ПЛАТФОРМЕНО-НЕЗАВИСИМИ ПРОГРАМНИ ЕЗИЦИ

Лекция 10

КОЛЕКЦИИ

проф. дн И. Атанасов

СЪДЪРЖАНИЕ

- ПОНЯТИЕ ЗА ОБОБЩЕНИ
  - КЛАСОВЕ И
  - ИНТЕРФЕЙСИ
- ОСНОВНИ КОЛЕКЦИИ

КОЛЕКЦИИ

Понятие за обобщени интерфейси с пример

```
public interface Comparable<T> {

        public int compareTo( T o );

}
```

КОЛЕКЦИИ

Интерфейс Collection:

- Основни методи

java.util

Interface Collection<E>

```
public abstract boolean add(Object);
public abstract boolean addAll(Collection);
public abstract boolean remove(Object);
public default  boolean removeIf(Predicate);
public abstract boolean removeAll(Collection);
public abstract boolean retainAll(Collection)
public abstract boolean containsAll(Collection)
public abstract void    clear();
```

КОЛЕКЦИИ

Интерфейс Collection:

- Основни методи (продължение)

java.util

Interface Collection<E>

```
public abstract boolean equals(Object);
public abstract int     hashCode();
public abstract boolean contains(Object);
public abstract boolean isEmpty();
public abstract int     size();

public abstract Object[] toArray(Object[]);
public default  Object[] toArray(IntFunction);
public abstract Object[] toArray();
public abstract Iterator iterator();
```

КОЛЕКЦИИ

Интерфейс List:

- Основни методи

java.util

Interface List<E>

```
public abstract void   add(int,Object);
public abstract Object remove(int);
public abstract Object get(int);
public static   List   copyOf(Collection);
public abstract int    indexOf(Object);
public abstract int    lastIndexOf(Object);
public default  void   replaceAll(UnaryOperator);
public abstract List   subList(int,int);
```

КОЛЕКЦИИ

Интерфейс List:

- Основни методи (продължение)

java.util

Interface List<E>

```
public abstract boolean addAll(int,Collection);
public abstract Object  set(int,Object);
public default  void    sort(Comparator);

public abstract ListIterator listIterator(int);
public abstract ListIterator listIterator();
```

КОЛЕКЦИИ

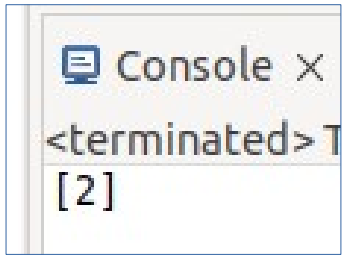
Интерфейс List:

- пример

java.util

Interface List<E>

```
26          List<Integer> l1 = new ArrayList<Integer>(
27                  List.of( 1, 2, 3, 4 )
28                  );
29          List<Integer> l2 = List.of(2, 5);
30          l1.retainAll(l2);
31          System.out.println( l1 );
```



КОЛЕКЦИИ

Интерфейс Set:

java.util

Interface Set<E>

- Реализиращи класове

  AbstractSet, ConcurrentSkipListSet, CopyOnWriteArraySet,

  EnumSet, HashSet, JobStateReasons, LinkedHashSet, TreeSet

- методи

```
public static Set copyOf(Collection)
```

КОЛЕКЦИИ

Интерфейс Set:

- пример

java.util

Interface Set<E>

```
34          Set<Integer> s = new TreeSet<>();
35          s.addAll( List.of( 3, 2, 4, 1 ) );
36          System.out.println( s );
```

Console ×

<terminated>T

[1, 2, 3, 4]

КОЛЕКЦИИ

Интерфейс Map:

