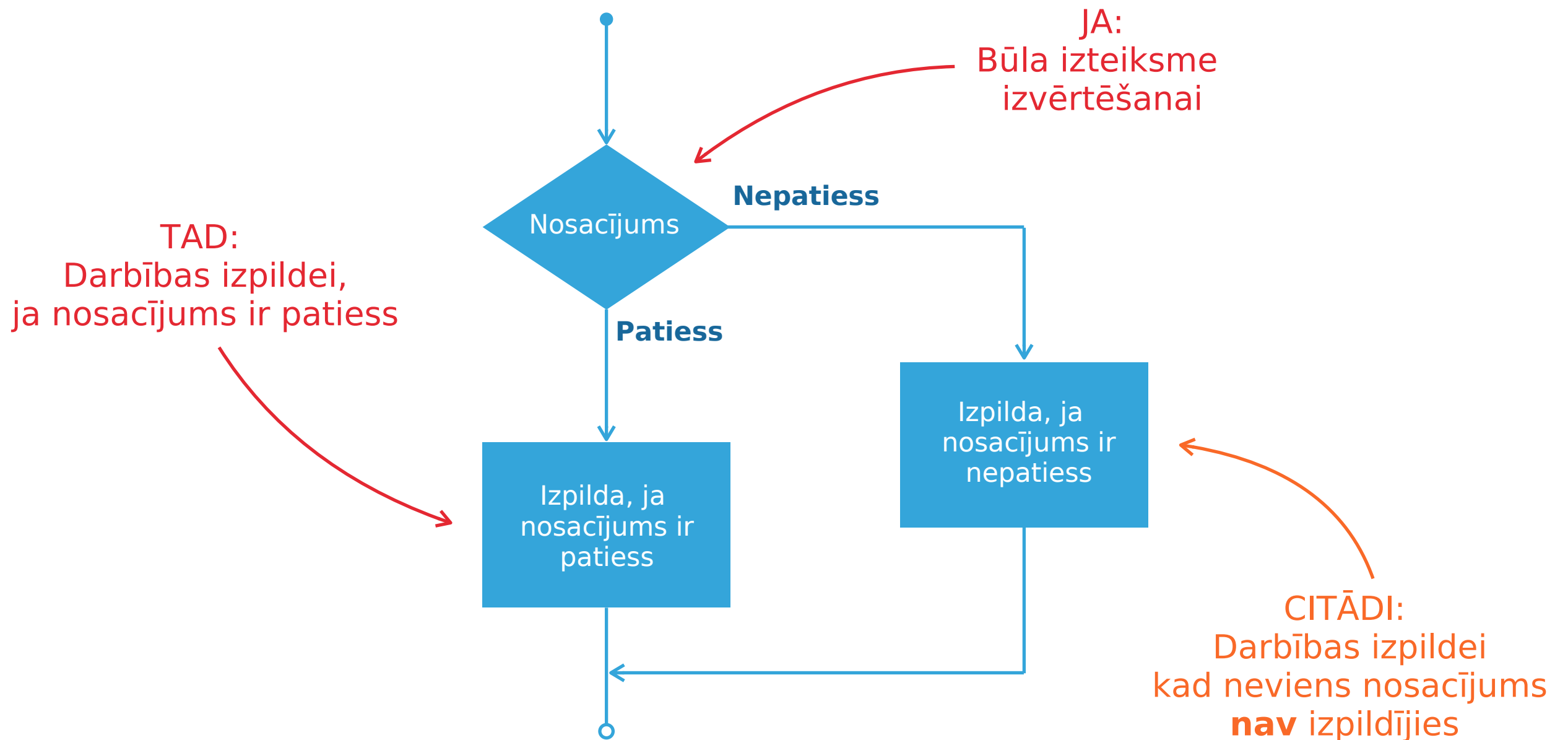
```
int x = 5;  
  
if (x > 10) {  
    System.out.print("x > 10");  
}
```

