

ПЛАТФОРМЕНО-НЕЗАВИСИМИ ПРОГРАМНИ ЕЗИЦИ

Лекция 11
КОЛЕКЦИИ
/продължение/

проф. дн И. Атанасов

СЪДЪРЖАНИЕ

- ЛАМБДА ИЗРАЗИ
- ПОНЯТИЕ ЗА ПОТОЦИ
- ВИДОВЕ ПОТОЦИ

ЛАМБДА ИЗРАЗИ

Понятие за ламбда изрази чрез пример.
Какво е неудобството?

```
5 public class Person {  
6     public enum Sex { MALE, FEMALE }  
7  
8     String name;  
9     Sex    gender;  
10  
11     public int getAge() {  
12         // ...  
13     }  
14  
15     public void printPerson() {  
16         // ...  
17     }  
18 }
```

```
public static void printOlderThan(  
    List<Person> roster, int age  
) {  
    for (Person p : roster) {  
        if (p.getAge() >= age) {  
            p.printPerson();  
        }  
    }  
}
```

ЛАМБДА ИЗРАЗИ

Понятие за ламбда изрази чрез пример.
Какво е неудобството?

```
interface CheckPerson {  
    boolean test(Person p);  
}
```

```
public static void printPersons(  
    List<Person> roster,  
    CheckPerson tester  
    )  
{  
    // ....  
}
```

```
printPersons(  
    roster,  
    new CheckPerson() {  
        public boolean test(Person p) {  
            return p.getGender() == Person.Sex.MALE  
                && p.getAge() >= 18  
                && p.getAge() <= 25;  
        }  
    }  
);
```

ЛАМБДА ИЗРАЗИ

Понятие за ламбда изрази чрез пример.
Какво е неудобството?

```
interface CheckPerson {  
    boolean test(Person p);  
}
```

```
public static void printPersons(  
    List<Person> roster,  
    CheckPerson tester  
)  
{  
    // ....  
}
```

```
printPersons(  
    roster,  
    (Person p) -> p.getGender() == Person.Sex.MALE  
        && p.getAge() >= 18  
        && p.getAge() <= 25  
);
```

ЛАМБДА ИЗРАЗИ

Понятие за ламбда изрази чрез пример.
Синтаксис

```
p -> p.getGender() == Person.Sex.MALE  
    && p.getAge() >= 18  
    && p.getAge() <= 25
```

```
p -> {  
    return p.getGender() == Person.Sex.MALE  
    && p.getAge() >= 18  
    && p.getAge() <= 25;  
}
```

ЛАМБДА ИЗРАЗИ

Понятие за ламбда изрази чрез пример.
Пример за използване

```
public class Calculator {  
    interface IntegerMath {  
        int operation(int a, int b);  
    }  
  
    public int operateBinary(int a, int b, IntegerMath op) {  
        return op.operation(a, b);  
    }  
}
```

ЛАМБДА ИЗРАЗИ

Понятие за ламбда изрази чрез пример.
Пример за използване

```
interface IntegerMath {  
    int operation(int a, int b);  
}
```

```
Calculator myApp = new Calculator();  
IntegerMath addition = (a, b) -> a + b;  
IntegerMath subtraction = (a, b) -> a - b;  
  
System.out.println("40 + 2 = " +  
    myApp.operateBinary(40, 2, addition));  
  
System.out.println("20 - 10 = " +  
    myApp.operateBinary(20, 10, subtraction));
```


ЛАМБДА ИЗРАЗИ

Понятие за ламбда изрази чрез пример.
Пример за използване в GUI

```
rdbtnStone.addItemListener(  
    new ItemListener() {  
        public void itemStateChanged(ItemEvent arg0) {  
            btnDone.setEnabled(true);  
        }  
    }  
);
```

```
rdbtnStone.addItemListener(  
    (e) -> {  
        btnDone.setEnabled(true);  
    }  
);
```

ПОТОЦИ

Понятие за потоци чрез пример

```
for (Person p : roster) {  
    System.out.println(p.getName());  
}
```

```
roster.stream()  
    .forEach(e -> System.out.println(e.getName()));
```

ПОТОЦИ

java.util.stream

Interface Stream<T>

Type Parameters:

T - the type of the stream elements

All Superinterfaces:

AutoCloseable, BaseStream<T,Stream<T>>

```
public interface Stream<T>  
extends BaseStream<T,Stream<T>>
```

Понятие за потоци

ПОТОЦИ

Понятие за потоци чрез пример.
Филтриране и агрегация

```
int sum = 0;
for (Person p : roster) {
    if( p.getAge() > 50 )
        sum += p.getAge();
}
System.out.println( sum );
```

```
roster.stream()
    .filter(e -> e.getAge() > 50)
    .mapToInt( e -> e.getAge() )
    .sum();
```

ПОТОЦИ

Понятие за потоци чрез пример.
Филтриране и агрегация

```
double average = roster
    .stream()
    .filter(p -> p.getGender() == Person.Sex.MALE)
    .mapToInt(Person::getAge)
    .average()
    .getAsDouble();
```

ПОТОЦИ

java.util.stream

Interface IntStream

All Superinterfaces:

AutoCloseable, BaseStream<Integer,IntStream>

java.util.stream

Interface DoubleStream

All Superinterfaces:

AutoCloseable, BaseStream<Double,DoubleStream>

Видове потоци

java.util.stream

Interface LongStream

All Superinterfaces:

AutoCloseable, BaseStream<Long,LongStream>

ПОТОЦИ

Видове потоци-
пример

```
List<Double> list = DoubleStream.generate( Math::random )  
                                .limit(100)  
                                .boxed()  
                                .toList();
```

ВЪПРОСИ