Interface JavaContainer<T>

All Known Implementing Classes:
JavaSet, JavaVector

---

public interface JavaContainer<T>
JavaContainer

Method Summary

All Methods | Instance Methods | Abstract Methods

Modifier and Type
Method
Description

`void`
`add(T valueT)`
abstract add method which is used to add values to containers

`myIter`
`getIterator()`
abstract method for returning iterators from containers

`void`
`remove(T keyT)`
abstract remove method which is used to remove senden key value from containers

`int`
`size()`
abstract size method which is used to return used size in containers

`String`
`toString()`
abstract toString method

Method Details

add

void add(T valueT)
abstract add method which is used to add values to containers

Parameters:
`valueT` - value to add to container

remove

void remove(T keyT)
abstract remove method which is used to remove senden key value from containers

Parameters:
`valueT` - value to remove

size

int size()
abstract size method which is used to return used size in containers

Returns:
returns the containers used size

getIterator

myIter getIterator()
abstract method for returning iterators from containers

Returns:
returns the iterator with index 0

toString

String toString()
abstract toString method

Overrides:
`toString` in class `Object`
Returns:
returns the contents of Container in string form

Package HOMEWORK

Class JavaSet<T>

java.lang.Object
HOMEWORK.JavaSet<T>

All Implemented Interfaces:
JavaContainer<T>

---

public class JavaSet<T> extends Object implements JavaContainer<T>

Constructor Summary

Constructors
Constructor
Description
JavaSet()
we call javaset's parametirezed constructor if the default constructor gets called
JavaSet(int c)
main constructor of our javaset class

Method Summary

All Methods | Instance Methods | Concrete Methods

Modifier and Type
Method
Description
void
add(T val)
abstract add method which is used to add values to containers
boolean
equals(Object object)

T
getElementAt(int val)

returns the object at given index, simply an [] indexing operator
myIter
getIterator()
abstract method for returning iterators from containers
void
remove(T key)
abstract remove method which is used to remove senden key value from containers
int
size()
abstract size method which is used to return used size in containers
void
to_txt(Path p_set)
to_txt method to write contents of set to an txt file
String
toString()
Overriding the toString function for our Java Set class Using mutable StringBuilder class to create our string

Methods inherited from class java.lang.Object

clone, finalize, getClass, hashCode, notify, notifyAll, wait, wait, wait

Constructor Details

JavaSet

public JavaSet(int c)
main constructor of our javaset class

Parameters:
c - our innerArray's capacity gets the value of this parameter

JavaSet

public JavaSet()
we call javaset's parametirezed constructor if the default constructor gets called

Method Details

add

public void add(T val)
Description copied from interface: JavaContainer
abstract add method which is used to add values to containers

Specified by:
add in interface JavaContainer<T>
Parameters:
val - value to add to container

size

public int size()
Description copied from interface: JavaContainer
abstract size method which is used to return used size in containers

Specified by:
size in interface JavaContainer<T>
Returns:
returns the containers used size

remove

public void remove(T key)
Description copied from interface: JavaContainer
abstract remove method which is used to remove senden key value from containers

Specified by:
remove in interface JavaContainer<T>

getElementAt

public T getElementAt(int val)
returns the object at given index, simply an [] indexing operator

Parameters:
`val` - an int for wanted value's index
Returns:
returns the value at the given index

getIterator

public myIter getIterator()
Description copied from interface: `JavaContainer`
abstract method for returning iterators from containers

Specified by:
`getIterator` in interface `JavaContainer<T>`
Returns:
returns the iterator with index 0

toString

public String toString()
Overriding the toString function for our Java Set class Using mutable StringBuilder class to create our string

Specified by:
`toString` in interface `JavaContainer<T>`
Overrides:
`toString` in class `Object`
Returns:
returns the contents of inner array in string format

equals

public boolean equals(Object object)

Overrides:
`equals` in class `Object`

to_txt

public void to_txt(Path p_set)
to_txt method to write contents of set to an txt file

Parameters:
`p_set` - path variable to the txt file

Package HOMEWORK

Class JavaVector<T>

java.lang.Object
HOMEWORK.JavaVector<T>

All Implemented Interfaces:
JavaContainer<T>

---

public class JavaVector<T> extends Object implements JavaContainer<T>

Constructor Summary

Constructors
Constructor
Description
JavaVector()
default constructor of JavaVector calls parameterized constructor with default value of 20
JavaVector(int _c)
parameterized constructor of our JavaVector class, our innerArray's capacitiy gets assigned to the _c parameter this constructor also creates the iterator of our JavaVector