## IF: KOPSAVILKUMS

- ▶ Sastāv no **būla izteiksmes**, kam seko **viena vai vairākas** izpildes komandas
- ▶ Būla izteiksme var tik **sastādīta** no vairākām Būla tipa apakšizteiksmēm



# IF – ELSE: BLOKSHĒMA



# IF - ELSE: SINTAKSE

Atslēgvārds, kas  
norāda uz  
nosacījuma izteiksmi

Mainīgais vai izteiksme  
ar Būla tipa rezultātu

```
if (izteiksme) {  
    //Komandas izpildei  
    //Kad izteiksme ir patiesa  
}  
else {  
    //Komandas izpildei  
    //Kad izteiksme ir nepatiesa  
}
```

Atslēgvārds, kas nosaka  
alternatīvo koda bloku

# IF - ELSE: PIEMĒRS

## Būla mainīgā izteiksme

```
boolean flag = false;

if (flag) {
    System.out.print("True");
} else {
    System.out.print("False");
}
```

## Iekļautā Būla izteiksme

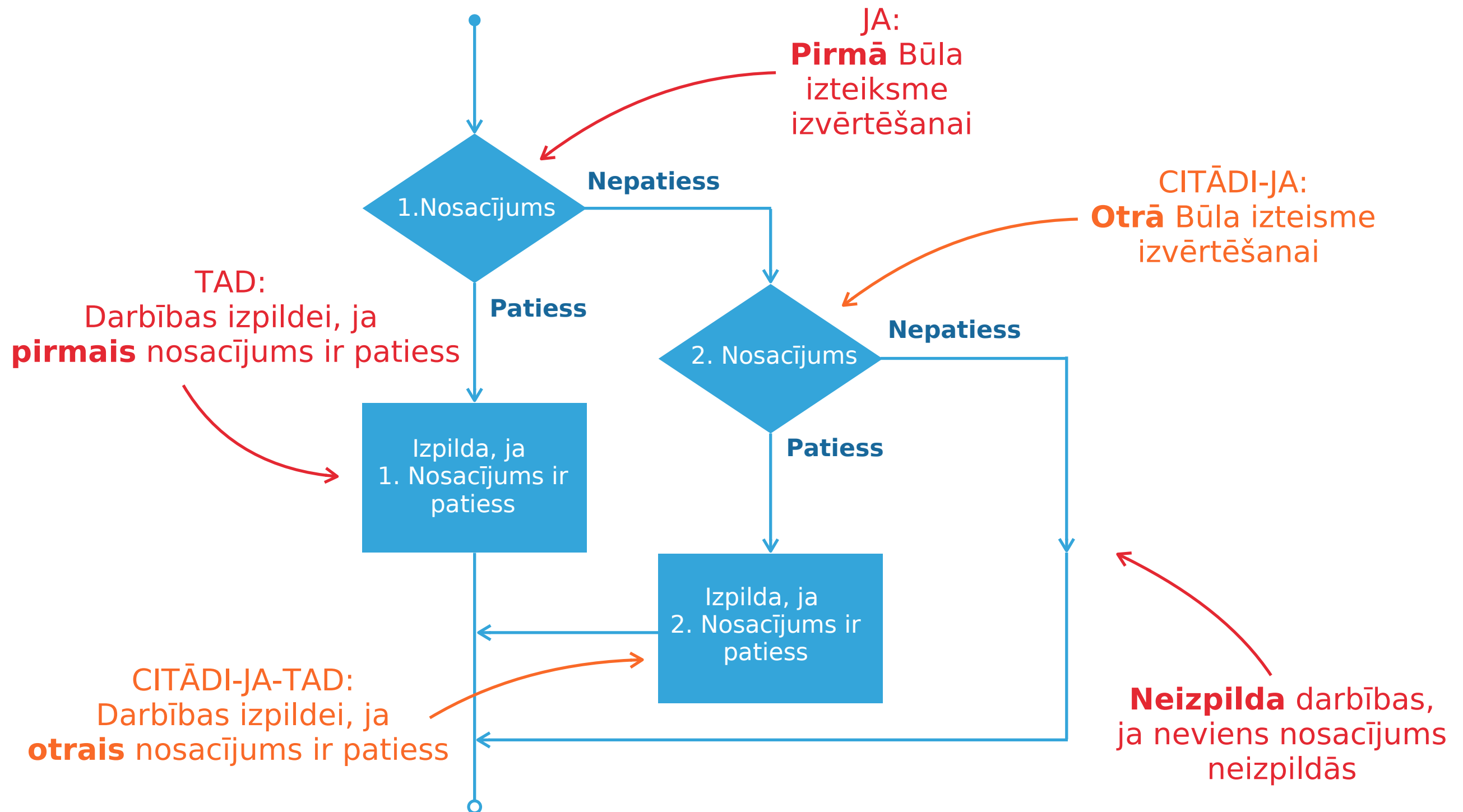
```
int x = 5;

if (x > 10) {
    System.out.print("x > 10");
} else {
    System.out.print("x =< 10");
}
```

