



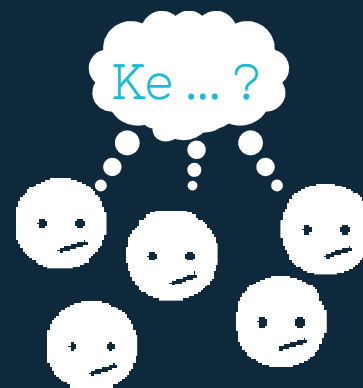
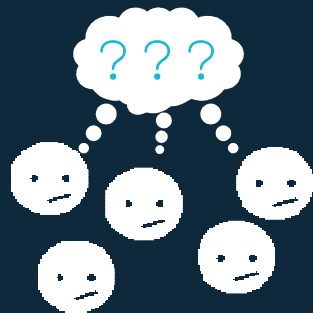
JavaCroatia

Funkcijsko programiranje u Javi 8

Tomislav Biško

INFODOM

Što je funkcijsko programiranje?



Krenimo redom ...



Funkcijsko ...


```
public int minFunction (int n1, int n2) {  
    int min;  
    if (n1 > n2)  
        min = n2;  
    else  
        min = n1;  
  
    return min;  
}
```

Funkcijsko, ali ...

Funkcijsko programiranje je programiranje s funkcijama.

Onim matematičkim ...

$$f(x) = x^2 + 1$$



Što je funkcija (u matematici)?

$$f(x) = x^2$$

za 2 dobijemo 4

za 3 dobijemo 9

za 4 dobijemo 16

za 2 opet dobijemo 4

itd ...



Referencijalna prozirnost

(eng. referential transparency)

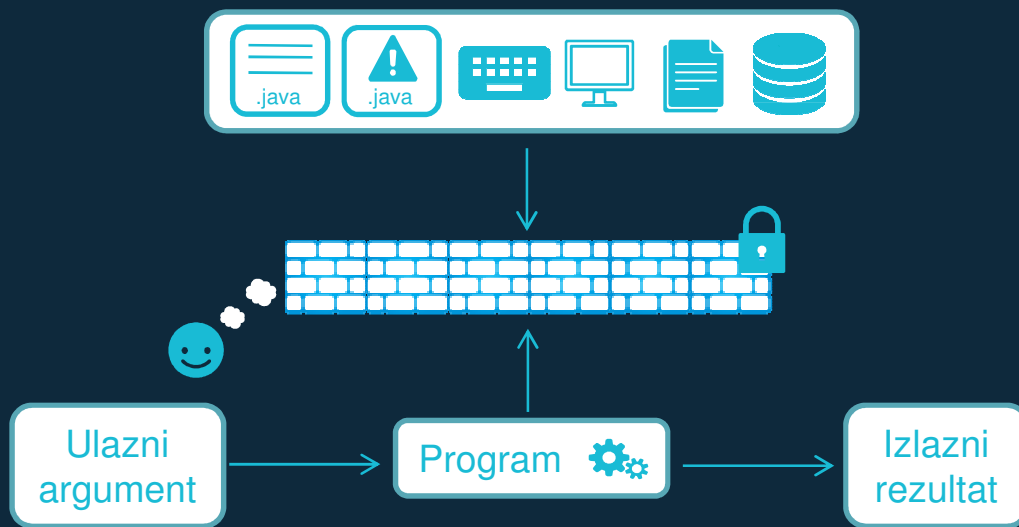


Potpis funkcije

... da joj je poslovanje transparentno ...

```
<povratni tip> <naziv> (<argumenti>) {  
    ...  
}
```


Transparentno poslovanje programskog kôda



Bez popratnih efekata



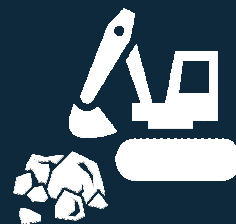
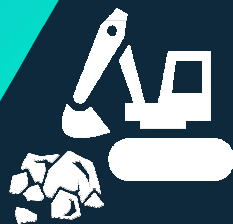
(eng. no side-effects)



Ali, kakva je korist od programa bez popratnih vrijednosti?

- ◇ lakše testiranje
- ◇ prostor za optimizaciju
- ◇ i ...

Paralelno izvršavanje



(eng. Concurrent programming)





Građani prvog reda



Dajmo građanska prava
našim funkcijama!

(eng. first-class citizens)




Što imamo u Javi?

- ◇ Lambda izraze λ
- ◇ `java.util.function` paket
- ◇ Stream API



Lambda izrazi u Javi



- ◇ „anonimne” funkcije
- ◇ first class functions na Java način
- ◇ syntactic sugar 



Kako je to bilo do sada

```
package java.awt.event;  
import java.util.EventListener;
```

```
public interface ActionListener extends EventListener {  
    public void actionPerformed(ActionEvent e);  
}
```

```
JButton testButton = new JButton("Test Button");
```

```
testButton.addActionListener (  
    new ActionListener() {  
        @Override public void actionPerformed(ActionEvent ae) {  
            System.out.println("Click Detected by Anon Class");  
        }  
    }  
);
```



A kako su nam zasladili život u Javi 8

(Lista argumenata) -> { Tijelo funkcije }

```
(int x, int y) -> x + y ili () -> 42
```

```
(String s) -> {  
    s = s.toUpperCase();  
    System.out.println(s);  
}
```

```
Runnable r2 = () -> System.out.println("Hello world!");
```



Funkcijska sučelja

- ◇ `java.util.function`
- ◇ Funkcijsko sučelje ima samo jednu apstraktnu metodu
- ◇ Bilo koje sučelje s jednom apstraktnom metodom je funkcijsko (eng. `SingleAbstractMethod`)





```
java.util.function
```

```
public interface Predicate<T> {
```

```
    public boolean test (T t);
```

```
}
```

Meet SAM ;)



Predicate<T> i filter()

`Stream<T> filter(Predicate<? super T> predicate)`

```
int sum = widgets.stream()
    .filter(w -> w.getColor() == RED)
    .mapToInt(w -> w.getWeight())
    .sum();
```



Stream API

- ◇ `java.util.stream`
- ◇ želimo reći što, a ne kako
- ◇ za rad na kolekcijama
- ◇ jednostavna paralelizacija



Stream API

Sadrži funkcije karakteristične za funkcijsko programiranje:

- ◇ map
- ◇ reduce
- ◇ sorted
- ◇ forEach
- ◇ ...



Stream API

```
List<String> cars = Arrays.asList("Audi", "BMW", " Opel", " Fiat", "Ferrari");
```



```
cars.stream()  
    .filter(s -> s.startsWith("F"))  
    .map(String::toUpperCase)  
    .sorted()  
    .forEach(System.out::println);
```




Stream API u paraleli

```
IntStream randomIntegers = (new Random()).ints().limit(500);
```

```
randomIntegers.parallel()  
    .sorted((x, y) -> x - y)  
    .limit(10)  
    .toArray();
```





„Nije programski jezik ono što čini programiranje funkcijskim. –Već način na koji pišemo naše programe.”

Pierre-Yves Saumont



Hvala!

Pitanja?

Kontakt:

◇ tomislav.bisko@infodom.hr





Reference i materijali

- ◇ Java 8 in Action, Raoul-Gabriel Urma, Mario Fusco, and Alan Mycroft
- ◇ Java SE 8 for the really impatient, Cay S. Horstmann
- ◇ Functional Programming in Java, Pierre-Yves Saumont

