ТЕХНОАТОМ

Java

Collections, Annotations, HTTP

Сергей Рыбалкин

Collections, Annotations, HTTP

ТЕХНОАТОМ

Отметьтесь на портале.

План лекции

ТЕХНОАТОМ

- Collection
- Map
- Annotations
- HTTP

План лекции

ТЕХНОАТОМ

- Collection
- Map
- Annotations
- HTTP

Iterator

TEXHOATOM

interface Iterator<E>

- boolean hasNext();
- E next();
- default void remove() {
  throw new UnsupportedOperationException("remove");
  }

Iterable

TEXHOATOM

interface Iterable<T>

- Iterator<T> iterator();
- E next();
- default void forEach(Consumer<? super T> action) {
  Objects.requireNonNull(action);
  for (T t: this) {
  action.accept(t);
  }
  }

Collection

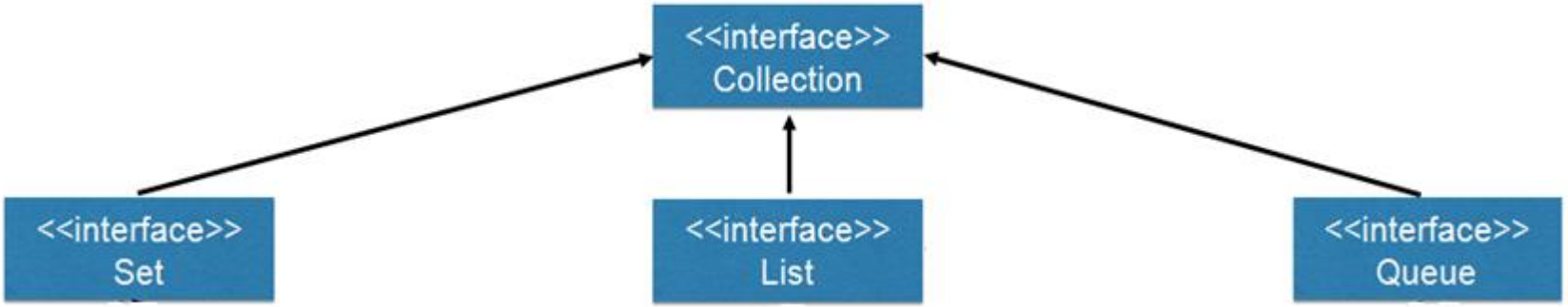
TEXHOATOM

interface Collection<E> extends Iterable<E>

- boolean contains(Object o)
- boolean add(E e);
- boolean remove(Object o);
- Iterator<E> iterator()
- int size();

Collection

TEXHOATOM



List

TEXHOATOM

interface List<E> extends Collection<E>

- E get(int index);
- E set(int index, E element);
- void add(int index, E element);
- E remove(int index);
- List<E> subList(int fromIndex, int toIndex);

Ordered collection (sequence)

Random access

TEXHOATOM

interface RandomAccess

Marker interface.
Indicate that List support fast (generally constant time) random access.

For generics algorithms.

List. Implementations

TEXHOATOM

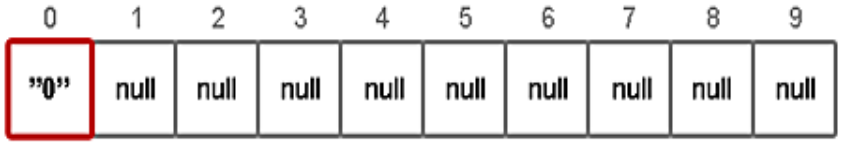
- ArrayList
- CopyOnWriteArrayList
- LinkedList
- Vector
- …

ArrayList

TEXHOATOM

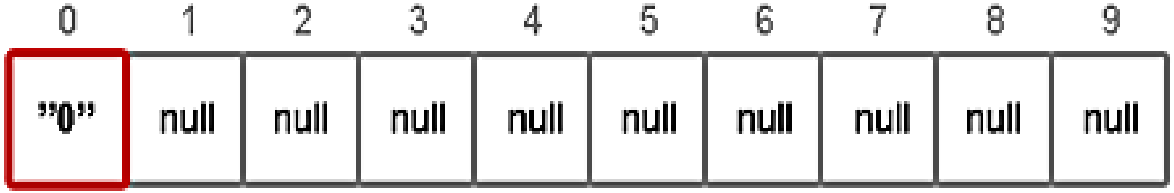
class ArrayList<E> implements List<E>

Also:
extends AbstractList<E>
Implements RandomAccess, Cloneable, Serializable



ArrayList

TEXHOATOM



ArrayList. Complexity

TEXHOATOM

contains	add	get	set	remove
O(n)	O(1) *	O(1)	O(1)	O(n)

CopyOnWriteArrayList

TEXHOATOM

class CopyOnWriteArrayList<E>
implements List<E>

Also:
Implements RandomAccess, Cloneable, Serializable

A thread-safe variant of ArrayList

CopyOnWriteArrayList. Complexity

TEXHOATOM

contains	add	get	set	remove
O(n)	O(n)	O(1)	O(n)	O(n)