## IF – ELSE: KOPSAVILKUMS

- ▶ **IF** izteiksmei **var tikt** papildināta ar **ELSE** izteiksmi, kas tiek izpildīta, kad Būla izteiksme ir **nepatiesa**

# IF - ELSE IF: BLOKSHĒMA



# IF - ELSE IF: SINTAKSE

Atslēgas vārds, kas  
norāda uz  
nosacījuma izteiksmi

Mainīgais vai izteiksme  
ar Būla tipa rezultātu

```
if (izteiksme1) {  
    //Komandas izpildei  
    //Kad 1. izteiksme ir patiesa  
}  
else if (izteiksme2) {  
    //Komandas izpildei  
    //Kad 2. izteiksme ir patiesa  
}
```

Atslēgvārds, kas nosaka  
alternatīvo koda bloku



# IF - ELSE IF: PIEMĒRS

## Būla mainīgā izteiksme

```
boolean flag1 = false;
boolean flag2 = true;

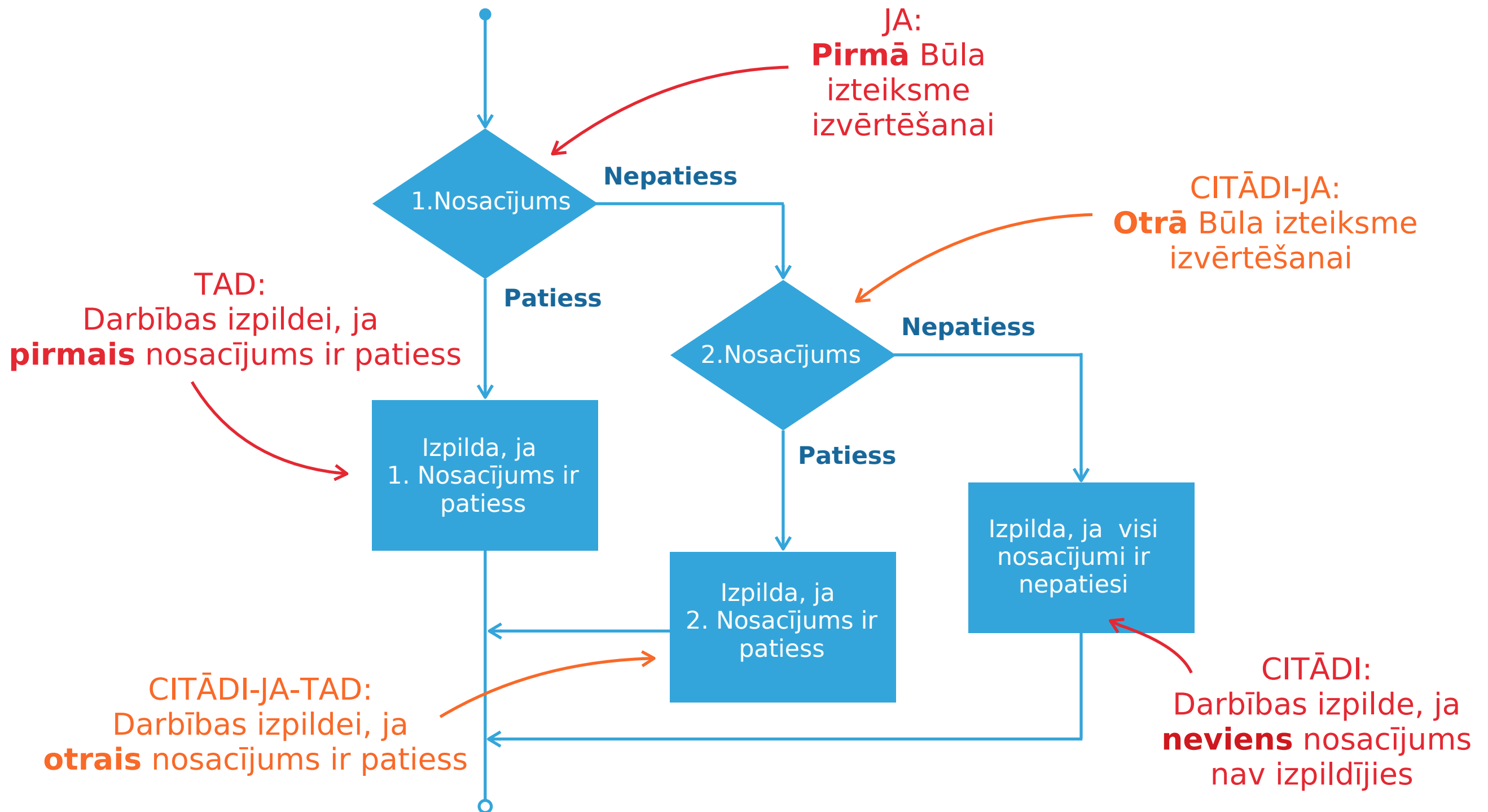
if (flag1) {
    System.out.print("flag1");
} else if (flag2) {
    System.out.print("flag2");
}
```

## Iekļautā Būla izteiksme

```
int x = 7;

if (x == 3) {
    System.out.print("x == 3");
} else if (x == 7) {
    System.out.print("x == 7");
}
```

