

# Język programowania JAVA

© 2011-12 Radosław Klimek



Vincent Van GOGH: *Mężczyzna pijący filiżankę kawy*

## Zadanie. Wątki

- zad.1 Napisać program dwuwątkowy, z których jeden wypisuje na ekran '1' a drugi '0'. Wątki zaimplementować przy użyciu dziedziczenia po klasie Thread
- zad.2 Wątki z zad.1 zaimplementować przy użyciu interfejsu Runnable
- zad.3 Wprowadzić synchronizację do programu z zad.2.

zad.1 Napisać program dwuwątkowy, z których jeden wypisuje na ekran '1' a drugi '0'. Wątki zaimplementować przy użyciu dziedziczenia po klasie Thread

```
1 public class Uruchom {
2
3     public class Watek1 extends Thread {
4         public void run() {
5             for(int i = 0; i < 100000; i++){
6                 System.out.println("1");
7             }
8         }
9     }
10
11    public class Watek2 extends Thread {
12        public void run() {
13            for(int i = 0; i < 100000; i++){
14                System.out.println("0");
15            }
16        }
17    }
18    public static void main( String [] args ) {
19        Watek1 w1 = new Watek1();
20        Watek2 w2 = new Watek2();
21
22        w1.start();
23        w2.start();
24    }
25 }
```

## zadanie 1 - przykładowe rozwiązanie (1/2)

```
1 public class MyThread extends Thread
2 {
3     private int whichThread;
4     private int delay;
5     public My thread(int whichThread, int delay)
6     {
7         supper();
8         this.delay = delay;
9         this.whichThread = whichThread;
10    }
11    public void run()
12    {
13        switch(whichThread)
14        {
15            case 1: thread1(); break;
16            case 2: thread2(); break;
17        }
18    }
19    public static void main(String args[])
20    {
21        new MyThread1(1, 1).start();
22        new MyThread2(2, 2).start();
23    }
```

## zadanie 7 - przykładowe rozwiązanie (2/2)

```
1  public void thread1()  
2  {  
3      for (i = 0; i < 10; i++){  
4          try{  
5              sleep(delay);  
6          }  
7          catch (InterruptedException e){  
8              }  
9          System.out.println("1");  
10     }  
11 }  
12 public void thread2()  
13 {  
14     for (i = 0; i < 10; i++){  
15         try{  
16             sleep(delay);  
17         }  
18         catch (InterruptedException e){  
19             }  
20         System.out.println("0");  
21     }  
22 }  
23 }
```

## zad.2 Wątki z zad.1 zaimplementować przy użyciu interfejsu Runnable

```
1 public class Uruchom {
2
3     public class Watek1 implements Runnable {
4         public void run() {
5             for(int i = 0; i < 100000; i++){
6                 System.out.println("1");
7             }
8         }
9     }
10
11     public class Watek2 implements Runnable {
12
13         public void run() {
14             for(int i = 0; i < 100000; i++){
15                 System.out.println("0");
16             }
17         }
18     }
19     public static void main( String [] args ) {
20         Watek1 w1 = new Watek1();
21         Watek2 w2 = new Watek2();
22
23         (new Thread(w1)).start();
24         (new Thread(w2)).start();
25     }
26 }
```

### zad.3 Wprowadzić synchronizację do programu z zad.2.

```
1 public class Uruchom {
2     public static void main( String[] args ) {
3         Watek1 w1 = new Watek1();
4         Watek2 w2 = new Watek2();
5
6         (new Thread(w1)).start();
7         w2.start();
8     }
9
10    public static synchronized void wypisz( String tekst){
11        for(int i = 0; i<10000; i++)
12            System.out.println(tekst);
13    }
14
15    public class Watek1
16        implements Runnable {
17        public void run() {
18            Uruchom.wypisz("1");
19        }
20    }
21
22    public class Watek2
23        extends Thread {
24        public void run() {
25            Uruchom.wypisz("0");
26        }
27    }
```

zadanie 1. Dana jest tablica jednowymiarowa n-elementowa elementów typu `int`. Napisać program, który wyświetli element o indeksie podanym przez użytkownika. Program musi zawierać wyjątek obsługujący sytuację kiedy użytkownik poda niepoprawny indeks. (spoza zakresu) . Wyjątek ma być zgłaszany przy pomocy bloku `try...catch`



```
1 import java.util.Scanner;
2
3 public class Odczyt{
4     public static void main(String [] args){
5         int tab [] = {1,2,3,4,5};
6         Scanner odczyt = new Scanner(System.in);
7         int index = -1;
8
9         System.out.println("Podaj indeks tablicy , który chcesz zobaczyć :");
10        index = odczyt.nextInt();
11
12        try {
13            System.out.println(tab[index]);
14        } catch (ArrayIndexOutOfBoundsException e) {
15            System.out.println("Niepoprawny parametr , rozmiar tablicy to :");
16        }
17    }
18 }
```

Zadanie.2 Napisać program , który oblicza wynik dzielenia dwóch liczb podanych przez użytkownika. Program musi obsługiwać wyjątek, który obsługuje sytuację kiedy druga liczba przez którą dzielimy jest równa 0. Wyjątek ma być generowany przy pomocy instrukcji **throw** .

```
1 import java.util.Scanner;
2
3 public class Main {
4     public static void main(String[] args) throws ArithmeticException
5     {
6         double liczba_1;
7         double liczba_2;
8         double iloraz;
9         Scanner liczba = new Scanner(System.in);
10        System.out.print("Podaj pierwszą liczbę : ");
11        liczba_1 = liczba.nextDouble();
12        System.out.print("Podaj drugą liczbę : ");
13        liczba_2 = liczba.nextDouble();
14        if (liczba_2 == 0)
15            throw new ArithmeticException("Nie można dzielić przez 0");
16        else
17        {
18            iloraz = liczba_1/liczba_2;
19            System.out.println("wynik dzielenia : "+ iloraz);
20        }
21    }
22 }
```

zadanie.3 Przerobić program z zadania.2 tak aby jeżeli zostanie wygenerowany wyjątek `ArithmeticException` użytkownik został poproszony o poprawne dane. (Pętla **while**).

```

1 import java.util.Scanner;
2
3 public class Main {
4     public static void main(String[] args) throws ArithmeticException
5     {
6         double liczba_1;
7         double liczba_2;
8         double iloraz;
9         Scanner liczba = new Scanner(System.in);
10        System.out.print("Podaj pierwszą liczbę: ");
11        liczba_1 = liczba.nextDouble();
12        System.out.print("Podaj drugą liczbę: ");
13        liczba_2 = liczba.nextDouble();
14        boolean Poprawne_Dane = false;
15
16        while (!Poprawne_Dane) {
17            if (liczba_2 == 0)
18            {
19                throw new ArithmeticException("Nie można dzielić przez 0");
20                System.out.println("Niepoprawne dane! +
21                                "podaj dzielnik:");
22                liczba_2 = liczba.nextDouble();
23
24            }
25            else
26            {
27                {
28                    iloraz = liczba_1/liczba_2;
29                    System.out.println("wynik dzielenia: "+ iloraz);
30                }
31                Poprawne_dane = liczba_2 == 0? false : true;
32            }
33        }
34    }
35 }

```