

Pętle i instrukcje warunkowe

Lekcja 2

Krzysztof Gębicz

Instrukcje warunkowe: „if”

- Służą do podejmowania decyzji.
- Składnia:

```
if (warunek) {
```

```
    // kod do wykonania, jeśli warunek jest  
    prawdziwy
```

```
}
```

Instrukcje warunkowe: „if”

```
int liczba = 10;
```

```
if (liczba > 5) {
```

```
    System.out.println("Liczba jest większa od 5");
```

```
}
```

„if – else”

- Co jeśli warunek nie jest spełniony?
- Używamy **else**.

```
int liczba = 3;

if (liczba > 5) {

    System.out.println("Liczba
jest większa od 5");

}

else {

    System.out.println("Liczba
jest mniejsza od 5");

}
```

„else if”

- Możemy sprawdzić kilka warunków po kolei:

```
int ocena = 4;

if (ocena == 5) {
    System.out.println("Ocena bardzo dobra");
}

else if (ocena == 4) {
    System.out.println("Ocena dobra");
}

else {
    System.out.println("Inna ocena");
}
```

Instrukcja „switch”

- Wygodny sposób, gdy mamy wiele możliwych przypadków.

```
int dzien = 3;

switch (dzien) {

    case 1:

        System.out.println("Poniedziałek");

        break;

    case 2:

        System.out.println("Wtorek");

        break;

    default:

        System.out.println("Inny dzień");

}
```

Pętle

- Pętla = powtarzanie czynności.
- W Javie są trzy główne rodzaje:
 1. for – kiedy wiemy, ile razy powtórzyć.
 2. while – dopóki warunek jest prawdziwy.
 3. do...while – przynajmniej raz, potem sprawdza warunek.

Pętla „for”

Program wypisze liczby od 0 do 4.

- `int i = 0` – początek
- `i < 5` – warunek
- `i++` – co zrobić po każdym obrocie

```
for (int i = 0; i < 5; i++) {  
    System.out.println("i = " + i);  
}
```


Pętla „while”

- Wykonuje kod, dopóki warunek jest prawdziwy.

```
int i = 0;
```

```
while (i < 5) {
```

```
    System.out.println("i = " + i);
```

```
    i++;
```

```
}
```

Pętla „do...while”

- Kod wykona się przynajmniej raz, nawet jeśli warunek nie jest spełniony.

```
int i = 0;
```

```
do {
```

```
    System.out.println("i = " + i);
```

```
    i++;
```

```
} while (i < 5);
```

Wczytywanie danych od użytkownika (Scanner)

- W Javie można pobierać dane od użytkownika z klawiatury.
- Służy do tego klasa Scanner.
- Aby jej używać, musimy ją zaimportować
 - Biblioteka:
 - `import java.util.Scanner`
 - Tworzenie obiektu:
 - `Scanner sc = new Scanner(System.in);`

Wczytywanie liczby przykład kod

```
import java.util.Scanner;

public class Main {

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);

        System.out.print("Podaj liczbę: ");

        int liczba = scanner.nextInt();

        System.out.println("Wpisałeś: " + liczba);

    }

}
```

Zadania Utrwalające

Zadanie 1 Sprawdzenie parzystości

- Poprosi użytkownika o podanie liczby.
- Sprawdzi, czy liczba jest parzysta czy nieparzysta.

Zadanie 2 Liczby od 1 do N

- Poprosi o liczbę n .
- Wypisze wszystkie liczby od 1 do n w jednej linii.

Zadanie 3 Suma liczb

- Poprosi użytkownika o podanie liczby n .
- Obliczy sumę wszystkich liczb od 1 do n .

Zadanie 4 Tabliczka mnożenia

- Poprosi użytkownika o podanie liczby.
- Wypisze tabliczkę mnożenia tej liczby od 1 do 10.

Zadanie 5 Zgadnij liczbę

- Wylosuje liczbę z zakresu 1–10 (`int los = (int)(Math.random()*10 + 1);`).
- Poprosi użytkownika o zgadnięcie liczby.
- Wyświetli informację, czy użytkownik trafił, czy nie.