# IF - ELSE IF - ELSE: BLOKSHĒMA



# IF - ELSE IF - ELSE: SINTAKSE

Atslēgas vārds, kas  
norāda uz  
nosacījuma izteiksmi

Mainīgais vai izteiksme  
ar Būla tipa rezultātu

Atslēgvārds, kas  
norāda alternatīvo  
nosacījuma koda bloku

```
if (izteiksme1) {  
    //Komandas izpildei  
    //Kad 1. izteiksme ir patiesa  
}  
else if (izteiksme2) {  
    //Komandas izpildei  
    //Kad 2. izteiksme ir patiesa  
}  
else {  
    //Komandas izpildei  
    //Kad visas izteiksmes ir nepatiasas  
}
```

Atslēgvārds, kas  
norāda uz alternatīvo  
koda bloku

# IF - ELSE IF - ELSE: PIEMĒRS

## Būla mainīgā izteiksme

```
boolean flag1 = false;
boolean flag2 = false;

if (flag1) {
    System.out.print("flag1");
} else if (flag2) {
    System.out.print("flag2");
} else {
    System.out.println("none");
}
```

## Iekļautā Būla izteiksme

```
int x = 7;

if (x == 3) {
    System.out.print("x == 3");
} else if (x == 7) {
    System.out.print("x == 7");
} else {
    System.out.print("NOTA");
}
```

