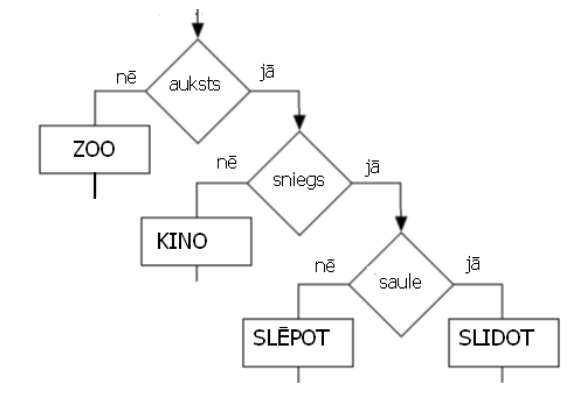
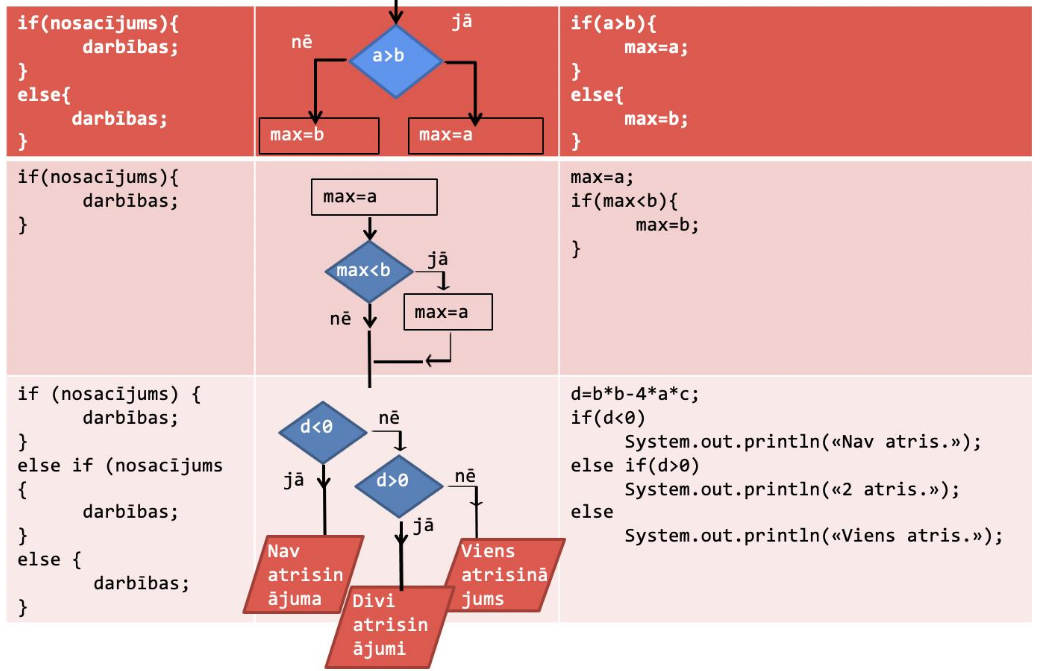
# Sazarojuma konstrukcijas programmēšanas valodā Java

## 1. Teorijas izklāsts

Sazarojuma konstrukcijas ļauj programmā izpildīt dažādus papildus koda daļas **atkarībā no noteiktajiem nosacījumiem** vai tie ir patiesi, vai nav patiesi. (**true** vai **false**)

Ar zarošanu izlemj, kuru **darbību veikt** un ar kompleksu zarošanos izlemj, kura **darbība no vairākām iespējamām jāizpilda**. Zemāk aplūkojama blokshēma, kā tiek izmantoti sazarojumi.

Start(IT) materiāls

Javā sazarojuma konstrukcijas tiek apzīmētas ar maziem burtiem, un izmanto atslēgvārdus - **if**, **else if**, **else**, **switch**. Ja tiek **uzrakstīts ar lieliem burtiem**, **tiks** **rādīta kļūda**.  
Galvenie veidi:  
• if – izpilda vienu koda bloku, ja nosacījums ir patiess.  
• if, else – izpilda koda bloku, ja nosacījums ir patiess, ja nē veic citu darbību.  
• if, else if, else – izmanto papildus nosacījumus vairākām darbībām.  
• switch, case – izvēlas izpildāmo bloku pēc vērtības.

