

# Novinky v Javě 8

Ing. František Kučera

20. 11. 2015

# Předchozí verze

- září 2004: Java 5
- prosinec 2006: Java 6
- červenec 2011: Java 7

# Vydání Javy 8

- duben 2012: M1
- červen 2013: M7 – Feature Complete
- září 2013: M8 – Developer Preview
- leden 2014: M9 – Final Release Candidate
- březen 2014: GA – General Availability
- duben 2015: konec podpory Javy 7

# Novinky ve verzi 8

- funkcionální programování, lambda výrazy
- Nashorn – JavaScript (JSR-223)
- Date and Time API
- opakující se anotace
- PermGen → Metaspace

# Funkční rozhraní

- @FunctionalInterface
- právě jedna abstraktní metoda
- výchozí metody rozhraní
- Supplier, Function, BiFunction, Predicate

# Posluchače událostí – 7

```
JButton tlačítko = new JButton();

tlačítko.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {
        System.out.println("Tlačítko: " + e);
    }
});
```

# Posluchače událostí – 8

```
JButton tlačítko = new JButton();  
  
tlačítko.addActionListener(  
    e -> System.out.println("Tlačítko: " + e));
```

# Coalesce – 7

```
public static <T> T coalesce(T... hodnoty) {  
    for (T hodnota : hodnoty) {  
        if (hodnota != null) {  
            return hodnota;  
        }  
    }  
    return null;  
}
```



# Coalesce – 8

```
public static <T> T coalesce(Supplier<T>... s) {
    for (Supplier<T> d : s) {
        if (d != null) {
            T hodnota = d.get(); // vyhodnocení
            if (hodnota != null) {
                return hodnota;
            }
        }
    }
    return null;
}
```

# Funkce jako odkaz na metodu

- instanční i statické
- `System::currentTimeMillis`

# Stream API

```
Collection<String> písmena = Arrays.asList(  
    new String[]{"a", "b", "c"});
```

```
Stream<String> proud = písmena.stream();  
proud.forEach(p -> System.out.println(p));  
// p je typu String
```

```
// odkazem na metodu:  
proud.forEach(System.out::println);
```

# Mapování

```
Collection<String> pozdravy = Arrays.asList(  
    new String[]{ "ahoj", "nazdar", "dobrý den"});  
  
pozdravy.stream()  
    .map(s -> s.length())  
    .forEach(i -> System.out.println("délka: " + i));
```

# Filtrování

```
Collection<Integer> n = Arrays.asList(  
    new Integer[]{0, 1, 2, 3, 4, 5, 6, 7, 8, 9});  
n.stream()  
    .filter(i -> i % 2 == 0)  
    .forEach(i -> System.out.println("sudé: " + i));
```

# Map and reduce

```
int součet = n.stream().reduce((a, b) -> a + b).get
```

```
int d = pozdravy.stream().collect(
    Collectors.summingInt((String s) -> s.length()));
System.out.println("Celková délka pozdravů: " + d);
```

# Limit

```
Random r = new Random();  
r.ints()  
    .limit(10)  
    .forEach(i -> System.out.println("náhodné: " + i))
```

# Odkazy a zdroje

- <https://blog.frantovo.cz/c/308/Java%20%3A%20novinky%20jazyka>
- <http://openjdk.java.net/projects/jdk8/features>
- <http://winterbe.com/posts/2014/03/16/java-8-tutorial/>
- <http://www.javaworld.com/article/2092260/java-se/java-programming-with-lambda-expressions.html>