## IF - ELSE IF – ELSE: KOPSAVILKUMS

- ▶ IF blokam var sekot **neviens** vai **viens ELSE** bloks un tam ir jābūt pēc **ELSE-IF** bloka
- ▶ IF blokam var sekot **viens** vai **vairāki ELSE-IF** bloki un tiem ir jābūt pirms **ELSE** bloka
- ▶ Kad kāds no **ELSE-IF** blokiem ir izpildījies, tad atlikušie vairs **netiks** pārbaudīti

## SWITCH IZTEIKSME

- ▶ Sniedz iespēju efektīvi strādāt ar kodu, kurš var zarties vairākos virzienos atkarībā no viena mainīgā
- ▶ **Nav** iespējams izmantot **nosacījuma izteiksmes**
- ▶ **Nav** iespējams izmantot **vairākus** mainīgos



# SWITCH: SINTAKSE

Atslēgvārds, kas  
nosaka switch  
izteiksmi

Mainīgais pret  
kuru veic pārbaudi

```
switch (singleVariable) {  
  case value1:  
    //Koda bloks izpilde  
    //Kad value1 sakrīt ar doto  
    break;  
  
  case value2:  
    //Koda bloks izpilde  
    //Kad value2 sakrīt ar doto  
    //Bez atslēgvārda break  
    //Kods izpildās tālāk pa  
    //case nosaijumiem  
    //un apstājas pie pirmā break  
  
  default:  
    //Koda bloks izpilde  
    //ja neviens iepriekš  
    //norādītais case neizpildās  
}
```

Atslēgvārds, kas  
nosaka switch  
darbības beigas

Mainīgā vērtībai ir  
jāatbilst vienai no šīm  
(else - if)

Rezerves gadījums  
(else)

# SWITCH: PIEMĒRS

```
String drink = "coffee";

switch (drink) {
    case "coffee":
        System.out.println("I would go for Java!");
        break;

    case "tea":
        System.out.println("Everything but Lipton");
        break;

    default:
        System.out.println("Ugh.. What?");
}
```