Start(IT) materiāls 2

## 2. Salīdzināšanas izteiksmes un operatori

Sazarojuma komandās drīkst izmantot jebkādu apgalvojumu, par kuru var pateikt, vai tas ir **patiess vai nepatiess.**

Java valodā ir pieejami vairākas **salīdzināšanas izteiksmes**, kā arī **loģiskās izteiksmes**, ko mēs izmantojam saistībā ar mainīgajiem, lai veiktu vairākas darbības ar nosacījumiem, **matemātiskās izteiksmes** arī bieži izmanto!

|  |  |
| --- | --- |
| Salīdzināšanas apzīmējums | Salīdzināšanas nozīme |
| == | Vienāds |
| != | Nav vienāds |
| < | Lielāks |
| <= | Lielāks vai vienāds |
| > | Mazāks |
| >= | Mazāks vai vienāds |

|  |  |
| --- | --- |
| Matemātiska operatora apzīmējums | Operatora nozīme |
| + | Saskaitīt |
| - | Atņemt |
| \* | Reizināt |
| / | Dalīt |
| % | Modulis |
| ++ | Pieskaitīt 1 pie sevis |
| -- | Atņemt 1 pie sevis |

|  |  |
| --- | --- |
| Loģiskā operatora apzīmējums | Operatora nozīme |
| ! | NOT |
| && | AND |
| || | OR |

## 

## 3. Koda fragmenti ar komentāriem un piebildes

**if piemērs:**

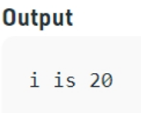
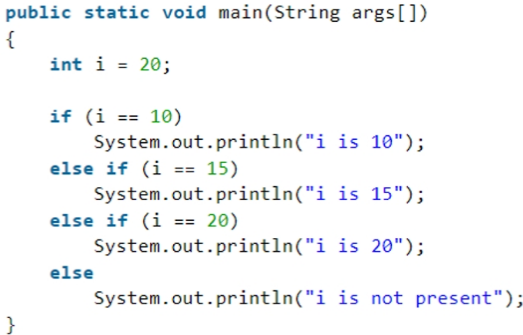
int x = 5; // Mainīgā vērtība  
if (x > 0) { // Uzdod loģiskā operatora nosacījumu  
 System.out.println("x ir pozitīvs!"); // Izvada ja ir patiess, izprintēs!  
}

**if, else piemērs:**

boolean suns = false; // Mainīgā vērtība  
if (suns == true) { // Uzdod salīdzināšanas nosacījumu  
 System.out.println("Tas ir suns!"); // Izvada ja ir patiess  
} else {  
 System.out.println("Tas nav suns!"); //Izvada ja nav patiess, izprintēs!  
}

**if, else if, else piemērs:**

int atzime = 75; // Mainīga vērtība  
if (atzime > 80) { // Pirmais salīdzinašanas nosacījums  
 System.out.println("Izcili!");  
} else if (atzime >= 60) { // Otrais salīdzināšanas nosacījums  
 System.out.println("Labi!"); // Atzime >=60 bet < 80, izprintēs!  
} else { // Pretējais gadījums iepriekšējiem nosacījumiem  
 System.out.println("Neapmierinoši!");  
}



Java valodā **switch** un **case** tiek izmantoti, lai vienkāršotu vairāku izvēles nosacījumus, kad ir nepieciešams salīdzināt vienu mainīgo ar vairākām vērtībām. Tas ir **vieglāk** un **efektīvāk** nekā vairāki if–else bloki.

Case vērtībai ir jābūt **tāda paša mainīgā tipa,** kā switch izteiksmē mainīgā, un case **dublikāti** **nav atļauti**. Tad, kad case nosacījums ir izpildīts tiek pielietots **break**, lai izietu no switch un turpinātu programmas darbību. Break ir nepieciešams, lai novērstu visu nākamo case kodu izpildīšanu pēc kārtas. Switch beigās var pielietot **default** kā noklusējumu, ja neviens no case neizpildās, bet nav obligāti.