LinkedList

TEXHOATOM

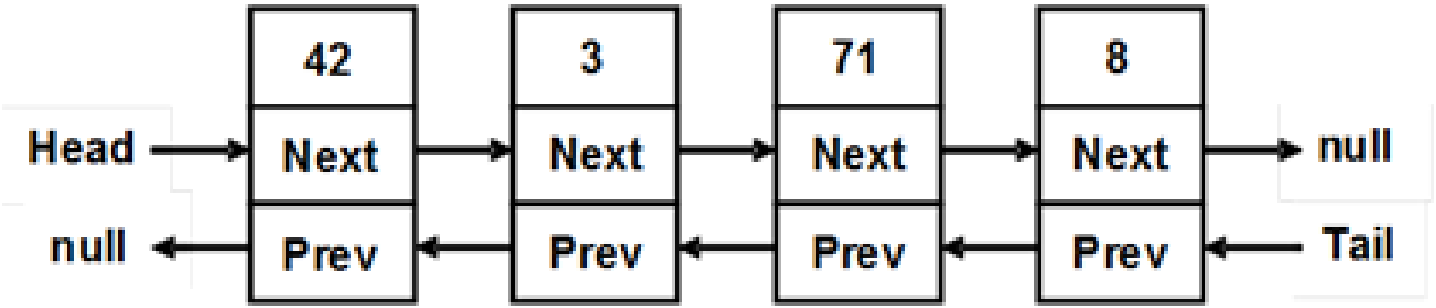
class LinkedList<E> implements List<E>

Also:
Implements Deque<E>, Cloneable, Serializable

Doubly-linked list implementation

LinkedList

TEXHOATOM



LinkedList. Complexity

TEXHOATOM

contains	add	get	set	remove
O(n)	O(1)	O(n)	O(n)	O(n)

Set

TEXHOATOM

interface Set<E> implements Collection<E>

A collection that contains no duplicate elements

Duplicate elements

TEXHOATOM

How to check?

Duplicate elements. Formal

TEXHOATOM

For each A != B in Set<E>
A.equals(B) == false

What about null?

Duplicate elements. Formal

TEXHOATOM

What about null?

At most one null element

Set. Restrictions

TEXHOATOM

Care about mutable objects!

The behavior of set is not specified if changes in object affects comparison.

Set. Implementation

TEXHOATOM

- HashSet
- LinkedHashSet
- EnumSet
- TreeSet
- CopyOnWriteArraySet
- …

HashSet

TEXHOATOM

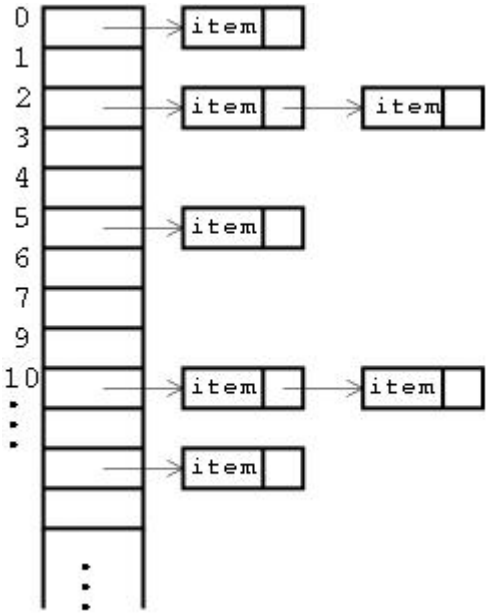
class HashSet<E> implements Set<E>

Also:
extends AbstractSet<E>
Implements Cloneable, Serializable

hashCode metters
Object::hashCode(): int

HashSet

TEXHOATOM



General contract

TEXHOATOM

For objects a and b:

- a.equals(b) => a.hashCode() == b.hashCode()
- if a.hashCode() == b.hashCode()
  a may be not equal b
- a.hashcode() is the same during object lifetime

HashSet. Complexity

TEXHOATOM

contains	add	remove
O(1)	O(1)	O(1)

TreeSet

TEXHOATOM

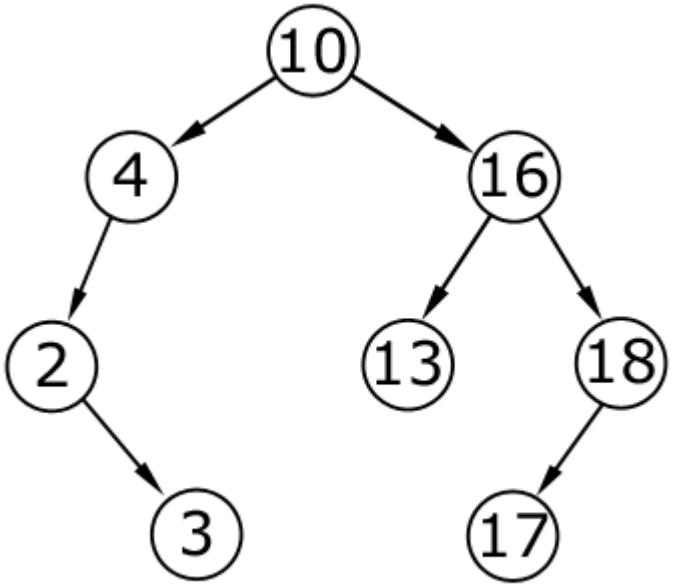
class HashSet<E> implements Set<E>

Also:
extends AbstractSet<E>
Implements SortedSet<E>, NavigableSet<E>, Cloneable, Serializable