BŪLA

IZTEIKSME

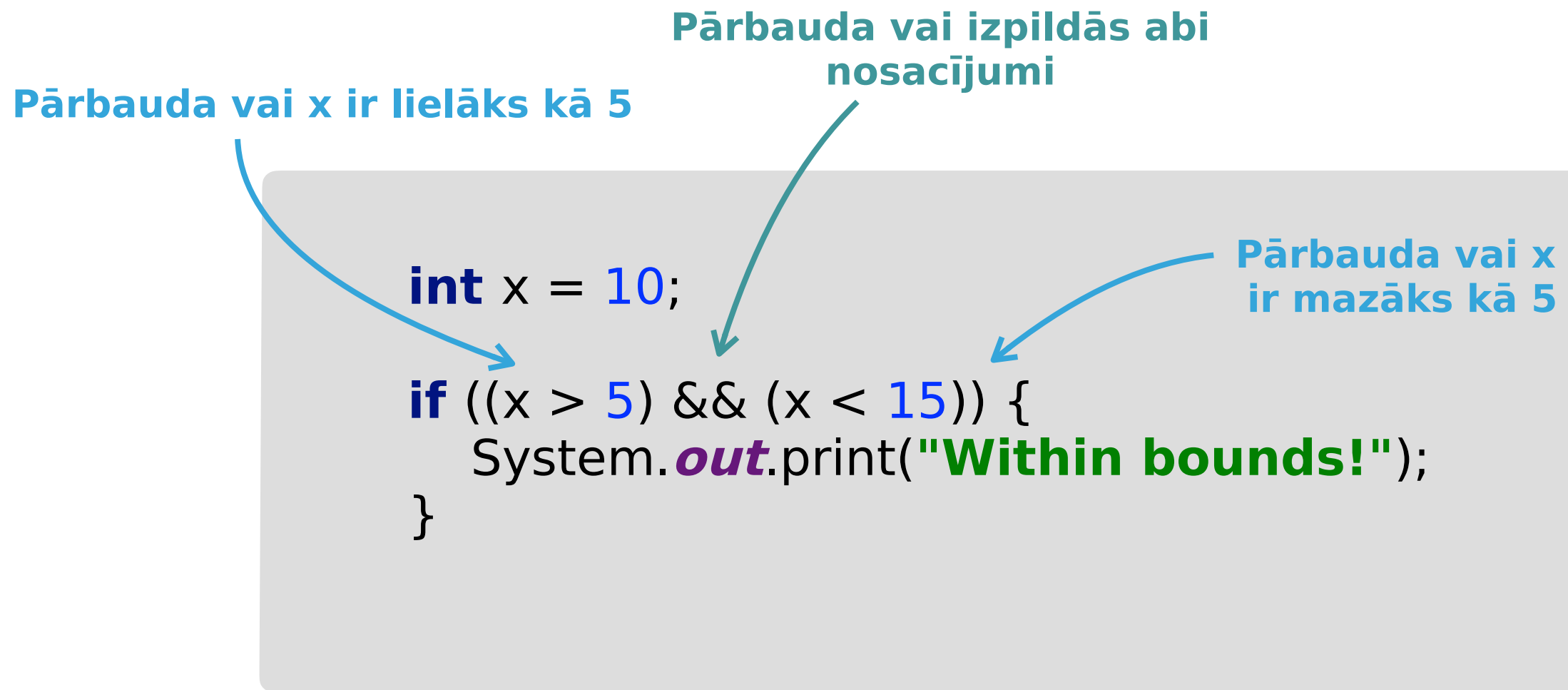
# RELĀCIJU OPERATORI

Operators	Operācija
<code>==</code>	Vienāds ar
<code>!=</code>	Nav vienāds ar
<code>&gt;</code>	Lielāks kā
<code>&gt;=</code>	Lielāks vai vienāds kā
<code>&lt;</code>	Mazāks kā
<code>&lt;=</code>	Mazāks vai vienāds kā

# NOSACĪJUMA OPERATORI

Operators	Operācija
&&	Nosacījuma UN
	Nosacījuma VAI
!	Nosacījuma NE

# BŪLA IZTEIKSME: PIEMĒRS







# KODA TESTĚŠANA

# UZDEVUMS

1. Uzraksti klasi ar metodi, kas aprēķina **lielāko skaitli** no diviem padotajiem skaitļiem
2. Uzraksti testu, kas izsauc uzrakstīto metodi un **pārbaudi**, ka atgrieztais rezultāts ir **pareizs**
3. **Palaid** testu

# 1. UZRAKSTI KLASI UN METODI

```
public class QuickMaths {  
  
    public int max(int a, int b) {  
        if (a > b) {  
            return a;  
        } else {  
            return b;  
        }  
    }  
}
```

## 2.A. UZRAKSTI TESTU

```
public class QuickMathsTest {  
  
    public void test1() {  
  
        QuickMaths victim = new QuickMaths();  
  
        int a = 3;  
        int b = 5;  
  
        int expectedResult = 5;  
  
        int actualResult = victim.max(3, 5);  
  
        check(actualResult, expectedResult, "test1");  
    }  
  
    public void check(int actualResult, int expectedResult, String testName) {  
        if (actualResult == expectedResult) {  
            System.out.println(testName + " has passed!");  
        } else {  
            System.out.println(testName + " has failed!");  
            System.out.println("Expected " + expectedResult + " but was " + actualResult);  
        }  
    }  
}
```

## 2.B. IZVEIDO TEST KLASĒS OBJEKTU UN IZSAUC TESTU

```
public class QuickMathsTest {  
  
    public static void main(String[] args) {  
  
        QuickMathsTest testRunner = new QuickMathsTest();  
        testRunner.test1();  
  
    }  
  
    ...  
  
}
```

### 3. PALAID UN PĀRBAUDI REZULTĀTU

test1 has passed!

Process finished with exit code 0





LATVIJAS  
UNIVERSITĀTE  
ANNO 1919