**switch, case piemērs:**

int diena = 4; // Mainīgā vērtība  
switch (day) { // Pielieto switch priekš dienas pārbaudes  
 case 1: // Pārbaude vai diena == 1  
 System.out.println("Pirmdiena!");  
 break;  
 case 2: // Pārbaude vai diena == 2  
 System.out.println("Otrdiena!");  
 break;

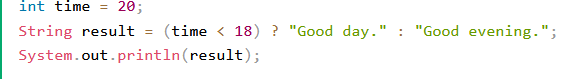
case 3: // Pārbaude vai diena == 3  
 System.out.println("Trešdiena!");  
 break;

case 4: // Pārbaude vai diena == 4  
 System.out.println("Ceturtdiena!"); // Diena ir 4!  
 break;  
 default: // Ja neatbilst neviena vērtība, aiziet uz default  
 System.out.println("Diena neatbilst nevienam case!");  
}

Pastāv arī saīsinātais operators "if else" vietā, kas pazīstams kā **trīskāršais operators**, jo tas sastāv no **trīs operandiem**. To var izmantot, lai aizstātu koda rindiņas ar vienu rindiņu, un to visbiežāk izmanto, lai aizstātu vienkāršus "if else" paziņojumus:



W3Schools materiāls 1



W3Schools materiāls 2

## 4. Programmēšanas uzdevumu piemēri

### Uzdevums 1: Pāra vai nepāra skaitlis

Uzrakstīt programmu, kas nosaka vai ievadītais veselais skaitlis ir pāra vai nepāra.

Darba gaita:  
1. Iegūt skaitli n.  
2. Ja n ir bez atlika, izvada "Pāra skaitlis".  
3. Pretēji izvada "Nepāra skaitlis".

Scanner scan = new Scanner(System.in); // Izmanto skenera bibliotēku ievadei  
System.out.println"Ievadi skaitli: ");  
int n = scan.nextInt(); // Pieškir ievadīto vērtību mainīgajam  
if (n % 2 == 0) { // Pārbaude ar matemātisko atlikuma operatoru  
 System.out.println("Pāra skaitlis"); // Ja n dalās ar 2 bez atlikuma  
} else {  
 System.out.println("Nepāra skaitlis"); // Ja n daloties ar 2 ir atlikums  
}

### Uzdevums 2: Atzīmes iedalījums

Uzrakstīt programmu, kas iedala studentu atzīmi (0-100) vairākās kategorijās: 90+ Izcili, 75-89 Labi, 50-74 Viduvēji, <50 Neapmierinoši.

int vertejums = scan.nextInt(); // Pieškir ievadīto vērtību mainīgajam  
if (vertejums >= 90) { // Pārbaude vai vērtejums ir 90 vai lielāks  
 System.out.println("Izcili!");  
} else if (vertejums >= 75) { // Pārbaude vai vērtejums ir 75 vai lielāks  
 System.out.println("Labi!");  
} else if (vertejums >= 50) { // Pārbaude vai vērtejums ir 50 vai lielāks  
 System.out.println("Viduvēji!");  
} else { // Prētējā gadījumā  
 System.out.println("Neapmierinoši!");  
}

### Uzdevums 3: Vadības izvēlne

Izveidot programmu izmantojot switch, kas pēc izvēles izvada konkrētus paziņojumus:  
1 - Sākt, 2 - Apturēt, 3 - Turpināt, 4 - Iziet.

System.out.print("Izvēlies darbību (1-4): ");  
int izvele = scan.nextInt(); // Pieškir ievadīto vērtību mainīgajam  
switch(izvele) { // Ar switch pārbauda izvēles vērtību   
 case 1: // Ja izvele == 1  
 System.out.println("Programma startē");  
 break;  
 case 2: // Ja izvele == 2  
 System.out.println("Programma apturēta");  
 break;  
 case 3: // Ja izvele == 3  
 System.out.println("Programma turpinās");  
 break;  
 case 4: // Ja izvele == 4  
 System.out.println("Programma beidz darbu");  
 break;  
 default: // Ja izvele nav neviena case   
 System.out.println("Nederīga izvēle");  
}

## Atsauces

- Start(IT): <https://kursi.startit.lv/system/lecture_literatures/files/000/000/011/original/7._Sazarojumi_un_nosacījumi_(JAVA).pdf?1585579161>

- W3Schools Java Conditions: <https://www.w3schools.com/java/java_conditions.asp>

-W3Schools Java Switch: <https://www.w3schools.com/java/java_switch.asp>

- GeeksforGeeks Java If-else Statement: <https://www.geeksforgeeks.org/java-if-else-statement-with-examples/>

- GeeksforGeeks Java Switch statement: <https://www.geeksforgeeks.org/switch-statement-in-java/>

- Programiz Java: <https://www.programiz.com/java-programming/if-else-statement>

-ChatGPT: <https://chatgpt.com>