- Основни методи

java.util

Interface Map<K,V>

```
public abstract Object get(Object);
public default  Object getOrDefault(Object,Object)
public abstract Object put(Object,Object);
public default  Object putIfAbsent(Object,Object);
public abstract void   putAll(Map);
public default  boolean remove(Object,Object);
public abstract boolean containsKey(Object);
public abstract boolean containsValue(Object);
public abstract Set     keySet();
public abstract Set     entrySet();
public abstract Collection values();
```

КОЛЕКЦИИ

Интерфейс Map:

java.util

Interface Map<K,V>

- Основни методи (продължение)

```
public static Map       copyOf(Map);
public default Object   replace(Object,Object);
public default boolean  replace(Object,Object,Object);
public default void     replaceAll(BiFunction);
public default Object   merge(Object,Object,BiFunction);
public default Object   compute(Object,BiFunction);
public default Object   computeIfAbsent(Object,Function);
public default Object   computeIfPresent(Object,BiFunction);
public default void     forEach(BiConsumer);
public static Map$Entry entry(Object,Object);
public static Map       ofEntries(Map$Entry[]);
```

КОЛЕКЦИИ

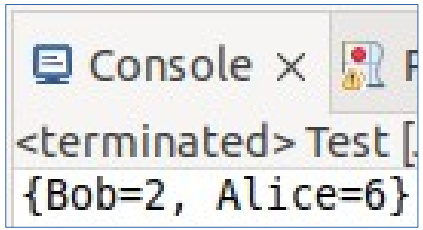
Интерфейс Map:

- пример

java.util

Interface Map<K,V>

```
39          Map<String,Integer> m = new HashMap<>();
40          m.putIfAbsent("Alice", 6);
41          m.putIfAbsent("Bob", 2);
42          System.out.println( m );
```



КОЛЕКЦИИ

Интерфейс Queue:

- Реализиращи класове

  AbstractQueue, ArrayBlockingQueue, ArrayDeque, ConcurrentLinkedDeque, ConcurrentLinkedQueue, DelayQueue, LinkedBlockingDeque, LinkedBlockingQueue, LinkedList, LinkedTransferQueue, PriorityBlockingQueue, PriorityQueue, SynchronousQueue

java.util

Interface Queue<E>

КОЛЕКЦИИ

Интерфейс Queue:

- Основни методи

java.util

Interface Queue<E>

```
public abstract boolean add(Object);
public abstract Object  remove();
public abstract Object  poll();
public abstract Object  peek();
public abstract Object  element();
public abstract boolean offer(Object);
```

КОЛЕКЦИИ

Интерфейс Queue:

- пример

java.util

Interface Queue<E>

```
45          Queue<Integer> q = new LinkedList<>();
46          q.addAll( List.of( 3, 2, 4, 1 ) );
47          System.out.println( "Queue: " + q );
48          int e = q.remove();
49          System.out.println( "Remove " + e + ". Queue: " + q );
50          e = q.peek();
51          System.out.println( "Peek   " + e + ". Queue: " + q );
```

Console ×

<terminated> Test [Java Applicat

```
Queue: [3, 2, 4, 1]
Remove 3. Queue: [2, 4, 1]
Peek   2. Queue: [2, 4, 1]
```

КОЛЕКЦИИ

Клас Stack:

- Основни методи

java.util

Class Stack<E>

```
public Object                push(Object);
public synchronized Object   pop();
public boolean               empty();
public synchronized Object   peek();
public synchronized int      search(Object);
```

КОЛЕКЦИИ

Клас Stack:

- пример

java.util

Class Stack<E>

```
53          Stack<Integer> stack = new Stack<>();
54          for(Integer i: List.of( 1, 3, 5, 7 )) {
55              stack.push(i);
56              System.out.println( "Pushed in stack "+i );
57          }
58          System.out.println( stack );
59          while( ! stack.isEmpty() )
60              System.out.println( stack.pop() );
```

```
Pushed in stack 1
Pushed in stack 3
Pushed in stack 5
Pushed in stack 7
[1, 3, 5, 7]
7
5
3
1
```

ВЪПРОСИ