Method Summary

All Methods | Instance Methods | Concrete Methods

Modifier and Type
Method
Description
void
add(T val)
abstract add method which is used to add values to containers
boolean
equals(Object object)
equals method checks for two JavaVectors to see if they are equal if user sends same object it returns true if its not the same object method checks for two inner arrays of objects first checks their sizes if they are equal checks for

every element in the array if elements are not same nor in the same order it returns false
T
getElementAt(int val)
simply an indexing operator to return the element in given index
myIter
getIterator()
abstract method for returning iterators from containers
void
remove(T key)
abstract remove method which is used to remove senden key value from containers
int
size()
abstract size method which is used to return used size in containers
void
to_txt(Path p_set)
to_txt method to write contents of vector to an txt file
String
toString()
Overriding the toString function for our Java Vector class Using mutable StringBuilder class to create our string

Methods inherited from class java.lang.Object

clone, finalize, getClass, hashCode, notify, notifyAll, wait, wait, wait

Constructor Details

JavaVector

public JavaVector(int _c)
parameterized constructor of our JavaVector class, our innerArray's capacitiy gets assigned to the _c parameter this constructor also creates the iterator of our JavaVector

Parameters:
_c - capacity value

JavaVector

public JavaVector()
default constructor of JavaVector calls parameterized constructor with default value of 20

Method Details

add

public void add(T val)
Description copied from interface: JavaContainer
abstract add method which is used to add values to containers

Specified by:
add in interface JavaContainer<T>
Parameters:
val - value to add to container

size

public int size()
Description copied from interface: JavaContainer
abstract size method which is used to return used size in containers

Specified by:
size in interface JavaContainer<T>
Returns:
returns the containers used size

getElementAt

public T getElementAt(int val)
simply an indexing operator to return the element in given index

Parameters:
val - index of the wanted element
Returns:
element on the innerArray at given index

getIterator

public myIter getIterator()
Description copied from interface: JavaContainer
abstract method for returning iterators from containers

Specified by:
getIterator in interface JavaContainer<T>
Returns:
returns the iterator with index 0

remove

public void remove(T key)
Description copied from interface: JavaContainer
abstract remove method which is used to remove senden key value from containers

Specified by:
remove in interface JavaContainer<T>

equals

public boolean equals(Object object)
equals method checks for two JavaVectors to see if they are equal if user sends same object it returns true if its not the same object method checks for two inner arrays of objects first checks their sizes if they are equal checks for every element in the array if elements are not same nor in the same order it returns false

Overrides:
equals in class Object

toString

public String toString()
Overriding the toString function for our Java Vector class Using mutable StringBuilder class to create our string

Specified by:
toString in interface JavaContainer<T>
Overrides:
toString in class Object
Returns:
returns the contents of innerarray in string format

to_txt

public void to_txt(Path p_set)
to_txt method to write contents of vector to an txt file

Parameters:
p_set - path of the text file to write

Class myIter<T>

java.lang.Object
HOMEWORK.myIter<T>

All Implemented Interfaces:
`Iterator`<T>

---

public class myIter<T> extends Object implements Iterator<T>
myIterator class which implements Iterator interface

Constructor Summary

Constructors
Constructor
Description
`myIter(int _i, JavaContainer ref)`
myIter constructor which is called only from Containers constructors takes 2 parameters

Method Summary

All Methods | Instance Methods | Concrete Methods

Modifier and Type
Method
Description
`boolean`
`hasNext()`

`T`
`next()`
if there is an element in the referenced object's inner array it returns that element

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface java.util.Iterator

forEachRemaining, remove

Constructor Details

myIter

public myIter(int _i, JavaContainer ref)
myIter constructor which is called only from Containers constructors takes 2 parameters

Parameters:
_i - private member index get assigned to the taken _i argument
ref - private member reference gets assigned to the taken ref argument

Method Details

hasNext

public boolean hasNext()

Specified by:
hasNext in interface Iterator<T>

next

public T next() throws NoSuchElementException
if there is an element in the referenced object's inner array it returns that element

Specified by:
next in interface Iterator<T>
Throws:
NoSuchElementException - when there is none element in referenced object's inner array.

Package HOMEWORK

---

package HOMEWORK

- All Classes and Interfaces | Interfaces | Classes

Class
Description
JavaContainer<T>
JavaContainer
JavaSet<T>

JavaVector<T>

myIter<T>
myIterator class which implements Iterator interface
test
Driver test code for JavaContainers

Hierarchy For Package HOMEWORK

Class Hierarchy

- java.lang.Object
  - HOMEWORK.JavaSet<T> (implements HOMEWORK.JavaContainer<T>)
  - HOMEWORK.JavaVector<T> (implements HOMEWORK.JavaContainer<T>)
  - HOMEWORK.myIter<T> (implements java.util.Iterator<E>)
  - HOMEWORK.test

Interface Hierarchy

- HOMEWORK.JavaContainer<T>

Class test

java.lang.Object
HOMEWORK.test

---

public class test extends Object
Driver test code for JavaContainers

Constructor Summary

Constructors
Constructor
Description
test()

Method Summary

All Methods | Static Methods | Concrete Methods

Modifier and Type
Method
Description
static void
main(String[] args)

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructor Details

test

public test()

Method Details

main

public static void main(String[] args)