ordering metters

TreeSet

TEXHOATOM



Comparator

TEXHOATOM

Functional interface Comparator<T>

static Comparator<Integer> *intComparator* =
(o1, o2) -> o1 - o2;

Comparable

TEXHOATOM

Functional interface Comparable<T>

```
@Override
public int compareTo(T o) {
        return this.field – o.field;
}
```

compareTo & equals

TEXHOATOM

Any type of contract?

compareTo & equals

TEXHOATOM

a.equals(b) == true => a.compareTo(b) == 0

What about null?

TreeSet. Complexity

TEXHOATOM

contains	add	remove
O(log(n))	O(log(n))	O(log(n))

План лекции

ТЕХНОАТОМ

- Collection
- **Map**
- Annotations
- HTTP

Map

TEXHOATOM

interface Map<K, V>

- An object that maps keys to values
- Cannot contain duplicate keys
- each key map to at most one value

Map

TEXHOATOM

- boolean containsKey(Object key);
- V get(Object key);
- V put(K key, V value);
- V remove(Object key);

Why, Map?

ТЕХНОАТОМ

Ну реально, почему не Collection.

Why, Map?

TEXHOATOM

From official FAQ:

>> This was by design.
>> We feel that mappings are not collections and collections are not mappings.
>> If a Map is a Collection, what are the elements?

Map. Implementations

TEXHOATOM

- HashMap
- TreeMap
- LinkedHashMap
- EnumMap
- …

HashMap

TEXHOATOM

HashSet is cutted HashMap

HashMap. Complexity

TEXHOATOM

containsKey	get	put	remove
O(1)	O(1)	O(1)	O(1)

TreeMap

TEXHOATOM

TreeSet is cutted TreeMap

TreeMap. Complexity

TEXHOATOM

containsKey	get	put	remove
O(log(n))	O(log(n))	O(log(n))	O(log(n))

Reference

TEXHOATOM

Java Collections – Performance (Time Complexity)

http://infotechgems.blogspot.ru/2011/11/java-collections-performance-time.html

План лекции

ТЕХНОАТОМ

- Collection
- Map
- **Annotations**
- HTTP

Annotations

TEXHOATOM

A form of syntactic metadata

May be available in run-time and compile-time

Annotations

TEXHOATOM

@Override
@Deprecated
@FunctionalInterface
@SuppressWarnings

…
@NotNull
@Nullable

Annotations. Example

TEXHOATOM

```
import java.lang.annotation.*;

@Target(ElementType.METHOD)
@Retention(RetentionPolicy.SOURCE)
public @interface Override {
}
```

Reflection API

TEXHOATOM

About:
https://docs.oracle.com/javase/tutorial/reflect/

In brief:
you can look for annotations and find annotated “things”
and then …

План лекции

ТЕХНОАТОМ

- Collection
- Map
- Annotations
- HTTP

HTTP

TEXHOATOM

Spec:

https://www.w3.org/Protocols/rfc2616/rfc2616-sec5.html

In brief:

```
GET  HTTP/1.1
Host: google.com
```

HTTP. Method

TEXHOATOM

- Get
- Post
- Put
- Delete
- …

HTTP. Post

TEXHOATOM

```
POST /auth/login
HTTP/1.1
Content-Type: application/x-www-form-urlencoded
Host: localhost:8080

login=user&password=password
```

HTTP. curl

TEXHOATOM

```
curl -X POST
      -H "Content-Type: application/x-www-form-urlencoded"
      -H "Host: localhost:8080"
      -d 'login=admin&password=admin'
http://localhost:8080/auth/login
```

```
curl
   -H 'Authorization: Bearer 2133e36c-8f31-455f-840e-1e034d4975fd'
http://localhost:8080/auth/dummy
```

HTTP. Client

TEXHOATOM

```
OkHttpClient client = new OkHttpClient();
MediaType mediaType = MediaType.parse("application/x-www-form-
                                      urlencoded");
RequestBody body = RequestBody.create(mediaType,
                                      "login=admin&password=admin");
Request request = new Request.Builder()
        .url("http://localhost:8080/auth/login")
        .post(body)
        .addHeader("content-type", "application/x-www-form-urlencoded")
        .addHeader("host", "localhost:8080")
        .build();

Response response = client.newCall(request).execute();
```

Оставьте отзыв.

ТЕХНОАТОМ

Спасибо за внимание!

Сергей Рыбалкин

s.rybalkin@corp.mail.ru

MVC

TEXHOATOM